Dr.WEB®

Антивирус + Антиспам
для почтовых серверов UNIX

Защити созданное

Руководство администратора

Антивирус Dr.Web® для почтовых серверов UNIX
Версия 6.0.2
Руководство администратора
01.12.2014

Dr.Web, Центральный офис в России
125124
Россия, Москва
3-я улица Ямского поля, вл.2, корп.12А

Веб-сайт: www.drweb.com
Телефон: +7 (495) 789-45-87

Информацию о региональных представительствах и офисах Вы можете найти на официальном сайте компании.

«Доктор Веб»

«Доктор Веб» - российский разработчик средств информационной безопасности.

«Доктор Веб» предлагает эффективные антивирусные и антиспам-решения как для государственных организаций и крупных компаний, так и для частных пользователей.

Антивирусные решения семейства Dr.Web разрабатываются
с 1992 года и неизменно демонстрируют превосходные результаты детектирования вредоносных программ, соответствуют мировым стандартам безопасности.

Сертификаты и награды, а также обширная география пользователей свидетельствуют об исключительном доверии к продуктам компании.

Мы благодарны пользователям за поддержку решений семейства Dr.Web!

Содержание

Введение

В настоящей документации представлено описание следующих программных комплексов:

- **Антивирус + Антиспам Dr.Web® для почтовых серверов UNIX;**
- **Антиспам Dr.Web® для почтовых серверов UNIX;**
- **Антивирус Dr.Web® для почтовых серверов UNIX;**
- **Антивирус + Антиспам Dr.Web® для почтовых шлюзов UNIX;**
- **Антиспам Dr.Web® для почтовых шлюзов UNIX;**
- **Антивирус Dr.Web® для почтовых шлюзов UNIX.**

Фактически они представляют собой один и тот же программный комплекс, поставляемый в различных комплектациях, т.е. отличается только набор подключаемых модулей, лицензированных для работы в составе конкретного программного комплекса. В зависимости от набора модулей, подключенных к программному комплексу, может осуществляться взаимодействие с различными почтовыми системами, функционирование в качестве почтового шлюза, фильтрация почты от вирусов, спама и прочей нежелательной корреспонденции.

Также каждый из комплексов представлен в трёх вариантах для трёх основных UNIX-подобных операционных систем (далее – UNIX-систем): **Linux**, **FreeBSD** и **Solaris** x86.

Поскольку между этими программными комплексами для разных UNIX-систем немного принципиальных различий, в дальнейшем в документации речь будет идти, в основном, об общем случае **Dr.Web® для почтовых серверов UNIX** (далее – **Dr.Web для почтовых серверов UNIX**), а отличиям будут посвящены отдельные главы.

Руководство адресовано лицу, отвечающему за антивирусную безопасность и настройку сетей, называемому в данном руководстве "Администратором".

Проблема фильтрации электронной почтовой корреспонденции в UNIX-системах имеет два аспекта:

- во-первых, это проверка всего входящего SMTP-трафика на наличие вирусов, их диагностика и обезвреживание. При этом вирусы могут быть (и в большинстве случаев являются) отнюдь не специфичными для UNIX-систем. Через электронную почту распространяются обычные Windows-вирусы, в том числе и макровирусы для **Word, Excel** и других офисных приложений;
- во-вторых, это защита почты от спама и прочей нежелательной корреспонденции.

Программный комплекс **Dr.Web для почтовых серверов UNIX** выполняет обе перечисленные функции.

Круг задач, решаемых комплексом, ограничен только набором подключаемых к нему модулей: библиотек, отвечающих за непосредственную обработку сообщений. При необходимости имеется возможность расширить функциональность программного комплекса за счет добавления в него дополнительных модулей, которые могут быть разработаны пользователем на языках программирования C и C++. Эта возможность доступна, поскольку все модули решения используют унифицированный интерфейс взаимодействия (API). Доступ клиентских модулей к API продукта реализуется через использование заголовочных файлов и подключаемых библиотек (SDK). Пользователю доступны два SDK:

- SDK разработки новых модулей, выполняющих функции компонентов **Receiver**/**Sender**, для осуществления поддержки новых MTA.
- SDK создания новых подключаемых модулей, обрабатывающих почтовые сообщения.

За счет того, что разработанные пользователем модули будут использовать унифицированный интерфейс, они могут быть подключены к **Dr.Web для почтовых серверов UNIX** стандартным образом, так же как и модули, входящие в стандартную поставку.

SDK, включающие в себя требуемые заголовочные файлы, примеры реализации и документацию, не поставляются с продуктом, но доступны для скачивания из репозитория продуктов компании **«Доктор Веб»**. При необходимости обратитесь в техническую поддержку.

Программный комплекс **Dr.Web для почтовых серверов UNIX** состоит из следующих компонентов:

- **Консольный сканер** **Dr.Web Scanner** служит для обнаружения и лечения вирусов на локальной машине, в том числе и в каталогах общего доступа;
- **Резидентный компонент** **Dr.Web Daemon** используется в качестве подключаемого внешнего антивирусного фильтра;
- **Резидентный компонент** **Dr.Web Monitor** используется для запуска и перезапуска прочих модулей **Dr.Web** в нужном порядке;
- **Резидентный компонент** **Dr.Web Agent** используется для управления конфигурацией модулей **Dr.Web**, сбора статистической информации и интеграции с **Dr.Web Enterprise Security Suite** (**Dr.Web ESS**);

По умолчанию в состав решения включен **Dr.Web Agent**, предназначенный для интеграции с **Dr.Web ESS** версии 6.0. Если вы хотите интегрировать ваш продукт с **Dr.Web ESS** версии 10.0, потребуется выполнить установку обновления для **Dr.Web Agent** и произвести дополнительную настройку. Подробнее см. в разделе Dr.Web Agent.

- **Антивирусное ядро** **Dr.Web Engine** и набор постоянно обновляемых вирусных баз данных;
- **Компонент** **Dr.Web Updater**, выполненный в виде perl-скрипта, используется для автоматического обновления вирусных баз данных, а также правил фильтрации почты для спам-фильтра;
- **Комплексный компонент** **Dr.Web MailD**, осуществляющий анализ и обработку почтового трафика и интегрирующий **Dr.Web для почтовых серверов UNIX** с почтовыми системами **Sendmail**, **Postfix**, **Courier**, **Qmail**, **CommuniGate Pro**, **ZMailer**, **Exim**. Кроме того, комплексный компонент **Dr.Web MailD** реализует функционал почтового сервера, позволяющего обрабатывать непосредственно почтовый трафик, передаваемый по протоколу SMTP/LMTP. **Dr.Web MailD** также может работать в режиме централизованной защиты, в составе **Антивирусной сети Dr.Web**, под управлением **Dr.Web Enterprise Server** (входит в состав **Dr.Web Enterprise Security Suite**). **Dr.Web MailD** поставляется совместно с набором подключаемых модулей для обработки писем, в частности:
  - Антивирусный модуль **Drweb** (использует **Dr.Web Daemon** для проверки содержимого писем на вирусы и вредоносное ПО).
  - Модуль проверки на спам **Vaderetro**.
- **Веб-интерфейс управления** **Консоль Dr.Web для почтовых серверов UNIX** – модуль, интегрирующийся в системный компонент **Webmin** и используемый для управления и настройки **Dr.Web для почтовых серверов UNIX** через веб-интерфейс с любого браузера.

Структура компонентов **Dr.Web для почтовых серверов UNIX** изображена на рисунке ниже:

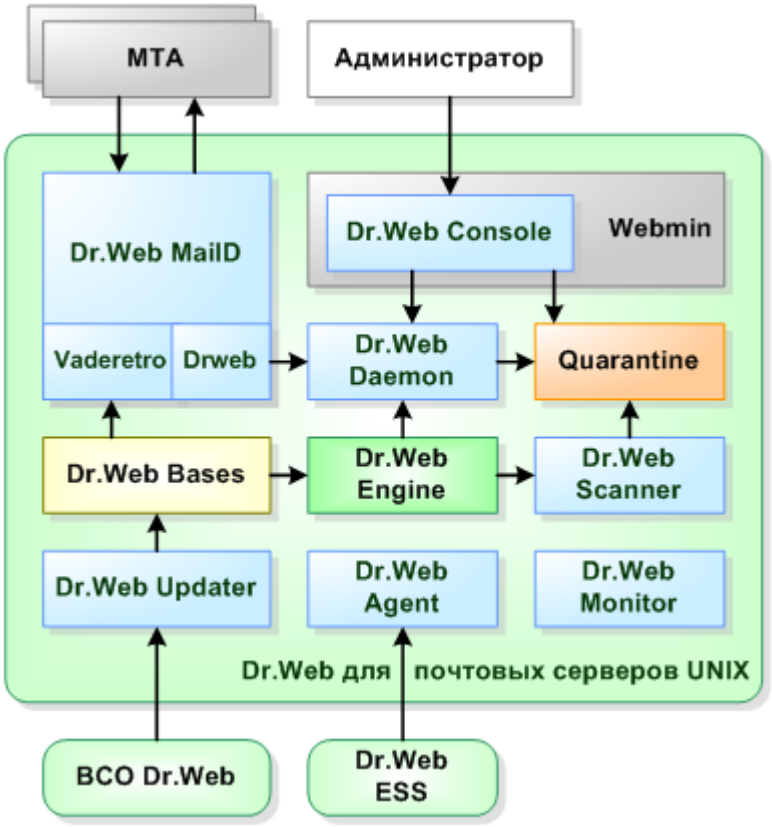


Рис. 1. Структура компонентов Dr.Web для почтовых серверов UNIX

В настоящем руководстве будет рассмотрен процесс настройки и использования программного комплекса **Dr.Web для почтовых серверов UNIX**, а именно:

- Общая характеристика продукта.
- Установка программного комплекса **Dr.Web для почтовых серверов UNIX**.
- Запуск программного комплекса **Dr.Web для почтовых серверов UNIX**.
- Использование модуля обновления **Dr.Web Updater**.
- Использование модуля **Dr.Web Agent**.
- Использование консольного сканера **Dr.Web Scanner**.
- Использование антивирусного модуля **Dr.Web Daemon**.
- Использование модуля **Dr.Web Monitor**.
- Настройка комплексного компонента **Dr.Web MailD**.
- Работа с веб-интерфейсом **Консоль Dr.Web для почтовых серверов UNIX**.

В заключении руководства приведена информация для контактов со службой технической поддержки.

Необходимо отметить, что продукты **Dr.Web** находятся в постоянном развитии. Обновления баз данных известных вирусов выходят ежедневно (как правило, несколько раз в день). Периодически появляются новые версии отдельных компонентов. Изменения в продуктах касаются как совершенствования приемов диагностики и борьбы с вирусами, так и средств интеграции с другими приложениями UNIX-систем. Кроме того, постоянно расширяется круг приложений, способных работать совместно с продуктами **Dr.Web**. Поэтому не исключено, что некоторые детали настройки и использования текущей версии будут отличаться от описанных в настоящем руководстве.

Используемые обозначения и сокращения

В данном руководстве используются следующие условные обозначения:

Обозначение	Комментарий
Полужирное начертание	Названия элементов графического интерфейса и примеры ввода, который необходимо выполнить в точности так, как он приведен в руководстве.
Зеленое и полужирное начертание	Наименования продуктов **Dr.Web** или их компонентов.
Зеленое и подчеркнутое начертание	Ссылки на страницы руководства и веб-сайты.
`Моноширинный шрифт`	Примеры кода, ввода для командной строки и информации, выводимой пользователю приложением.
Курсив	Термины и замещающий текст (приводится вместо информации, которую необходимо ввести пользователю). В случае примеров ввода командной строки курсив указывает на значения параметров.
ЗАГЛАВНЫЕ БУКВЫ	Названия клавиш клавиатуры.
Знак плюса (+)	Указывает на одновременность нажатия клавиш клавиатуры. Например, запись ALT+F1 обозначает, что необходимо нажать клавишу F1, удерживая нажатой клавишу ALT.
Восклицательный знак	Важное замечание или предупреждение о потенциально опасных или чреватых ошибками ситуациях.

Также для указания каталогов, в которые устанавливаются компоненты программного комплекса, используются условные обозначения `%bin_dir`, `%etc_dir` и `%var_dir`. В зависимости от ОС эти обозначения указывают на следующие каталоги:

для Linux и Solaris:

```
%bin_dir = /opt/drweb/
%etc_dir = /etc/drweb/
%var_dir = /var/drweb/
```

для FreeBSD:

```
%bin_dir = /usr/local/drweb/
%etc_dir = /usr/local/etc/drweb/
%var_dir = /var/drweb/
```

В документе используются следующие термины и сокращения:

Сокращение	Расшифровка
ASCII	American Standard Code for Information Interchange — американская стандартная кодировочная таблица для печатных символов и некоторых специальных кодов
CIDR	Classless Inter-Domain Routing — метод IP-адресации, позволяющий гибко управлять пространством IP-адресов, не используя жёсткие рамки классовой адресации
DEB	Расширение имён файлов «бинарных» пакетов для распространения и установки программного обеспечения в ОС проекта **Debian** и других, использующих систему управления пакетами **dpkg**
DNS	Domain Name System — компьютерная распределённая система для получения информации о доменах
HTML	HyperText Markup Language — язык разметки гипертекста, стандартный язык разметки Web-документов
IP	Internet Protocol — маршрутизируемый межсетевой протокол сетевого уровня семейства TCP/IP

Сокращение	Расшифровка
IPv4	Протокол IP, версия 4
IPv6	Протокол IP, версия 6
IPC	Inter-Process Communication — набор способов обмена данными между множеством потоков в одном или более процессах, запущенных на одном или более компьютерах, связанных между собой сетью
MD5	Message Digest 5 — 128-битный алгоритм хеширования, предназначенный для создания «отпечатков» или дайджестов сообщения произвольной длины и последующей проверки их подлинности
PID	Process IDentifier — уникальный идентификатор, присваиваемый ОС экземпляру процесса при его запуске
POSIX	Portable Operating System Interface for Unix — набор стандартов, определяющих интерфейсы взаимодействия между операционной системой и прикладной программой, созданный для обеспечения совместимости различных UNIX-подобных операционных систем и переносимости прикладных программ на уровне исходного кода
RFC	Request for Comments — документ из серии пронумерованных информационных документов Интернета, содержащих технические спецификации и стандарты, широко применяемые во всемирной сети
RPM	Формат пакетов распространения программного обеспечения и название менеджера управления ими
SSL	Secure Socket Layers — так же как и TLS – криптографический протокол, обеспечивающие защищённую передачу данных между узлами в сети Интернет
TCP	Transmission Control Protocol — один из основных сетевых протоколов Интернета, предназначенный для управления передачей данных в сетях и подсетях TCP/IP
TLS	Transport Layer Security — криптографический протокол, обеспечивающий защищённую передачу данных между узлами в сети Интернет. Использует асимметричную криптографию для обмена ключами, симметричное шифрование для конфиденциальности и коды аутентичности для сохранения целостности сообщений
URL	Uniform Resource Locator — единообразный локатор (определитель местонахождения) ресурса. Стандартизированный способ записи адреса ресурса в сети Интернет
UUID	Unique User IDdentifier — уникальный идентификатор пользователя
XML	eXtensible Markup Language — расширяемый язык разметки, текстовый формат, предназначенный для хранения структурированных данных, обмена информацией между программами, а также для создания на его основе более специализированных языков разметки
ОС	Операционная система — комплекс управляющих и обрабатывающих программ, предназначенных для управления устройствами, вычислительными процессами, эффективного распределения вычислительных ресурсов между вычислительными процессами и организации надёжных вычислений

В разделе описания работы компонента обработки почты **Dr.Web MailD** и **Консоль Dr.Web для почтовых серверов UNIX** используются следующие термины и сокращения:

Сокращение	Расшифровка
A-запись	Тип записи в DNS, указывающий соответствие между именем хоста и IP-адресом
API	Application Programming Interface — набор готовых классов, процедур, функций, структур и констант, предоставляемых приложением (библиотекой, сервисом) для использования во внешних программных продуктах
CGI	Common Gateway Interface — стандарт интерфейса, используемого для связи внешней программы с веб-сервером
CTE	Content Transfer Encoding — стандарт кодирования содержимого объектов MIME (7- или 8-битное кодирование) для сообщений электронной почты

Сокращение	Расшифровка
DNSBL	DNS blacklist (DNS blocklist) — списки хостов, подозрительных на рассылку спама, хранимые с использованием системы архитектуры DNS, используются почтовыми серверами для борьбы со спамом
DSN	Delivery Status Notification — сообщение электронной почты, которое отсылается почтовым сервером отправителю в случае ошибки при обработке сообщения. Обычно в тексте DSN указывается текст ошибки, адрес почтового ящика, список кодов ошибок и причина, по которой письмо не могло быть доставлено
DSN	Data Source Name — имя источника данных, используемого для доступа к данным через ODBC
FQDN	Fully Qualified Domain Name — Полностью определённое имя домена, т.е. имя домена, не имеющее неоднозначностей в определении. Включает в себя имена всех родительских доменов иерархии DNS
HAM	Противоположность спаму – письма, квалифицированные как "заведомо не спам".
IMAP	Internet Message Access Protocol — протокол прикладного уровня для доступа к электронной почте
JSON	JavaScript Object Notation — текстовый формат обмена данными, основанный на JavaScript
LDAP	Lightweight Directory Access Protocol («облегчённый протокол доступа к каталогам») — протокол прикладного уровня для доступа к службе каталогов X.500
LMTP	Local Mail Transfer Protocol — протокол локальной пересылки почты, производный от SMTP. Используется в ситуациях, когда получающая сторона не использует очередь сообщений, например, сервер хранения почты, работающий, как Mail delivery agent
MIME	Multipurpose Internet Mail Extensions — стандарт, описывающий передачу различных типов данных по электронной почте, а также спецификация для кодирования информации и форматирования сообщений таким образом, чтобы их можно было пересылать через Интернет
MDA	Mail Delivery Agent – агент доставки почты (как правило, составная часть почтового сервера)
MTA	Mail Transfer Agent — почтовый сервер или его составная часть, ответственная за прием и передачу почты
MUA	Mail User Agent — Почтовая программа (клиент электронной почты) — программное обеспечение, устанавливаемое на компьютере пользователя и предназначенное для получения, написания, отправки и хранения сообщений электронной почты
MX-запись	Тип записи в DNS, указывающий способ маршрутизации электронной почты. MX-запись домена указывает серверы, на которые нужно отправлять электронную почту, предназначенную для адресов в данном домене
ODBC	Open Database Connectivity — программный интерфейс (API) доступа к реляционным базам данных (или к любым источникам данных, которые могут быть представлены в реляционной форме). Представляет собой стандартный интерфейс для получения и отправки источникам данных различных типов
POP3	Post Office Protocol Version 3 — стандартный Интернет-протокол прикладного уровня, используемый клиентами электронной почты для извлечения электронного сообщения с удаленного сервера по TCP/IP-соединению
SASL	Simple Authentication and Security Layer — фреймворк для предоставления аутентификации и защиты данных в протоколах на основе соединений
SMTP	Simple Mail Transfer Protocol — широко используемый сетевой протокол, предназначенный для передачи электронной почты в сетях TCP/IP
SQL	Structured Query Language — язык, применяемый для создания, модификации и управления данными в реляционных базах данных
TCP	Transmission Control Protocol — один из основных сетевых протоколов Интернета, предназначенный для управления передачей данных в сетях и подсетях TCP/IP

Сокращение	Расшифровка
БД	База данных — некоторый набор постоянно хранимых данных, используемых прикладными программными системами
СУБД	Система управления базами данных — программное средство, обеспечивающее управление созданием и использованием баз данных

Системные требования

Компоненты программного комплекса **Dr.Web для почтовых серверов UNIX** совместимы:

- с дистрибутивами **Linux**, удовлетворяющим требованиям, приведенным в разделе Совместимость с дистрибутивами Linux;
- с **FreeBSD** версии 6.x и выше для платформы Intel x86 и amd64;
- с **Solaris** версии 10 для платформы Intel x86 и amd64.

Используемая платформа должна обеспечивать полную поддержку системы команд процессора архитектуры x86 в 32-битном и 64-битном режимах. На 64-битных системах обязательно должна быть включена поддержка выполнения 32-битных приложений.

Продукты, работающие под управлением операционной системы **FreeBSD** 6.x, не могут быть подключены к серверу **Dr.Web ESS** версии 10.

Пример:

Для поддержки 32-битных приложений в системах на основе **Debian**/**Ubuntu Linux** понадобится установить библиотеку `libc6-i386`, а для систем на основе **ALT Linux** – библиотеку `i586-glibc-core`.

Для успешной и стабильной работы **Dr.Web для почтовых серверов UNIX** требуются:

- Установленный и запущенный **Dr.Web Daemon** и Антивирусное ядро **Dr.Web Engine** версии не ниже 6.0.2.
- Модуль обновления **Dr.Web Updater** требует установленный **Perl** 5.8.0 и выше.

С точки зрения аппаратного обеспечения требования программного комплекса **Dr.Web для почтовых серверов UNIX** совпадают с требованиями консольного (текстового) режима операционной системы, для которой он предназначен. Для установки требуется около 190 Мбайт дискового пространства.

Для работы графического инсталлятора **Dr.Web для почтовых серверов UNIX** требуется **X Window System.** Для работы установочного скрипта в графическом режиме необходимо, чтобы в системе был установлен эмулятор терминала `xterm` или `xvt`.

Также в системе должны быть установлены следующие пакеты и утилиты:

- `base64`
- `unzip`
- `crond`

Для корректной работы **Dr.Web для почтовых серверов UNIX** в операционной системе **FreeBSD** старше восьмой версии необходимо наличие библиотеки `compat7x`.

В зависимости от задач, решаемых программным комплексом **Dr.Web для почтовых серверов UNIX**, и рабочей нагрузки, к аппаратному обеспечению компьютера могут предъявляться дополнительные требования.

Совместимость с дистрибутивами Linux

Программный комплекс **Dr.Web для почтовых серверов UNIX** поддерживает дистрибутивы **Linux x86** и **x86-64**.

Требования к версии **ядра** ОС и библиотеке **glibc** зависят от типа установочного пакета:

- Универсальный пакет для UNIX-систем (**Linux x86**):
  - версия **ядра** 2.4.x, версия **glibc** 2.2 (не рекомендуется) и выше,
  - либо версия **ядра** 2.6.x, версия **glibc** 2.3 и выше;
- Универсальный пакет для UNIX-систем (**Linux x86-64**):
  - версия **ядра** 2.6.x, рекомендована версия **glibc** 2.3 и выше;
- Пакеты RPM (**rpm-apt**, **urpmi**, **yum**, **zypper**):
  - версия **ядра** 2.6.18 и выше, версия **glibc** 2.5 и выше;
- Пакеты DEB (**apt**):
  - версия **ядра** 2.6.26 и выше, версия **glibc** 2.7 и выше;

Работоспособность комплекса протестирована на следующих дистрибутивах:

Дистрибутив Linux	Версии	
	32-бит	64-бит
ALT Linux	4.0 – 5.0 СПТ 6.0	5.0 СПТ 6.0
Arch Linux	–	все
ASPLinux	12.0 – 14.0	–
Debian	3.1 – 6.0	4.0 – 6.0
Fedora	–	14.0
Gentoo	все	
Mandriva Linux	старше 2009, CS4	2010.x
Mandrake	10.x	10.x
openSUSE	10.3 – 11.0	10.3 – 11.0
PCLinux	2010	2010
RedHat Enterprise Linux (RHEL)	4.0 – 6.0	5.0 – 6.0
Suse Linux Enterprise Server	9.0 – 11.0	10.0 – 11.0
Ubuntu	7.04 – 11.04	7.04 – 11.04

Совместимость с ОС МСВС

Дистрибутив совместим со следующими версиями операционной системы **МСВС**:

- **МСВС** 3.0 80001-12 (изм. 0, 1, 2, 3);
- **МСВС** 3.0 80001-14 (изм. 0, 1, 2);
- **МСВС** 3.0 80001-08;
- **МСВС** 3.0 80001-16;
- **МСВС** 3.0 ФСТЭК.

Прочие дистрибутивы **Linux**, которые соответствуют приведенным выше требованиям, тоже поддерживаются, но не были протестированы. При возникновении проблем с совместимостью с

вашим дистрибутивом, обратитесь в техническую поддержку: http://support.drweb.com/request/.

Расположение файлов продукта

По умолчанию **Dr.Web для почтовых серверов UNIX** устанавливается в каталоги `%bin_dir`, `%etc_dir` и `%var_dir`. В этих каталогах создается структура подкаталогов, не зависящая от ОС:

`%bin_dir/` – Исполняемые модули программного комплекса и модуль обновления компонентов **Dr.Web Updater** (perl-скрипт `update.pl`).

`%bin_dir/doc/` – Документация по продукту. Вся документация представлена в виде текстовых файлов и присутствует в двух вариантах — англоязычном и русскоязычном (в кодировках `KOI8-R` и `UTF-8`).

`%bin_dir/lib/` – Различные служебные библиотеки, и вспомогательные файлы, необходимые для работы компонентов программного комплекса, например:

- `ru_scanner.dwl` – файл языковых ресурсов модуля **Dr.Web Scanner**.

`%bin_dir/scripts/`,
`%bin_dir/maild/scripts/` – Дополнительные скрипты, скрипт автоконфигурации **Dr.Web для почтовых серверов UNIX**, скрипт миграции для переноса конфигурационных параметров со старых версий продуктов **Dr.Web**.

`%bin_dir/web/` – Модуль веб-интерфейса **Dr.Web для почтовых серверов UNIX** для подключения к **Webmin**.

`%etc_dir/` – Конфигурационные файлы программного комплекса, а также `enable`-файлы, управляющие запуском компонентов, работающих в режиме демонов*.

`%etc_dir/agent/` – Дополнительные конфигурационные файлы модуля **Dr.Web Agent**.

`%etc_dir/monitor/` – Дополнительные конфигурационные файлы модуля **Dr.Web Monitor**.

`%etc_dir/maild/templates/` – Шаблоны уведомлений, которые генерируются и высылаются различным типам получателей при обнаружении в письме вредоносных объектов, а также при возникновении ошибок в работе демона **Dr.Web Daemon** или подключаемых модулей.

`%var_dir/bases/` – Вирусные базы (файлы `*.vdb`).

`%var_dir/infected/` – Каталог **Карантина** для перемещения в него зараженных файлов, если такая реакция на обнаружение зараженных или подозрительных файлов задана в настройках компонентов программного комплекса.

`%var_dir/lib/` – Антивирусное ядро в виде подгружаемой библиотеки **`(drweb32.dll)`**.

*) Расположение `enable`-файлов зависит от способа установки **Dr.Web для почтовых серверов UNIX**:

- Установка при помощи **универсального пакета для UNIX**:
  Файлы располагаются в каталоге `%etc_dir` и называются
  `drwebd.enable`,
  `drweb-monitor.enable`.
- Установка из **нативных DEB-пакетов**:
  Файлы располагаются в каталоге `/etc/defaults` и называются
  `drwebd`,
  `drweb-monitor`.
- Установка из **нативных RPM-пакетов**:
  Файлы располагаются в каталоге `/etc/sysconfig` и называются
  `drwebd.enable`,
  `drweb-monitor.enable`.

Конфигурационные файлы

Общий формат конфигурационных файлов

Настройка большинства компонентов программного комплекса **Dr.Web для почтовых серверов UNIX** производится с помощью конфигурационных файлов. Конфигурационные файлы являются текстовыми файлами, что позволяет редактировать их любым текстовым редактором).

Общий формат файла конфигурации:

```
--- начало файла ---

[Имя секции 1]
Параметр1 = значение1, ..., значениеK
...
ПараметрN = значение1, ..., значениеK

[Имя секции X]
Параметр1 = значение1, ..., значениеK
...

--- конец файла ---
```

Файлы конфигурации формируются по следующему принципу:

- Символы ";" или "#" в строках конфигурационного файла обозначают начало комментария – весь текст, идущий в строке за этими символами, пропускается модулями **Dr.Web для почтовых серверов UNIX** при чтении параметров из конфигурационного файла.
- Содержимое файла разбивается на последовательность именованных секций. Возможные имена секций жестко заданы и не могут быть произвольными. Имя секции задается в квадратных скобках.
- Каждая секция содержит группу параметров конфигурации, объединенных по смыслу.
- В одной строке файла задается значение только одного параметра.
- Основной формат задания значения параметра (пробелы, окружающие символ '=', если встречаются, игнорируются):

  ```
  <Имя параметра> = <Значение>
  ```

- Возможные имена параметров жестко заданы и не могут быть произвольными.
- Все имена секций и параметров в файле регистронезависимы.
- Порядок следования секций в файле и параметров внутри секций не имеет значения.
- Значения параметров в конфигурационном файле могут быть заключены в кавычки (и должны быть заключены в кавычки в том случае, если содержат пробелы).
- Некоторые параметры могут иметь несколько значений, в этом случае значения параметра разделяются запятой, или значение параметра задается несколько раз в разных строках конфигурационного файла. При перечислении значений параметра через запятую пробелы между значением и запятой, если встречаются, игнорируются. Если пробел является частью значения, всё значение необходимо заключить в кавычки.

Возможность присвоения параметру несколько значений в данном документе указывается явно. Если для некоторого параметра в данном документе или в комментариях в файле конфигурации явно не указано, что ему можно присвоить несколько значений, то параметр может обладать только одним значением.

Пример задания параметра, имеющего несколько значений:

1) Перечисление нескольких значений через запятую:

```
Parameter = Value1, Value2,"Value 3"
```

2) Задание тех же значений параметра в разных строках конфигурационного файла:

```
Parameter = Value2
Parameter = Value1
Parameter = "Value 3"
```

Если какой-либо параметр не задан (отсутствует) в конфигурационном файле, это не означает, что у данного параметра нет значения. В таких случаях значение параметра считается заданным по умолчанию. Лишь некоторые параметры являются необязательными или не имеют значений по умолчанию, о чем, как правило, упоминается отдельно.

Правила описания параметров, принятые в данном документе

В данном руководстве каждый параметр описывается следующим образом:

[Статус использования в Правилах] **ИмяПараметра** = {Тип параметра | Возможные значения}	Описание параметра. {Может ли иметь несколько значений}. {Особые замечания} {Важные замечания}
	Значение по умолчанию: **ИмяПараметра** = {значение | отсутствует}

Статус использования в Правилах обозначается с использованием следующих пиктограмм:

- R Параметр может быть использован в SETTINGS-части Правил обработки писем для временного изменения его значения при обработке конкретного письма, для которого условная часть правила истинна.
- A Параметр при использовании в Правилах обработки писем имеет "аддитивную" (накапливающую) семантику, т.е. если для письма истинно несколько Правил, задающих разное значение этого параметра, то в качестве значения параметра выступает объединенный список его значений из сработавших Правил.
- C Параметр при использовании в Правилах обработки писем поддерживает клонирование писем, т.е. если у письма несколько получателей, и для разных получателей письма истинны разные Правила, задающие *различные* значения этого параметра, то письмо будет клонировано (по числу получателей), и к каждой копии письма в качестве значения параметра будет использовано значение из Правила, истинного для этого письма.

Если Статус использования в Правилах для параметра не указан, то данный параметр не может быть использован в Правилах обработки писем.

Описание параметров и секций конфигурационных файлов дано в порядке их следования в файле конфигурации, создаваемом при установке программного комплекса **Dr.Web для почтовых серверов UNIX**.

Поле Тип параметра может принимать следующие значения:

- **числовое значение (numerical value)** — значение параметра является целым неотрицательным числом.
- **время (time)** — значение параметра задается в единицах измерения времени. Значение состоит из целого числа, после которого может идти буква, определяющая вид единиц измерения времени (s – секунды, m – минуты, h – часы, регистр букв не учитывается). Если в значении параметра буквы нет, то считается, что время задано в секундах.

  Примеры: 30s, 15m

- **размер (size)** — значение параметра задается в единицах измерения объема памяти (дисковой или оперативной). Значение состоит из целого числа, после которого может идти буква, определяющая вид единиц измерения объема памяти (`b` – байты, `k` – килобайты, `m` – мегабайты, `g` – гигабайты, регистр букв не учитывается). Если в значении параметра буквы нет, то считается, что размер задан в байтах.

  Примеры: `20b`, `15k`

- **права (permissons)** — значение параметра задаётся трехзначным числом, обозначающим права доступа к файлам в формате, принятом в UNIX-системах.
  Каждое право является комбинацией (суммой) трех базовых прав:
  - Право чтения (r) обозначается числом 4;
  - Право записи (w) обозначается числом 2;
  - Право исполнения (x) обозначается числом 1.

  При этом первая цифра числа задает права для владельца файла, вторая – для группы владельцев файла, а третья – для всех остальных, не являющихся ни владельцами, ни членами соответствующей группы.

  Примеры: `755`, `644`

- **логический (Yes/No)** — Логический тип, значения которого представляются строками "`Yes`" и "`No`".

- **путь к файлу/каталогу** (`path to file/directory`) — строка, задающая расположение файла или каталога в файловой системе. Помните, что в ОС семейства Linux/ UNIX имена файлов и каталогов регистрозависимы. Если указано, что значением параметра может быть **маска**, то в качестве значений параметра можно использовать файловые маски, содержащие следующие специальные символы:
  - `?` – замещает любой один символ;
  - `*` – замещает любую (в том числе пустую) последовательность символов.

  Пример: "`?.e*`" – маска, под которую подпадают файлы, имя которых состоит из любого одного символа, а расширение любой длины, и начинается с буквы '`e`' (`x.exe`, `g.e`, `f.enable` и т.п.).

- **действие (action)** — строка, содержащая наименование действий, совершаемых над объектами, вызвавшими какую-либо реакцию компонентов программного комплекса **Dr.Web для почтовых серверов UNIX**. В некоторых случаях для параметра можно задать одно основное действие и до трех дополнительных. Тип параметра в этом случае называется **список действий (actions list)**. Основное действие в этом случае всегда должно быть первым в списке. Для разных параметров набор допустимых действий может различаться, и в этом случае он указывается отдельно для каждого параметра. Общий перечень действий, которые могут использоваться, см. ниже.

- **адрес (address)** — строка, содержащая адрес сокета компонента **Dr.Web для почтовых серверов UNIX** или внешнего модуля или программы.
  Имеет формат `ТИП:АДРЕС`. Допустимы следующие типы:
  - `inet` — используются TCP-сокеты, `АДРЕС` имеет формат `ПОРТ@ИМЯ_УЗЛА`. `ИМЯ_УЗЛА` может быть как прямым IP-адресом, так и доменным именем узла.

    Пример:

    ```
    Address = inet:3003@localhost
    ```

  - `local` — используются локальные UNIX-сокеты, в этом случае адрес является путем к файлу сокета.

    Пример:

    ```
    Address = local:%var_dir/.daemon
    ```

  - `pid` — реальный адрес процесса должен быть прочитан из его PID файла. Такой тип адреса доступен лишь в некоторых случаях и при возможности его использования в

описании параметра это указывается явно.

- **текст (text value), строка (string)** — значение параметра задается в виде текстовой строки, текст в строке может быть заключен в кавычки (если в строке есть пробелы, то кавычки обязательны).
- **настройки пула (pool options)** – настройки пула потоков. Имеют специальный формат, описанный в разделе Специальные типы параметров.
- **Lookup** — строки, задающие разделенные запятыми объекты для поиска.
- **LookupLite** — упрощенный `Lookup`, в котором можно указывать только либо непосредственное значение, либо `Lookup` типа `file`.
- **хранилище (Storage)** — объекты для хранения данных. Синтаксис аналогичен `Lookup`, за исключением использования другого списка префиксов и того, что в `Storage` нельзя использовать макрос `$s`.
  Подробнее о типах `Lookup`, `LookupLite` и `Storage` см. в разделе Lookup.
- **настройки TLS/SSL (TLSSettings)** — настройки для работы шифрованного соединения с использованием криптографических протоколов TLS и SSL. Имеют специальный формат, описанный в разделе Специальные типы параметров.
- **список строк (strings list)** — набор текстовых значений, разделенных запятыми.

  Если значение параметра соответствует шаблону `file:/path_to_file` (где `path_to_file` – путь к файлу), то текстовые значения получаются из указанного в параметре файла. Каждое значение в файле должно записываться в отдельной строке. Если при получении информации из файла произошла ошибка, в файл журнала выводится соответствующее диагностическое сообщение и загрузка программы продолжится.
- **уровень подробности (log level)** — строка, указывающая уровень подробности вывода информации в некоторый журнал или в службу **syslog**.
- **возможные значения (value)** — параметр имеет тип, не описанный в предыдущих пунктах данного списка. В этом случае перечисляется список разрешенных для него значений.

Поведение модулей при некорректно заданных файлах конфигурации

- Если значение какого-либо параметра задано некорректно, **Dr.Web для почтовых серверов UNIX** выводит сообщение об ошибке и завершает свою работу.
- Если при загрузке какого-либо конфигурационного файла в нем обнаруживаются неизвестные параметры, работа программы продолжается в нормальном режиме, но в файл журнала выводится соответствующее предупреждение.

Некоторые параметры могут использовать в качестве значений регулярные выражения (для каждого параметра отмечается в его описании). По умолчанию используется синтаксис регулярных выражений **Perl**. С основами регулярных выражений вы можете ознакомиться, например, в **Wikipedia** (статья "Регулярные выражения").

Ведение журналов (логов)

Все компоненты программного комплекса **Dr.Web для почтовых серверов UNIX** ведут журналы (логи) своей работы. Для каждого компонента имеется возможность указать способ ведения журнала (самостоятельная запись событий в файл или использование системной службы журналирования **syslog**).

Уровень подробности ведения журнала работы компонента может быть как очень высоким (например, если задано значение **Debug** для отладочных целей), так и отсутствовать вовсе (например, если задано значение **Quiet**, когда файл журнала не ведется).

Для задания уровня подробности используется параметр с именем **LogLevel**. Также некоторые

модули могут иметь дополнительные параметры, регулирующие уровни подробности вывода некоторых сообщений в журнал (например, вывод сообщений подситемы IPC, там, где она используется, регулируется параметром **IPCLevel)** .

Если в настройках модуля отсутствуют параметр конфигурации **LogLevel**, то это означает, для него регулирование уровня подробности в журнал невозможно. По умолчанию в этом случае используется уровень журналирования, примерно равный `Debug`.

Используемые уровни подробности ведения журнала

Значения параметров, отвечающих за уровень подробности ведения журнала работы компонентов в общем случае могут задаваться из следующего набора (упорядочен от менее к более подробным):

- **Quiet** – Уровень "Тишина". Запись событий в журнал не ведется.
- **Error** – Уровень "Ошибки". Фиксируются записи только об критических ошибках.
- **Alert** – Уровень "Тревога". Фиксируются записи об ошибках и важных предупреждениях.
- **Warning** – Уровень "Предупреждения". Фиксируются записи об ошибках, важных и обычных предупреждениях.
- **Info** – Уровень "Информационный". Ведется запись сообщений об ошибках, предупреждениях и информационных сообщений.
- **Notice** – Уровень "Уведомительный". То же, что и "Информационный", но добавляются записи уведомлений.
- **Debug** – Уровень "Отладочный", То же, что и "Уведомительный", но добавляются записи отладочной информации.
- **Verbose** – Уровень "Подробный", ведется запись в журнал всех возможных сообщений (режим не рекомендуется из-за большого объема информации, выводимой в журнал, что тормозит как работу приложения, так и службу журналирования **syslog** операционной системы, если она используется).

Для каждого модуля **Dr.Web для почтовых серверов UNIX** набор допустимых уровней подробности может различаться, о чем указано в описании соответствующих параметров

Использование службы журналирования syslog

При использовании для ведения службы журналирования **syslog** кроме указания уровня подробности ведения журнала указывается также метка-источник сообщений, которая может быть использована службой **syslog** для внутренней маршрутизации сообщений по разным фаилам журналов. Эти правила маршрутизации настраиваются в собственном файле конфигурации демона службы **syslog** (обычно `/etc/syslogd.conf`).

Метка, присваиваемая сообщениям для службы **syslog**, указывается в конфигурационных файлах в параметре **SyslogFacility**.

Допускается использование следующих меток:

- `Daemon` – От имени резидентного системного сервиса (демона);
- `Local0`, ..., `Local7` – От имени локального пользовательского приложения (зарезервировано 8 номеров 0-7);
- `Kern` – От имени ядра системы;
- `User` – От имени пользовательского процесса;
- `Mail` – От имени почтовой системы.

Пожалуйста, обратите внимание, что при использовании **syslog** в файле конфигурации может дополнительно присутствовать параметр подробности ведения журнала, используемый для системы **syslog**. Этот параметр имеет название **SyslogPriority** и может принимать те же

значения, что и основной параметр уровня подробности **(LogLevel)**. В случае если вывод в **syslog** не используется, этот параметр, также как и **SyslogFacility**, игнорируется. В противном случае для вывода в **syslog** выбирается наименее подробный из двух указанных уровней.

Пример:

Пусть у некоторого модуля **LogLevel** = Debug, а **SyslogPriority** = Error. Тогда, если в качестве журнала для записей событий этого модуля выбрана служба **syslog,** фактически будет вестись запись на уровне подробности **Error** (будут фиксироваться только сообщения об ошибках, а отладочная информация **syslog** будет игнорироваться).

Действия с зараженными и подозрительными объектами

В настройках **Dr.Web для почтовых серверов UNIX** задаются действия, которые модули, входящие в его состав, должны совершать с объектами, которые по результатам проверки признаны вредоносными, опасными или подозрительными.

При настройке **Dr.Web MailD** и его подключаемых модулей для каждого параметра можно задать одно основное действие и до трех дополнительных. Основное действие всегда должно быть первым в списке. При настройке **Dr.Web Scanner** для соответствующих параметров может быть задано только одно действие. Для разных параметров набор допустимых действий может различаться, поэтому для каждого параметра всегда указывается перечень действий, которые могут быть в нем использованы.

При настройке параметров предусмотрено использование следующих действий:

- Cure — попытаться вылечить зараженный объект;
- Remove — удалить зараженный объект;
- Discard — отклонить письмо, не уведомляя отправителя. Письмо удаляется;
- Continue — проигнорировать угрозу и продолжить обработку письма
- Pass — пропустить письмо к получателю без дальнейшей проверки;
- Reject — отклонить письмо, уведомив отправителя. Письмо удаляется;
- Tempfail — уведомить отправителя, что письмо временно не может быть доставлено. Письмо удаляется;

Возможные дополнительные действия:

- Quarantine — отправить письмо в **Карантин**;
- Redirect [(адрес[|адрес|...])] — перенаправить письмо на другой адрес, указанный в скобках. Если адрес не указан, сообщение пересылается на адрес, определенный значением параметра **RedirectMail** в секции [MailD] конфигурационного файла **Dr.Web MailD**. Можно указать несколько адресов, разделяя их символом "|";
- Notify — выслать администратору отчет о найденных угрозах, обработка письма не прекращается;
- Score (СЧЕТ) — добавить СЧЕТ к значению счета сообщения. СЧЕТ может иметь отрицательное значение;

- `Add-header (ЗАГОЛОВОК)` — добавить к письму заголовок вида `[ИМЯ:]ЗНАЧЕНИЕ`

  где `ИМЯ` – название заголовка (по умолчанию – `X-DrWeb-MailD`), а `ЗНАЧЕНИЕ` – значение заголовка.

  Обратите внимание, что в данном действии можно использовать строки из языковых (`.lng`) файлов. Вставляемая строка указывается номером, например:

  ```
  add-header (X-Added-Header:$3)
  ```

  В данном случае будет добавлен заголовок `X-Added-Header` со значением `<value>`, взятым из строки `3="<value>"` из используемого языкового файла.

  При использовании в заголовке символа ";", а также символов "(" и ")" их необходимо *экранировать*, поскольку в противном случае конфигурация может быть интерпретирована некорректно.

  Правила экранирования символов:

  Для экранирования отдельных знаков препинания внутри заголовка, необходимо использовать 3 обратных слеша "\".

  Пример:

  ```
  EmptyFrom = continue, add-header (header:Empty header\\\; spam)
  ```

  Экранирование круглых скобок внутри выражения возможно с помощью обратного слеша "\".

  Пример:

  ```
  ProcessingErrors = tempfail,add-header(\(header:header\))
  ```

  Также возможно экранирование выражения целиком, заключив всю конструкцию в двойные кавычки: `"add-header (ЗАГОЛОВОК)"`.

  Пример:

  ```
  ProcessingErrors = tempfail,"add-header(header:(spam))"
  ```

  Для экранирования двойных кавычек внутри выражения также используется 3 обратных слеша "\".

  Примеры:

  ```
  EmptyFrom = continue,"add-header(header[X-Header]:new\\\"header\\\")"
  EmptyFrom = continue,add-header(header\[X-Header\]:new\\\"header\\\")
  ```

Доступные действия для **Dr.Web Scanner**:

- `Move` — переместить файл в каталог **Карантина**;
- `Delete` — удалить зараженный файл;
- `Rename` — переименовать файл;
- `Ignore` — пропустить файл;
- `Report` — только вывести информацию в отчет.
- `Cure` — попытаться вылечить зараженный объект.

Имена действий для указания в параметрах не чувствительны к регистру (например, значения `Report` и `report` обозначают одно и то же действие).

Установка и удаление Dr.Web для почтовых серверов UNIX

Ниже описывается процедура установки, обновления и удаления программного комплекса **Dr.Web для почтовых серверов UNIX** из универсального пакета для UNIX-систем. Для осуществления этих операций необходимы права суперпользователя (`root`). Их можно получить, введя команду **`su`** или указав префикс **`sudo`**.

Если ранее в системе продукт был установлен из пакетов других типов (например, rpm- или deb-пакетов), то желательно убедиться, что все эти пакеты удалены.

Универсальный пакет для UNIX-систем поставляется в формате EPM для использования с менеджером пакетов EPM (ESP Package Manager). Отдельные сценарии для установки и удаления компонентов, а также стандартные графические инсталляторы и деинсталляторы, входящие в состав пакетов такого типа, относятся исключительно к самому EPM-пакету, а не к упакованному в него программному комплексу в целом, и не к отдельным его модулям.

Соответственно, установка, обновление и удаление **Dr.Web для почтовых серверов UNIX** могут быть осуществлены с помощью:

- графических инсталлятора и деинсталлятора;
- консольных инсталляторов и деинсталляторов.

При установке поддерживается работа с зависимостями, т.е. если для установки какого-либо из компонентов программного комплекса должен быть предварительно установлен другой компонент (например, для установки компонента **`drweb-daemon`** предварительно должны быть установлены компоненты **`drweb-common`** и **`drweb-bases`**), то он будет установлен автоматически.

Необходимо отметить, что если вы устанавливаете программный комплекс **Dr.Web для почтовых серверов UNIX** на компьютер, куда ранее из аналогичного универсального EPM-пакета был установлен какой-либо другой продукт **Dr.Web**, то при каждом использовании графического деинсталлятора вам будет предложено удалить абсолютно все модули **Dr.Web**, включая установленные ранее в составе других продуктов.

Крайне внимательно подходите к удалению компонентов, чтобы по ошибке не удалить те из них, которые вы планируете использовать в дальнейшем.

Установка универсального пакета для UNIX систем

Дистрибутив программного комплекса **Dr.Web для почтовых серверов UNIX** распространяется в виде самораспаковывающегося архива `drweb-mail-[название-продукта]_[номер версии]~[название ОС].run`.

В общем случае в архиве содержатся следующие пакеты:

- `drweb-common`: пакет содержит основной конфигурационный файл `drweb32.ini`, библиотеки, документацию и структуру каталогов. В процессе установки данного компонента будут созданы пользователь `drweb` и группа `drweb`;
- `drweb-bases`: пакет содержит Антивирусное ядро **Dr.Web Engine** и вирусные базы. Для установки требует пакет `drweb-common`;
- `drweb-libs`: пакет содержит библиотеки, общие для всех компонентов продукта;
- `drweb-epm6.0.2-libs`: пакет содержит библиотеки для графических инсталлятора и деинсталлятора. Для установки требует пакет `drweb-libs`;

- drweb-epm6.0.2-uninst: пакет содержит файлы графического деинсталлятора. Для установки требует пакет drweb-epm6.0.2-libs;
- drweb-boost147: пакет содержит библиотеки, использующиеся **Dr.Web Agent** и **Dr.Web Monitor** совместно. Для установки требует пакет drweb-libs;
- drweb-updater: пакет содержит модуль обновления Антивирусного ядра **Dr.Web Engine** и вирусных баз **Dr.Web Updater**. Для установки требует пакеты drweb-common и drweb-libs;
- drweb-agent: пакет содержит исполняемые файлы **Dr.Web Agent** и документацию к нему. Для установки требует пакеты drweb-boost147 и drweb-common;
- drweb-agent-es: пакет содержит файлы для работы **Dr.Web Agent** в режиме централизованной защиты с сервером **Dr.Web ESS** версии 6. Для установки требует пакеты drweb-agent, drweb-updater и drweb-scanner;
- drweb-agent10: пакет содержит исполняемые файлы и документацию обновленной версии **Dr.Web Agent** (предназначен для работы с сервером **Dr.Web ESS** версии 10).
- drweb-agent10-es: пакет содержит файлы для работы обновленной версии **Dr.Web Agent** с сервером **Dr.Web ESS** версии 10 в режиме централизованной защиты.
- drweb-monitor: пакет содержит исполняемые файлы **Dr.Web Monitor** и документацию к нему. Для установки требует пакеты drweb-boost147, drweb-agent и drweb-common;
- drweb-daemon: пакет содержит исполняемые файлы **Dr.Web Daemon** и документацию к нему. Для установки требует пакеты drweb-bases и drweb-libs;
- drweb-scanner: пакет содержит исполняемые файлы консольного сканера **Dr.Web Scanner** и документацию к нему. Для установки требует пакеты drweb-bases и drweb-libs;
- drweb-maild: пакет содержит исполняемые файлы **Dr.Web MailD** и документацию к нему. Для установки требует пакет drweb-maild-common;
- drweb-maild-common: пакет содержит библиотеки для **Dr.Web Agent**, **Dr.Web Monitor** и **Dr.Web MailD**. Для установки требует пакеты drweb-common, drweb-gperftools0, drweb-agent и drweb-monitor;
- drweb-maild-web: пакет содержит веб-интерфейс **Dr.Web консоль для почтовых серверов UNIX**;
- drweb-maild-plugin-drweb: пакет содержит библиотеку подключаемого модуля **Drweb**, его конфигурационный файл, документацию и скрипт конфигурации. Для установки требует пакет drweb-maild;
- drweb-maild-plugin-headersfilter: пакет содержит библиотеку подключаемого модуля **Headersfilter**, его конфигурационный файл, документацию и скрипт конфигурации. Для установки требует пакет drweb-maild;
- drweb-maild-plugin-modifier: пакет содержит библиотеку подключаемого модуля **Modifier**, его конфигурационный файл, документацию и скрипт конфигурации. Для установки требует пакет drweb-maild;
- drweb-maild-plugin-vaderetro: пакет содержит конфигурационный файл подключаемого модуля **Vaderetro**, документацию и скрипт конфигурации. Для установки требует пакеты drweb-maild и drweb-libvaderetro;
- drweb-libvaderetro: пакет содержит библиотеку подключаемого модуля **Vaderetro**;
- drweb-maild-smtp: пакет содержит исполняемые файлы модулей **Sender** и **Receiver** для обеспечения функционирования системы в качестве прокси-сервера для протоколов SMTP и LMTP, конфигурационный файл **Dr.Web MailD** с соответствующими настройками, документацию, скрипт для конфигурации компонента **Dr.Web Monitor**. Для установки требует пакет drweb-maild;
- drweb-maild-cgp: пакет содержит исполняемые файлы модулей **Sender** и **Receiver** для обеспечения взаимодействия с почтовой системой **Communigate Pro**, конфигурационный файл **Dr.Web MailD** с настройками для конкретной системы, документацию и скрипт для конфигурации **Communigate Pro** под **Dr.Web MailD**. Для установки требует пакет drweb-

maild;

- drweb-maild-courier: пакет содержит исполняемые файлы модулей **Sender** и **Receiver** для обеспечения взаимодействия с почтовой системой **Courier**, конфигурационный файл **Dr.Web MailD** с настройками для конкретной системы, документацию и скрипт для конфигурации **Courier** под **Dr.Web MailD**. Для установки требует пакет drweb-maild;
- drweb-maild-exim: пакет содержит исполняемые файлы модулей **Sender** и **Receiver** для обеспечения взаимодействия с почтовой системой **Exim**, конфигурационный файл **Dr.Web MailD** с настройками для конкретной системы, документацию и скрипт для конфигурации **Exim** под **Dr.Web MailD**. Для установки требует пакет drweb-maild;
- drweb-maild-postfix: пакет содержит исполняемые файлы модулей **Sender** и **Receiver** для обеспечения взаимодействия с почтовой системой **Postfix**, конфигурационный файл **Dr.Web MailD** с настройками для конкретной системы, документацию и скрипт для конфигурации **Postfix** под **Dr.Web MailD**. Для установки требует пакет drweb-maild;
- drweb-maild-qmail: пакет содержит исполняемые файлы модулей **Sender** и **Receiver** для обеспечения взаимодействия с почтовой системой **Qmail**, конфигурационный файл **Dr.Web MailD** с настройками для конкретной системы, документацию и скрипт для конфигурации **Qmail** под **Dr.Web MailD**. Для установки требует пакет drweb-maild;
- drweb-maild-sendmail: пакет содержит исполняемые файлы модулей **Sender** и **Receiver** для обеспечения взаимодействия с почтовой системой **Sendmail**, конфигурационный файл **Dr.Web MailD** с настройками для конкретной системы, документацию и скрипт для конфигурации **Sendmail** под **Dr.Web MailD**. Для установки требует пакет drweb-maild;
- drweb-maild-zmailer: пакет содержит исполняемые файлы модулей **Sender** и **Receiver** для обеспечения взаимодействия с почтовой системой **ZMailer**, конфигурационный файл **Dr.Web MailD** с настройками для конкретной системы, документацию и скрипт для конфигурации **ZMailer** под **Dr.Web MailD**. Для установки требует пакет drweb-maild;
- drweb-gperftools0: пакет содержит библиотеку **Google Performance Tools**, используемую **Dr.Web MailD**. Для установки требует пакет drweb-libs;
- drweb-mail-servers-gateways-doc: пакет содержит документацию к **Dr.Web для почтовых серверов UNIX**.

В версии для 64-битных систем в архив включены два пакета: drweb-libs и drweb-libs32, в которых содержатся библиотеки для 64-битных и 32-битных компонентов соответственно.

Для автоматической установки компонентов программного комплекса **Dr.Web для почтовых серверов UNIX** разрешите исполнение архива, например, командой:

```
# chmod +x drweb-mail-[название-продукта]_[номер версии]~[название ОС].run
```

и затем запустите его на исполнение командой:

```
# ./drweb-mail-[название-продукта]_[номер версии]~[название ОС].run
```

или воспользуйтесь стандартным файловым менеджером вашей графической оболочки как для изменения свойств файла, так и для его запуска.

При этом будет создан каталог drweb-mail-[название-продукта]_[номер версии]~[название ОС] с набором файлов внутри, и автоматически запустится графический инсталлятор. Если запуск был осуществлен не с правами администратора, то инсталлятор сам попытается получить нужные права.

Если запустить графический инсталлятор не удалось, то автоматически запустится интерактивный консольный инсталлятор.

Если необходимо только распаковать архив, не запуская при этом графический инсталлятор, следует воспользоваться параметром командной строки `--noexec`:

```
# ./drweb-mail-[название-продукта]_[номер версии]~[название ОС].run --noexec
```

Для продолжения установки с помощью графического инсталлятора запустите его командой:

```
# drweb-mail-[название-продукта]_[номер версии]~[название ОС]/install.sh
```

Для установки с использованием консольного инсталлятора потребуется выполнить команду:

```
# drweb-mail-[название-продукта]_[номер версии]~[название ОС]/setup.sh
```

При установке любым из описанных ниже способов происходит следующее:

- в каталог `%etc_dir/software/conf/` записываются оригиналы дистрибутивных конфигурационных файлов с названиями в формате `[имя_конфигурационного_файла].N`;
- конфигурационные файлы устанавливаются в соответствующие каталоги системы;
- устанавливаются остальные файлы, причем если файл с таким именем уже имеется (например, остался после неаккуратного удаления пакетов других типов), то на его место записывается новый файл, а копия старого сохраняется как `[имя_файла].O`. Если в каталоге уже имеется файл с таким именем (`[имя_файла].O`), то он будет удален, а новый файл будет записан на его место;
- Если в соответствующем окне графического инсталлятора установлен флаг **Run interactive postinstall script**, то после установки компонентов будет запущен инсталляционный скрипт для настройки базовой функциональности **Dr.Web для почтовых серверов UNIX**.

Пожалуйста, обратите внимание, что если ваш дистрибутив **Linux** оснащен подсистемой безопасности **SELinux,** то возможно возникновение ситуации, когда работа инсталлятора будет прервана подсистемой безопасности. В этом случае вам необходимо временно перевести **SELinux** в разрешающий `(Permissive)` режим, для чего выполните команду

```
# setenforce 0
```

и перезапустите инсталлятор.

Также в этом случае вам по окончании установки нужно будет выполнить настройку политик безопасности **SELinux** для того, чтобы в дальнейшем антивирусные компоненты работали корректно.

После успешного завершения установки run-файл и каталог `drweb-mail-[название-продукта]_[номер версии]~[название ОС]` можно удалить.

Пользовательский интерфейс графического инсталлятора

1. При запуске графического инсталлятора командой:

   ```
   # drweb-mail-[название-продукта]_[номер версии]~[название ОС]/install.sh
   ```

   открывается окно программы установки.

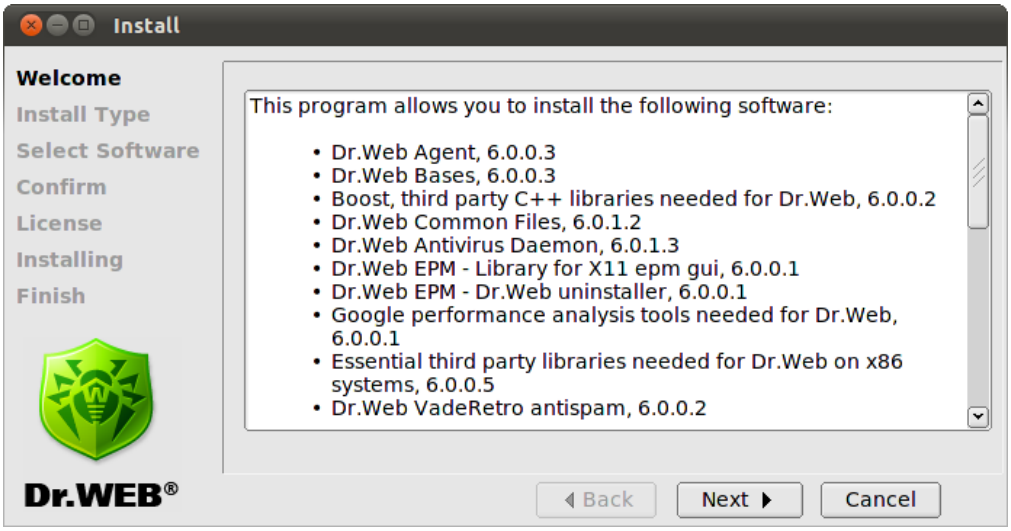


Рис. 2. Окно начала установки программы

Навигация осуществляется с помощью кнопок **Back** и **Next**. Установку можно прервать в любой момент, нажав кнопку **Cancel**.

2. В следующем окне **Install Type** вы можете выбрать тип установки: для почтового шлюза **Dr.Web for Mail Gateways** или для конкретной почтовой системы **Dr.Web for MTA (Full installation)** со всеми компонентами по умолчанию или пользовательский.

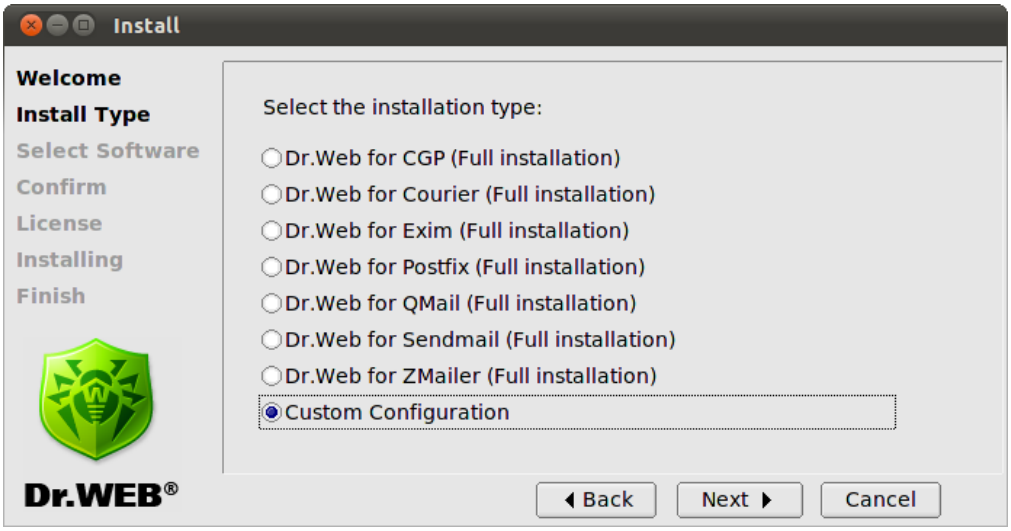


Рис. 3. Окно Install Type для Dr.Web для почтовых серверов UNIX

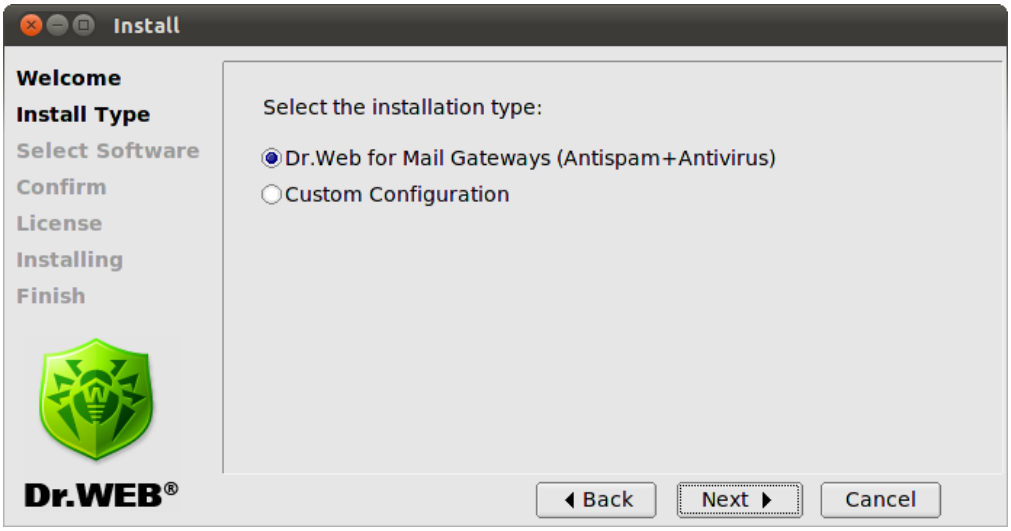


Рис. 4. Окно Install Type для Dr.Web для почтовых шлюзов UNIX

Если вы выбрали пункт **Custom Configuration**, то следующим откроется окно **Select Software**, в котором вы сможете указать необходимые вам компоненты.

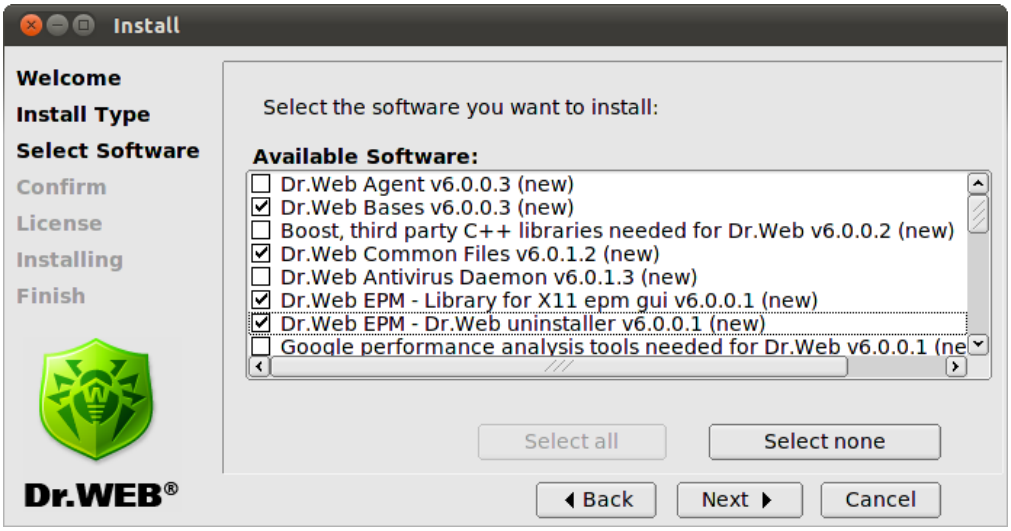


Рис. 5. Окно выбора компонентов для установки

Если для установки выбранного вами компонента должен быть предварительно установлен другой компонент, то соответствующая зависимость будет отмечена автоматически. Таким образом, если вы установите флаг напротив **Dr.Web Antivirus Daemon**, то флаги автоматически появятся напротив пунктов **Dr.Web Bases** и **Dr.Web Common Files**.

При установке **Dr.Web для почтовых серверов UNIX** пакеты для различных почтовых систем (`drweb-maild-smtp` и разнообразные `drweb-maild-MTA`) будут конфликтовать друг с другом. Например, при попытке отметить для установки одновременно два пакета **Dr.Web Mail Daemon – Exim Connector** и **Dr.Web Mail Daemon – Postfix Connector** вы получите сообщение об ошибке и предложение выбрать только один пакет из двух.

Нажатие на кнопку **Select all** выберет все компоненты, нажатие на кнопку **Select none** снимет все установленные флажки.

3. В окне **Confirm** вы увидите все выбранные вами компоненты и сможете принять окончательное решение.

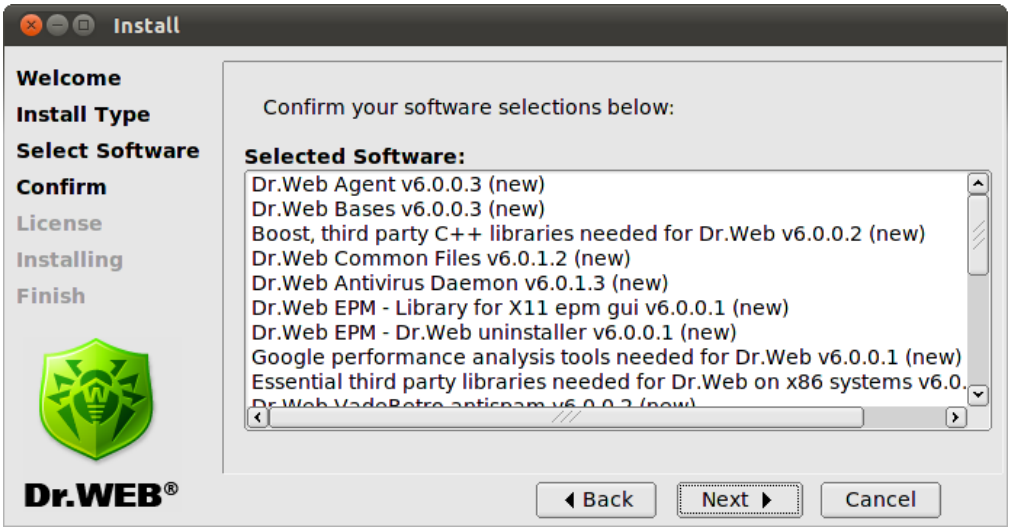


Рис. 6. Окно подтверждения установки компонентов

4. Ознакомьтесь с текстом **Лицензионного Договора** и подтвердите свое согласие с ним, чтобы продолжить установку. С помощью меню **Select language** вы можете выбрать язык (русский или английский), на котором будет изложен текст **Лицензионного Договора**.

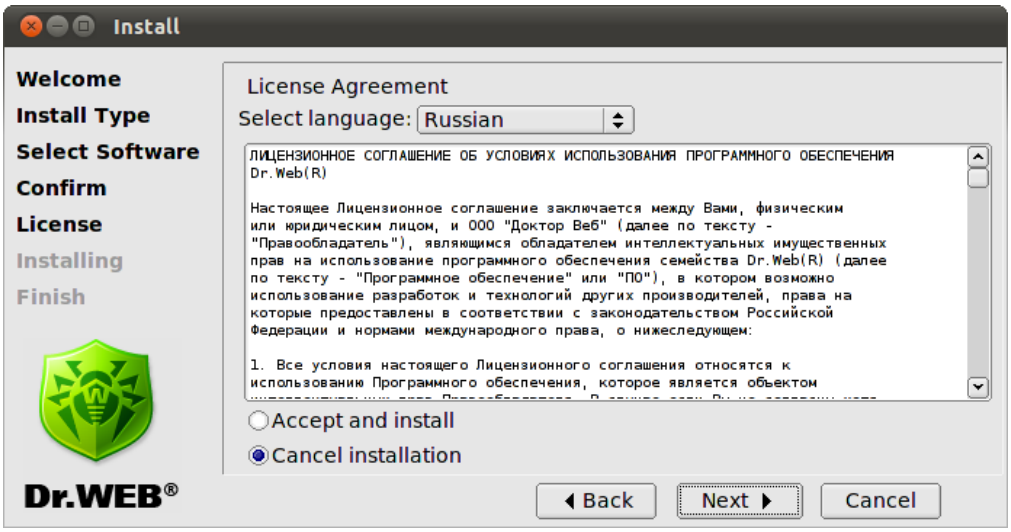


Рис. 7. Окно ознакомления с лицензионным соглашением

5. В следующем окне **Installing** выводится отчет о процессе установки в режиме реального времени.

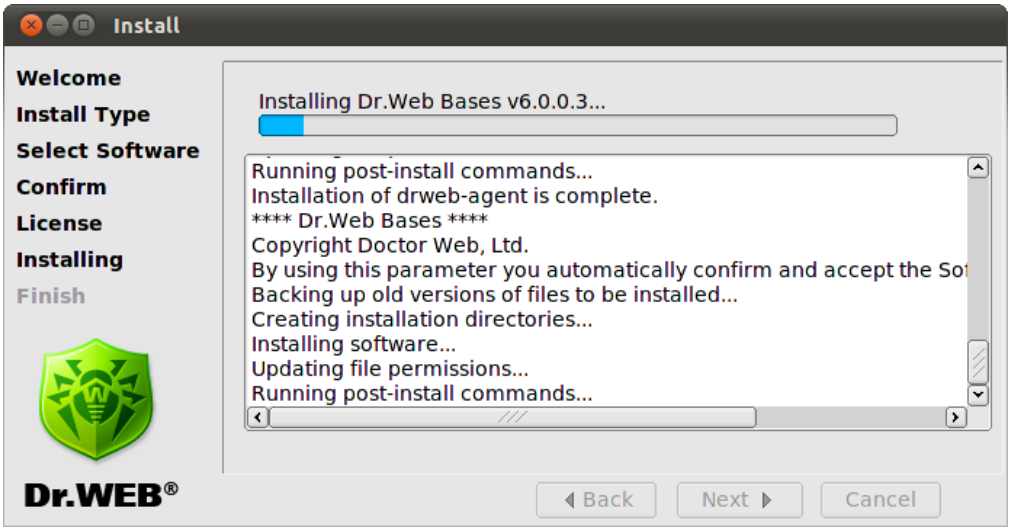


Рис. 8. Окно установки компонентов программы

Одновременно данный отчет копируется в файл `install.log`, расположенный в каталоге `drweb-mail-[название-продукта]_[номер версии]~[название ОС]`. Если установлен флаг **Run interactive postinstall script**, то после установки компонентов будет запущен инсталляционный скрипт для настройки базовой функциональности **Dr.Web для почтовых серверов UNIX**.

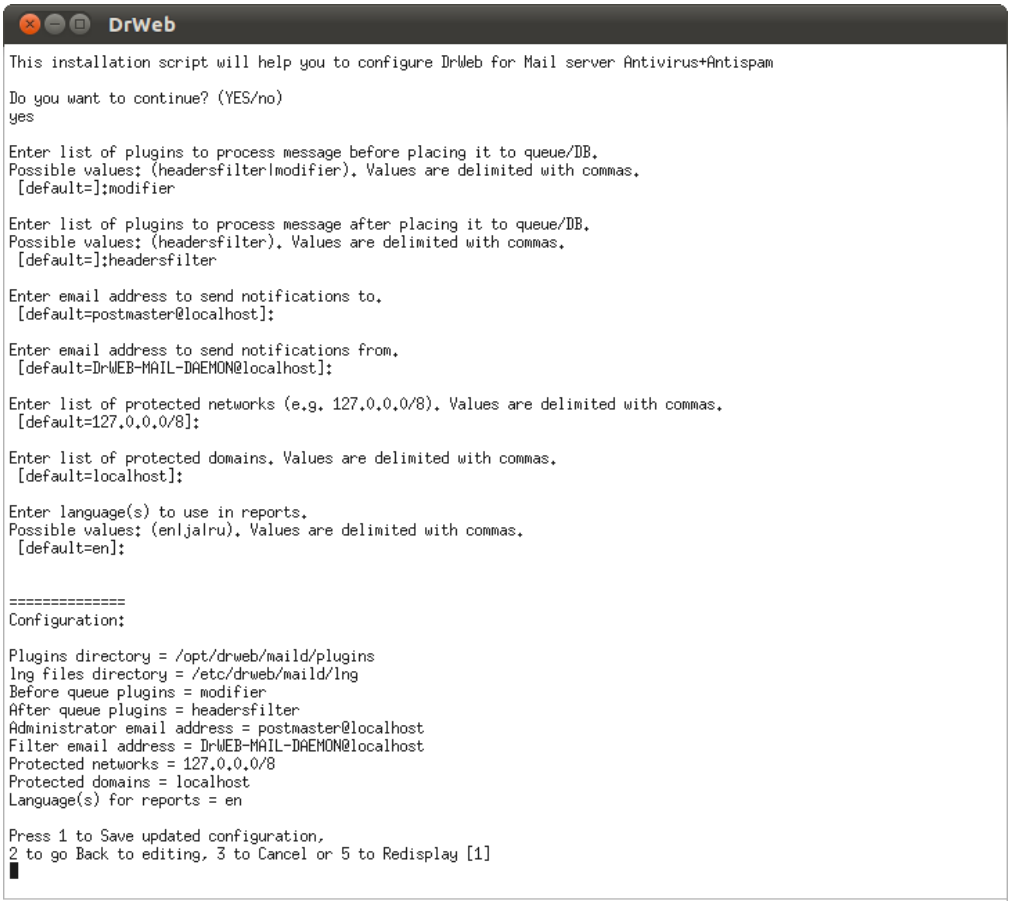


Рис. 9. Интерактивный установочный скрипт

Скрипт предложит указать путь к лицензионному ключевому файлу, установить порядок

работы подключаемых модулей, указать список защищаемых сетей и доменов и автоматически подключить необходимые для работы сервисы (**Dr.Web Daemon**, **Dr.Web Agent**, **Dr.Web Monitor**).

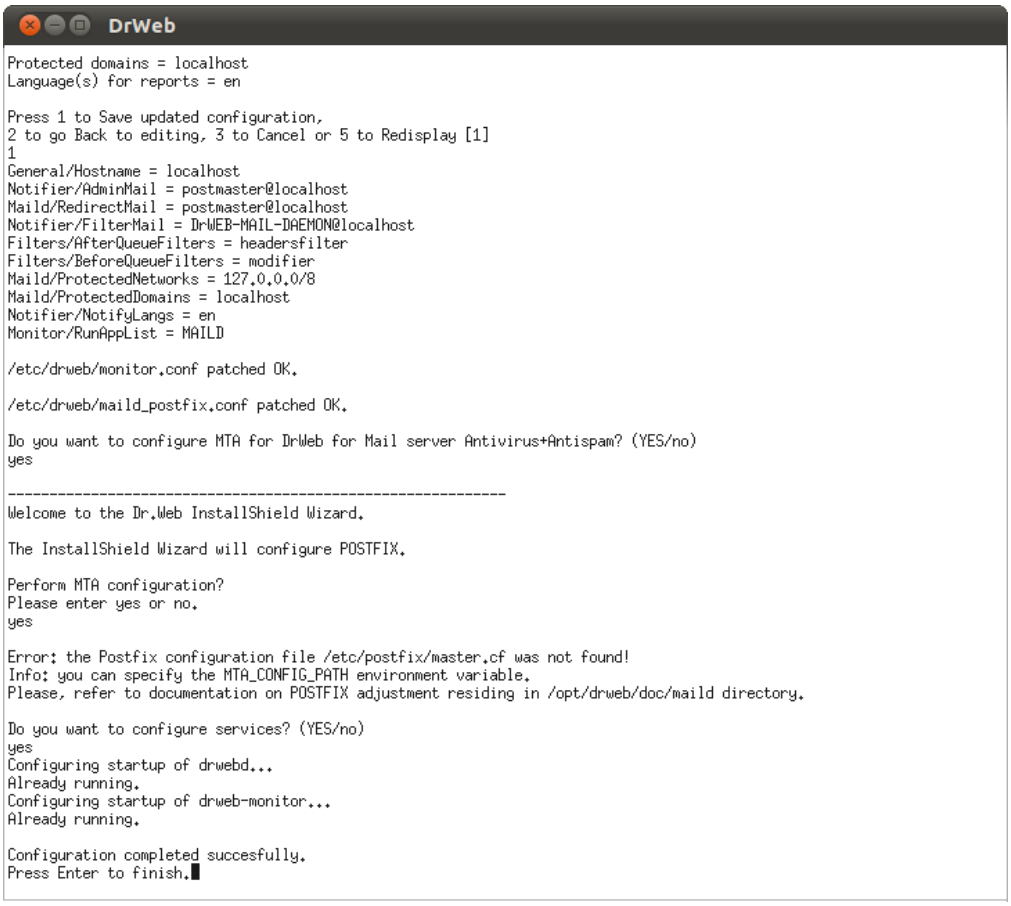


Рис. 10. Настройка MTA и автоматического запуска сервисов

6. В последнем окне **Finish** Нажав на кнопку **Close**, вы закроете окно программы установки компонентов.

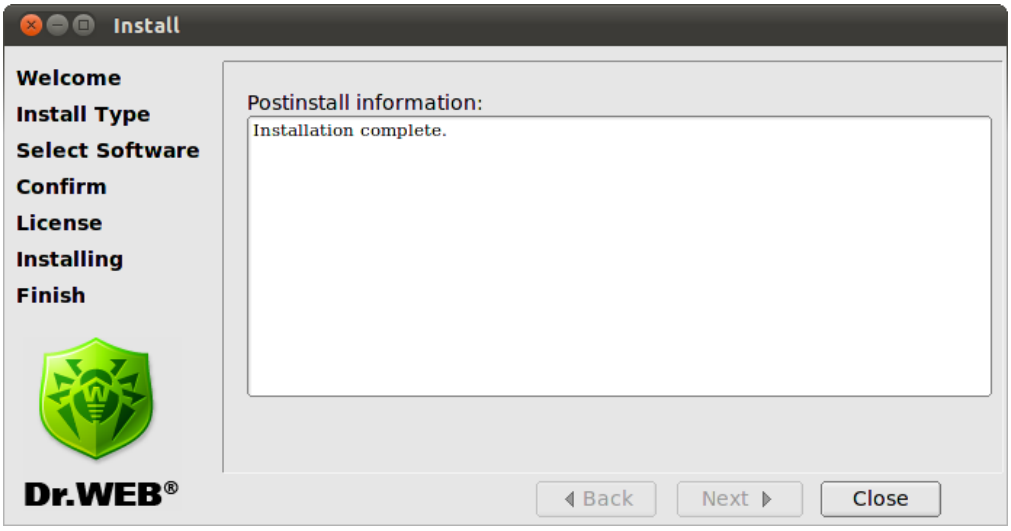


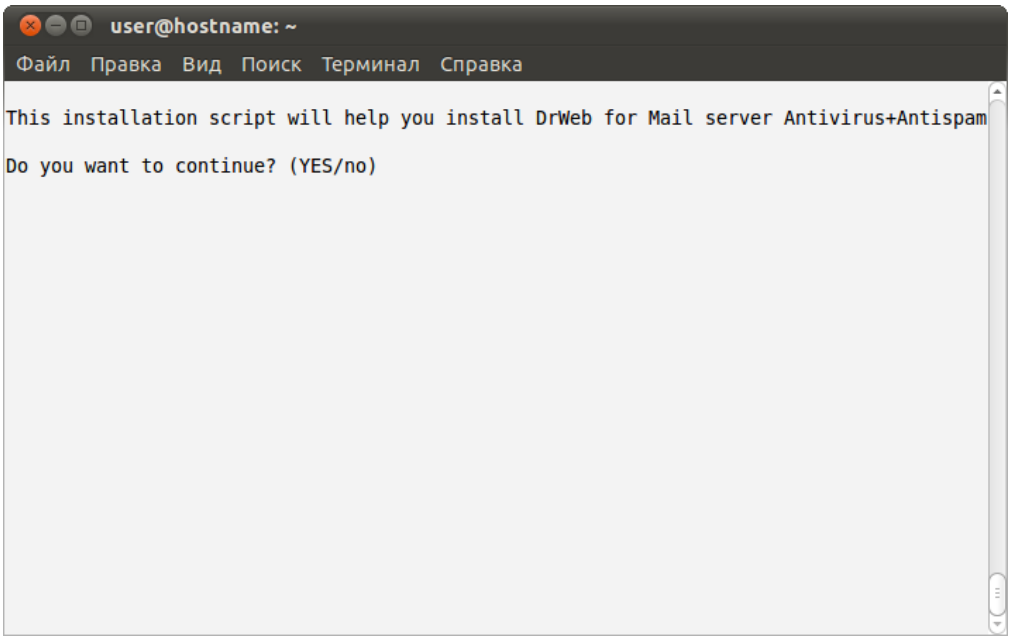
Рис. 11. Окно завершения установки программы

Использование консольного инсталлятора

Консольный инсталлятор запускается автоматически в том случае, если не удалось запустить графический инсталлятор. Если консольный инсталлятор не был запущен автоматически (как правило, это происходит при невозможности повысить права), то можно попробовать запустить его с привилегиями пользователя `root`, выполнив команду (для получения прав `root` воспользуйтесь командой **`su`** или **`sudo`**):

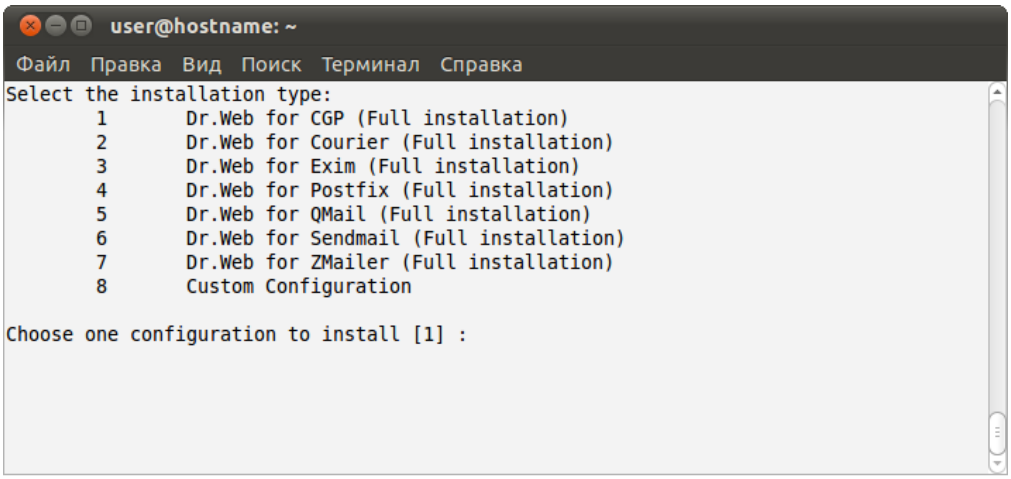
```
# drweb-mail-[название-продукта]_[номер версии]~[название ОС]/setup.sh
```

Откроется диалоговое окно консольного инсталлятора.

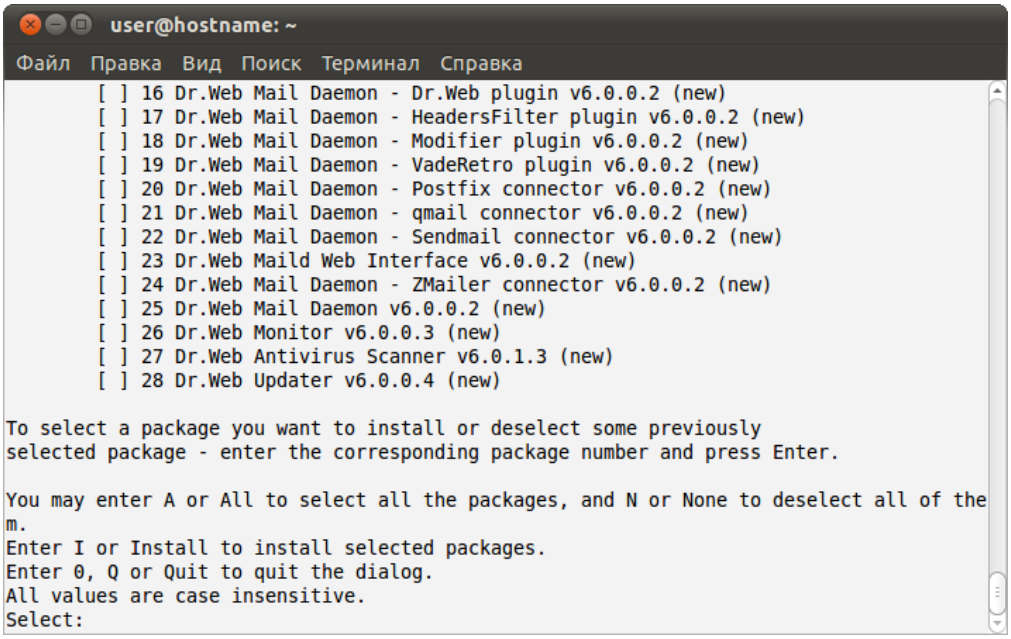


Если вы хотите установить **Dr.Web для почтовых серверов UNIX**, укажите **Y** или **Yes** в строке ввода (значения регистронезависимы) и нажмите клавишу ENTER. В противном случае введите **N** или **No**.

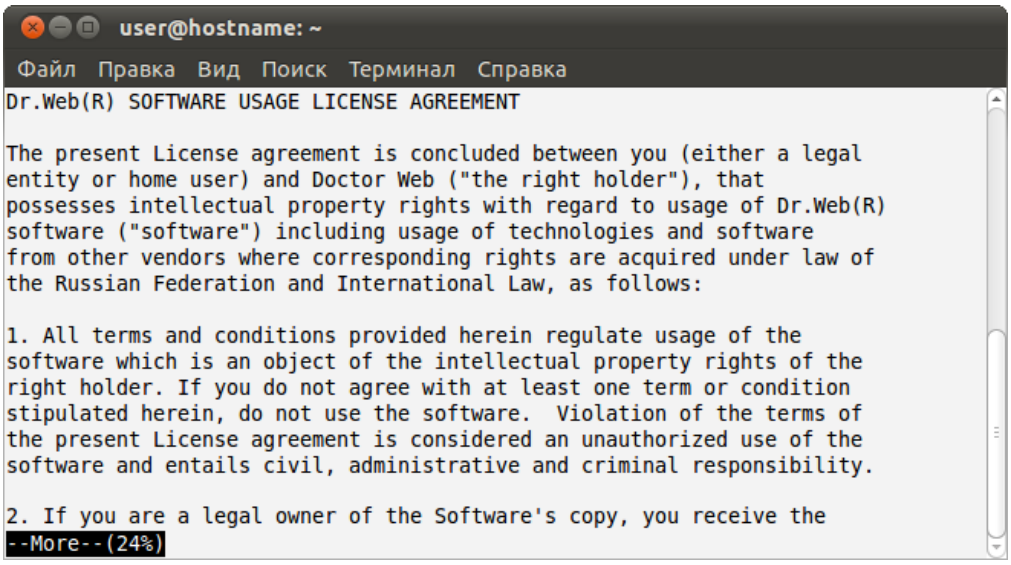
Затем вам будет предложено выбрать тип установки. Укажите номер соответствующего пункта в меню и нажмите ENTER.



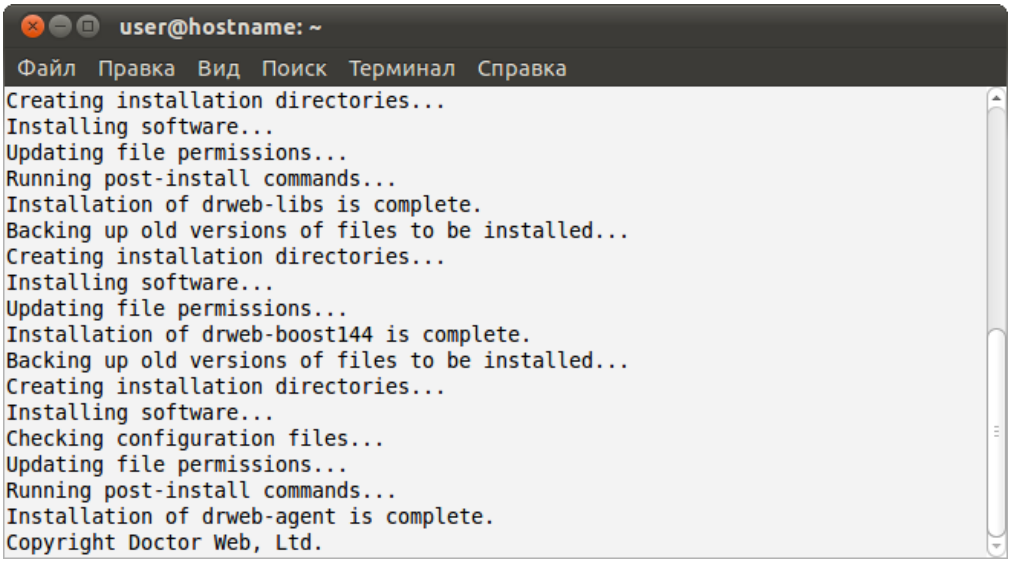
Если вы выбрали пункт **Custom Configuration**, то на следующем этапе вам будет предложено указать необходимые компоненты для установки. Укажите номер соответствующего компонента в меню и нажмите ENTER.



На следующем этапе вам будет предложено ознакомиться с текстом **Лицензионного Договора**. Для пролистывания текста договора нажимайте клавишу ПРОБЕЛ.



Для продолжения установки вы должны будете принять **Лицензионный Договор**, указав **Y** или **Yes** в строке ввода и нажав ENTER. В противном случае установка будет прекращена. После того, как вы примете **Лицензионный Договор**, будет запущен процесс установки. Отчет о результатах прохождения каждого из этапов процесса будет выводиться на консоль в режиме реального времени.



После установки компонентов будет запущен инсталляционный скрипт для настройки базовой функциональности **Dr.Web для почтовых серверов UNIX**. Скрипт предложит указать путь к лицензионному ключевому файлу и автоматически подключить необходимые для работы сервисы (**Dr.Web Daemon**, **Dr.Web Agent**, **Dr.Web Monitor**). Дополнительно Вам будет предложено установить порядок работы подключаемых модулей, указать список защищаемых сетей и доменов.

Удаление универсального пакета для UNIX систем

Для удаления с помощью графического деинсталлятора, запустите его командой:

```
# %bin_dir/remove.sh
```

Если запуск был осуществлен не с правами администратора, то деинсталлятор сам попытается получить нужные права.

Если запустить графический деинсталлятор не удалось, то автоматически запустится интерактивный консольный деинсталлятор.

После деинсталляции продукта можно удалить средствами ОС пользователя `drweb` и группу `drweb`.

При удалении любым из вышеописанных способов происходит следующее:

- из каталога `%etc_dir/software/conf/` удаляются все дистрибутивные конфигурационные файлы;
- если рабочие конфигурационные файлы не были изменены пользователем, то они тоже удаляются. Если пользователь вносил в них изменения, они остаются в неприкосновенности;
- удаляются остальные файлы, причем если при установке была создана копия какого-либо старого файла в виде `[имя_файла].O`, то этот файл восстанавливается в прежнем виде;
- лицензионные ключевые файлы и файлы отчетов различных компонентов программного комплекса в соответствующих каталогах сохраняются.

Пользовательский интерфейс графического деинсталлятора

1. При запуске графического деинсталлятора командой:

```
# %bin_dir/remove.sh
```

открывается окно программы удаления компонентов.

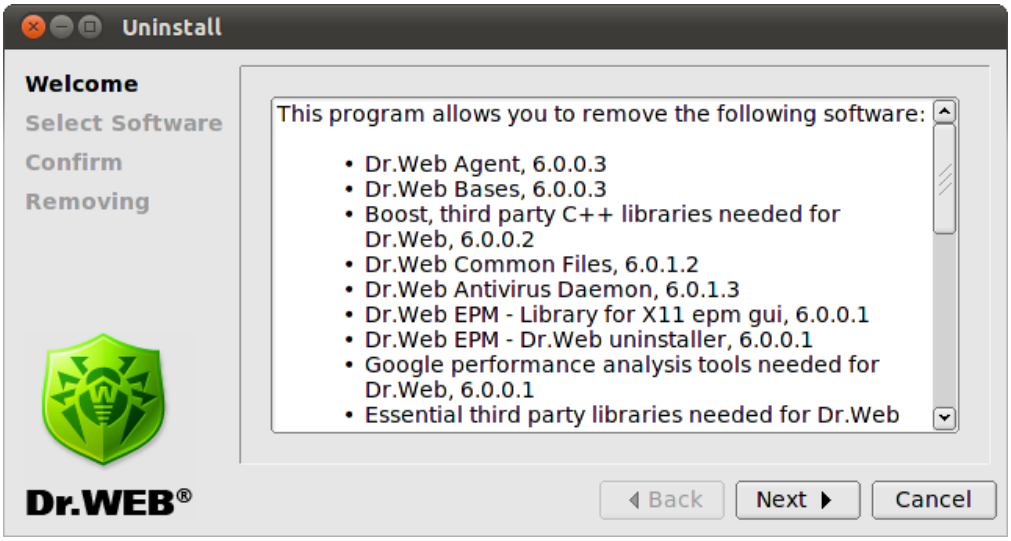


Рис. 12. Окно начала удаления программы

Навигация осуществляется с помощью кнопок **Back** и **Next**. Выйти из программы можно в любой момент, нажав кнопку **Cancel**.

2. В следующем окне **Select Software** вы можете выбрать компоненты, которые хотите удалить. Флаги для соответствующих зависимостей будут проставлены автоматически.

В случае, если ранее на этом компьютере из EPM-пакета был установлен какой-либо другой продукт **Dr.Web**, то в список компонентов для удаления войдут и его модули тоже. Поэтому необходимо быть крайне внимательным при выборе, чтобы случайно не удалить те компоненты, которые планируется использовать в дальнейшем.

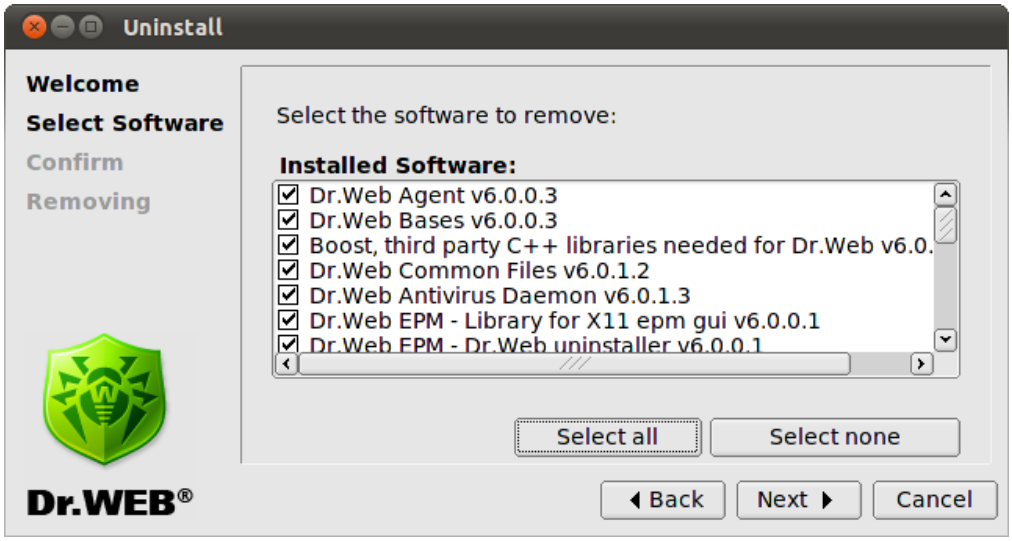


Рис. 13. Окно выбора компонентов для удаления

Нажав на кнопку **Select all**, вы сможете отметить сразу все компоненты. Нажатие на кнопку **Select none** удалит все проставленные флаги.

3. В следующем окне **Confirm** вы увидите все выбранные вами компоненты и сможете принять окончательное решение об их удалении.

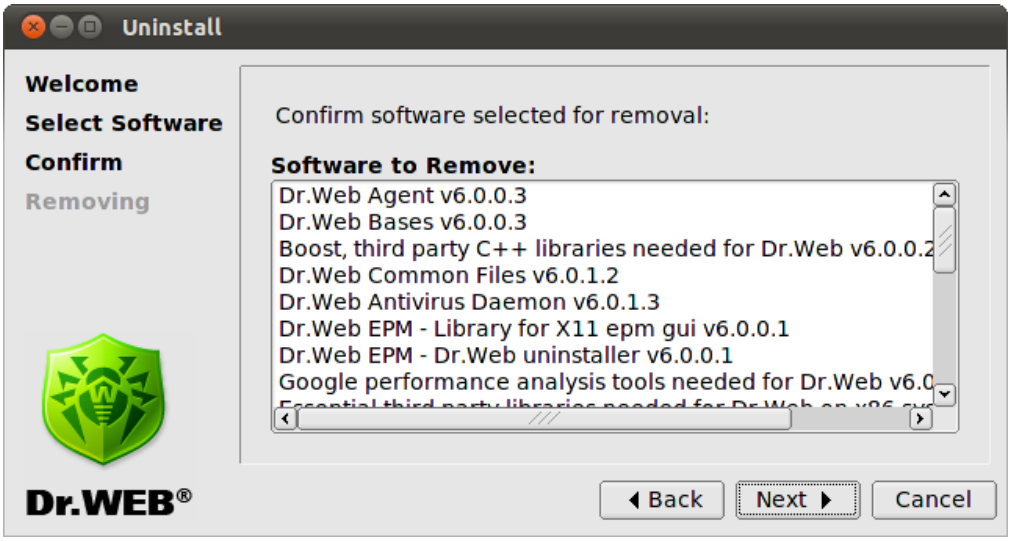


Рис. 14. Окно подтверждения удаления компонентов

4. В последнем окне **Removal** выводится отчет о процессе удаления компонентов программного комплекса в режиме реального времени.

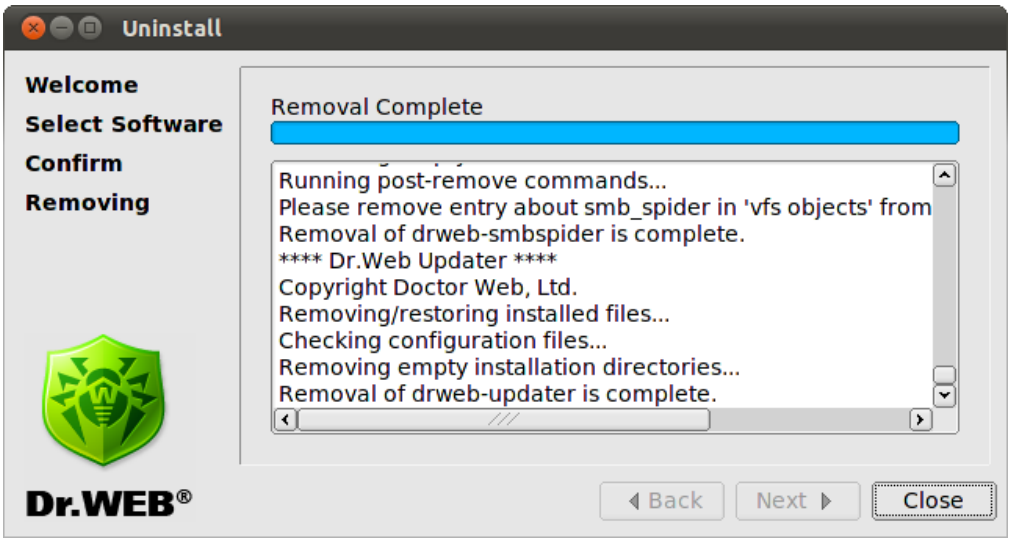


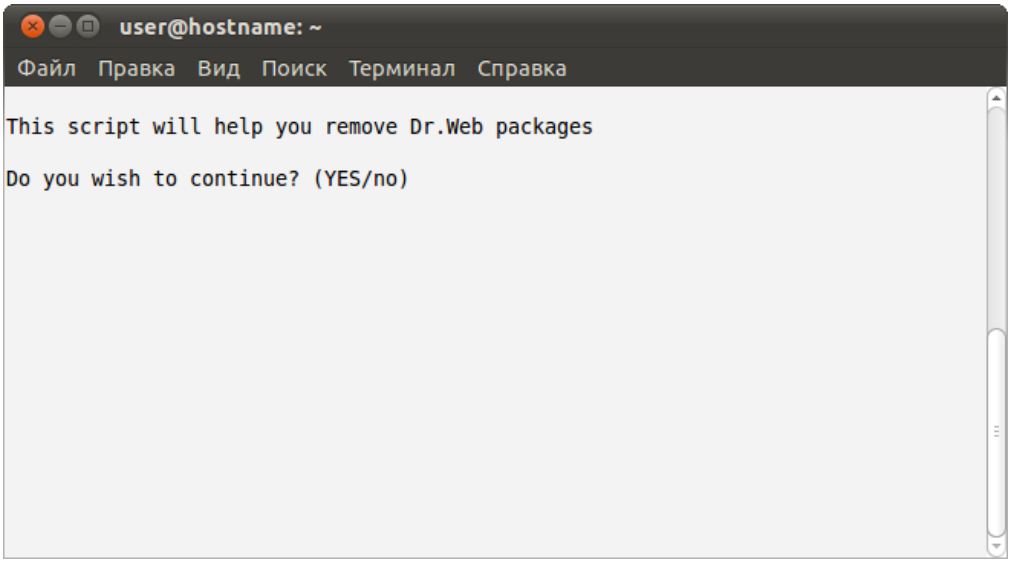
Рис. 15. Окно удаления компонентов программы

5. Нажав на кнопку **Close**, вы закроете окно программы удаления компонентов.

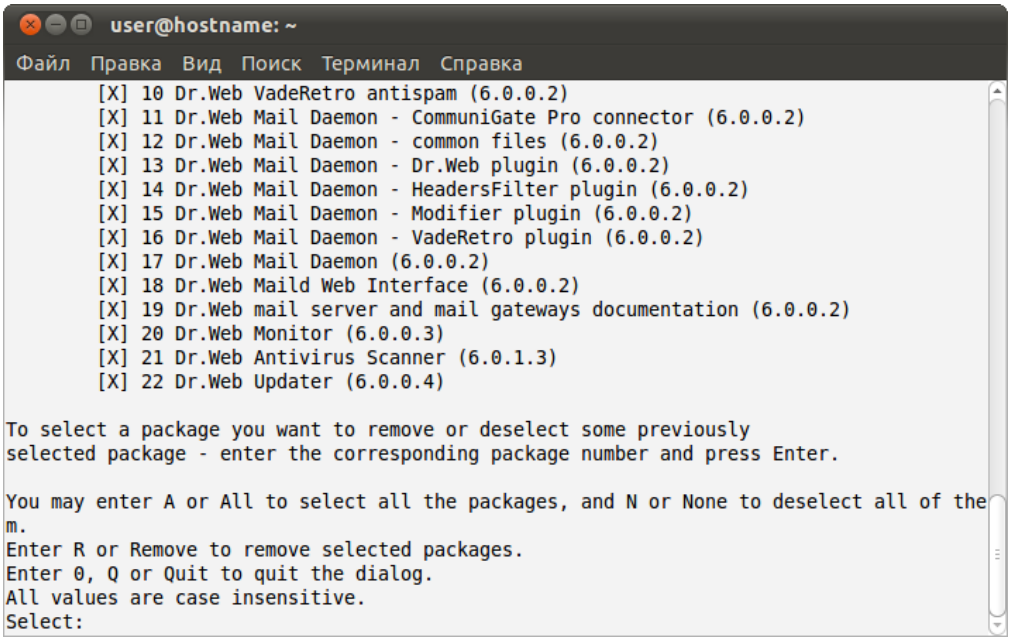
Использование консольного деинсталлятора

Консольный деинсталлятор запускается автоматически в том случае, если не удалось запустить графический деинсталлятор.

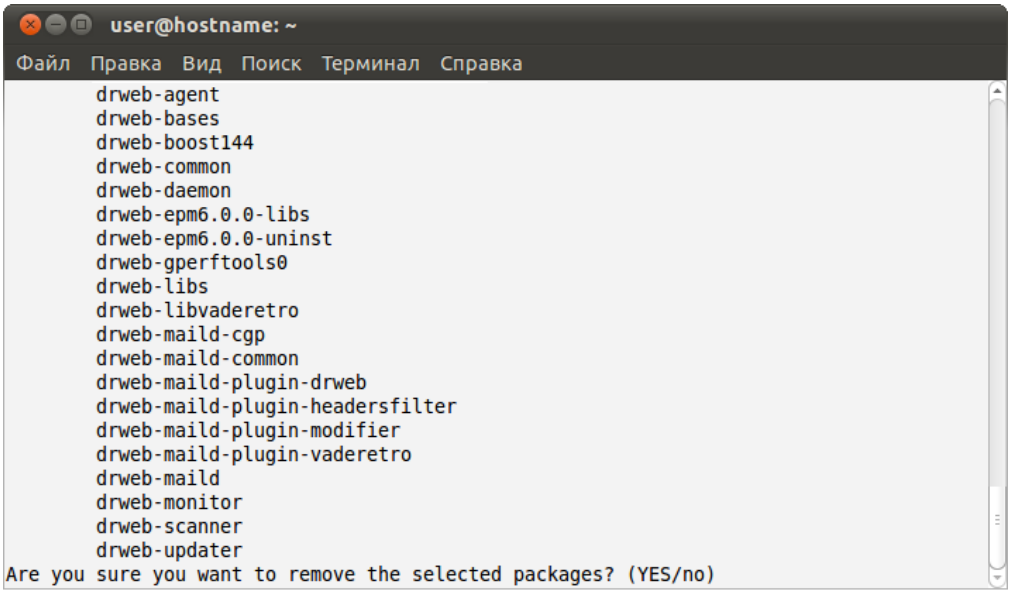
Откроется диалоговое окно консольного деинсталлятора.



Вам будет предложено выбрать из списка компонентов те, которые вы желаете удалить (следуйте инструкциям на экране).



Для запуска процедуры удаления компонентов вы должны будете подтвердить сделанный выбор, указав **Y** или **Yes** в строке ввода (значения регистронезависимы) и нажав клавишу ENTER.



Отчет о результатах прохождения каждого из этапов процесса удаления компонентов выводится на консоль в режиме реального времени.

Установка из нативных пакетов

Вы можете установить **Dr.Web для почтовых серверов UNIX** из нативных пакетов для распространенных дистрибутивов **Linux** или **FreeBSD**.

Пакеты находятся в официальном репозитории **Dr.Web** http://officeshield.drweb.com/drweb/. После подключения репозитория к менеджеру пакетов вашей системы, вы можете устанавливать пакеты как любую другую программу из репозитория. Необходимые зависимости будут разрешены автоматически.

После установки пакетов через репозиторий пост-инсталляционный скрипт для автоматической установки лицензионного ключевого файла не будет запущен. Ключевой файл необходимо вручную скопировать в каталог `%bin_dir`.

После обновления через репозиторий все сервисы **Dr.Web** необходимо перезапустить, чтобы обновления вступили в силу.

Ниже приведены инструкции для подключения репозитория **Dr.Web** к поддерживаемым менеджерам пакетов и установки **Dr.Web для почтовых серверов UNIX** с помощью консоли.

В зависимости от необходимой комплектации, в качестве `<имя пакета>` следует указать один из следующих пакетов:

- `drweb-mail-gateways-as` – **Антиспам Dr.Web для почтовых шлюзов UNIX**;
- `drweb-mail-gateways-av` – **Антивирус Dr.Web для почтовых шлюзов UNIX**;
- `drweb-mail-gateways-av-as` – **Антивирус и Антиспам Dr.Web для почтовых шлюзов UNIX**;
- `drweb-courier-as` – **Антиспам Dr.Web для почтовых серверов Courier**;
- `drweb-courier-av` – **Антивирус Dr.Web для почтовых серверов Courier**;
- `drweb-courier-av-as` – **Антивирус и Антиспам Dr.Web для почтовых серверов Courier**;
- `drweb-postfix-as` – **Антиспам Dr.Web для почтовых серверов Postfix**;
- `drweb-postfix-av` – **Антивирус Dr.Web для почтовых серверов Postfix**;

- `drweb-postfix-av-as` – **Антивирус и Антиспам Dr.Web для почтовых серверов Postfix**;
- `drweb-qmail-as` – **Антиспам Dr.Web для почтовых серверов qmail**;
- `drweb-qmail-av` – **Антивирус Dr.Web для почтовых серверов qmail**;
- `drweb-qmail-av-as` – **Антивирус и Антиспам Dr.Web для почтовых серверов qmail**;
- `drweb-sendmail-as` – **Антиспам Dr.Web для почтовых серверов Sendmail**;
- `drweb-sendmail-av` – **Антивирус Dr.Web для почтовых серверов Sendmail**;
- `drweb-sendmail-av-as` – **Антивирус и Антиспам Dr.Web для почтовых серверов Sendmail**;
- `drweb-cgp-as` – **Антиспам Dr.Web для почтовых серверов CommuniGate Pro**;
- `drweb-cgp-av` – **Антивирус Dr.Web для почтовых серверов CommuniGate Pro**;
- `drweb-cgp-av-as` – **Антивирус и Антиспам Dr.Web для почтовых серверов CommuniGate Pro**;
- `drweb-exim-as` – **Антиспам Dr.Web для почтовых серверов Exim**;
- `drweb-exim-av` – **Антивирус Dr.Web для почтовых серверов Exim**;
- `drweb-exim-av-as` – **Антивирус и Антиспам Dr.Web для почтовых серверов Exim** ;
- `drweb-zmailer-as` – **Антиспам Dr.Web для почтовых серверов ZMailer**;
- `drweb-zmailer-av` – **Антивирус Dr.Web для почтовых серверов ZMailer**;
- `drweb-zmailer-av-as` – **Антивирус и Антиспам Dr.Web для почтовых серверов ZMailer**.

Все нижеприведенные команды для подключения репозиториев, импортирования ключей, установки и удаления пакетов должны быть выполнены с правами администратора (root), для чего следует воспользоваться командами **sudo** или **su**.

Debian, Ubuntu (apt)

1. Установка:

Репозиторий для **Debian** защищен с помощью механизма цифровой подписи. Для корректной работы нужно импортировать ключ цифровой подписи командой

```
wget -O - http://officeshield.drweb.com/drweb/drweb.key | apt-key add -
```

или

```
curl http://officeshield.drweb.com/drweb/drweb.key | apt-key add -
```

Чтобы подключить репозиторий, добавьте следующую строку в файл `/etc/apt/sources.list` :

```
deb http://officeshield.drweb.com/drweb/debian stable non-free
```

Для установки **Dr.Web для почтовых серверов UNIX** выполните команды:

```
apt-get update
apt-get install <имя пакета>
```

2. Удаление:

Для удаления **Dr.Web для почтовых серверов UNIX** выполните команду:

```
apt-get remove <имя пакета>
```

Для удаления всех установленных пакетов **Dr.Web** выполните команду (в некоторых системах символ '*' требуется экранировать: '*'):

```
apt-get remove drweb*
```

Для автоматического удаления из системы всех более не используемых пакетов можно воспользоваться командой:

```
apt-get autoremove
```

Обратите внимание на следующие особенности удаления с использованием **apt-get**:

1. Первый вариант команды удалит только пакет <имя пакета>, а остальные пакеты, которые могли быть автоматически установлены при установке этого пакета для удовлетворения его зависимостей, останутся в системе.
2. Второй вариант команды удалит из системы все пакеты, название которых начинается на "drweb" (стандартное наименование для пакетов программных продуктов **Dr.Web**). Обратите внимание, что эта команда удалит из системы все пакеты с таким именем, а не только пакеты продукта **Dr.Web для почтовых серверов UNIX**.
3. Третий вариант команды удалит из системы все пакеты, которые были автоматически установлены для удовлетворения зависимостей других пакетов, но более не требуемые (например, ввиду удаления исходного пакета). Обратите внимание, что эта команда удалит из системы все более не требуемые пакеты, а не только пакеты продукта **Dr.Web для почтовых серверов UNIX**.

Установка и удаление пакетов также могут осуществляться с помощью альтернативных менеджеров (например, **Synaptic** или **aptitude**). Кроме того, альтернативные менеджеры, такие как **aptitude**, рекомендуется использовать для разрешения конфликта пакетов, если он возникнет.

ALT Linux, PCLinuxOS (apt-rpm)

1. Установка:

Чтобы подключить репозиторий, добавьте следующую строку в файл /etc/apt/sources.list:

Для 32-разрядной версии:

```
rpm http://officeshield.drweb.com/drweb/altlinux stable/i386 drweb
```

Для 64-разрядной версии:

```
rpm http://officeshield.drweb.com/drweb/altlinux stable/x86_64 drweb
```

Для установки **Dr.Web для почтовых серверов UNIX** выполните команды:

```
apt-get update
apt-get install <имя пакета>
```

2. Удаление:

Удаление **Dr.Web для почтовых серверов UNIX** в данном случае выполняется так же, как и в **Debian**, **Ubuntu** (см. выше).

Установка и удаление пакетов также могут осуществляться с помощью альтернативных менеджеров (например, **Synaptic** или **aptitude**).

Mandriva (urpmi)

1. Установка:

Загрузите ключ цифровой подписи репозитория с адреса: http://officeshield.drweb.com/drweb/drweb.key и сохраните на диск. Импортируйте ключ с помощью команды

```
rpm --import <путь к ключу репозитория>
```

Откройте файл

http://officeshield.drweb.com/drweb/drweb-i386.urpmi-media

или

http://officeshield.drweb.com/drweb/drweb-x86_64.urpmi-media

и вам будет предложено подключить репозиторий.

Вы также можете подключить репозиторий через командную строку с помощью команды:

```
urpmi.addmedia drweb http://officeshield.drweb.com/drweb/mandriva/stable/i386/
```

или

```
urpmi.addmedia drweb http://officeshield.drweb.com/drweb/mandriva/stable/x86_64/
```

Для установки **Dr.Web для почтовых серверов UNIX** выполните команды:

```
urpmi.update drweb
urpmi <имя пакета>
```

2. Удаление:

Для удаления **Dr.Web для почтовых серверов UNIX** выполните команду:

```
urpme <имя пакета>
```

Для автоматического удаления из системы всех более не используемых пакетов можно воспользоваться командой:

```
urpme --auto-orphans <имя пакета>
```

Обратите внимание на следующие особенности удаления с использованием **urpme**:

1. Первый вариант команды удалит только пакет <имя пакета>, а остальные пакеты, которые могли быть автоматически установлены при установке этого пакета для удовлетворения его зависимостей, останутся в системе.
2. Второй вариант команды удалит из системы пакет <имя пакета>, а также все пакеты, которые были автоматически установлены для удовлетворения зависимостей других пакетов, но более не требуемые (например, ввиду удаления исходного пакета). Обратите внимание, что эта команда удалит из системы все более не требуемые пакеты, а не только пакеты продукта **Dr.Web для почтовых серверов UNIX**.

Установка и удаление пакетов также могут осуществляться с помощью альтернативных менеджеров (например, **rpmdrake**).

Red Hat Enterprise Linux, Fedora, CentOS (yum)

1. Установка:

Добавьте файл со следующим содержимым в каталог /etc/yum.repos.d:

Для 32-разрядной версии:

```
[drweb]
name=DrWeb - stable
baseurl=http://officeshield.drweb.com/drweb/el5/stable/i386/
gpgcheck=1
enabled=1
gpgkey=http://officeshield.drweb.com/drweb/drweb.key
```

Для 64-разрядной версии:

```
[drweb]
name=DrWeb - stable
baseurl=http://officeshield.drweb.com/drweb/el5/stable/x86_64/
gpgcheck=1
enabled=1
gpgkey=http://officeshield.drweb.com/drweb/drweb.key
```

Для установки **Dr.Web для почтовых серверов UNIX** выполните команду:

```
yum install <имя пакета>
```

2. Удаление:

Для удаления **Dr.Web для почтовых серверов UNIX** выполните команду:

```
yum remove <имя пакета>
```

Для удаления всех установленных пакетов **Dr.Web** выполните команду (в некоторых системах символ '*' требуется экранировать: '*'):

```
yum remove drweb*
```

Обратите внимание на следующие особенности удаления с использованием **yum**:

1. Первый вариант команды удалит только пакет <имя пакета>, а остальные пакеты, которые могли быть автоматически установлены при установке этого пакета для удовлетворения его зависимостей, останутся в системе.
2. Второй вариант команды удалит из системы все пакеты, название которых начинается на "drweb" (стандартное наименование для пакетов программных продуктов **Dr.Web**). Обратите внимание, что эта команда удалит из системы все пакеты с таким именем, а не только пакеты продукта **Dr.Web для почтовых серверов UNIX**.

Установка и удаление пакетов также может осуществляться с помощью альтернативных менеджеров (например, **PackageKit** или **Yumex**).

SUSE Linux (Zypper)

1. Установка:

Чтобы подключить репозиторий, запустите следующую команду:

```
zypper ar -t YUM http://officeshield.drweb.com/drweb/el5/stable/i386/ drweb
```

или

```
zypper ar -t YUM http://officeshield.drweb.com/drweb/el5/stable/x86_64/ drweb
```

Для установки **Dr.Web для почтовых серверов UNIX** выполните команды:

```
zypper refresh
zypper install <имя пакета>
```

2. Удаление:

Для удаления **Dr.Web для почтовых серверов UNIX** выполните команду:

```
zypper remove <имя пакета>
```

Для удаления всех установленных пакетов **Dr.Web** выполните команду (в некоторых системах символ '*' требуется экранировать: '*'):

```
zypper remove drweb*
```

Обратите внимание на следующие особенности удаления с использованием **zypper**:

1. Первый вариант команды удалит только пакет <имя пакета>, а остальные пакеты, которые могли быть автоматически установлены при установке этого пакета для удовлетворения его зависимостей, останутся в системе.
2. Второй вариант команды удалит из системы все пакеты, название которых начинается на "drweb" (стандартное наименование для пакетов программных продуктов **Dr.Web**). Обратите внимание, что эта команда удалит из системы все пакеты с таким именем, а не только пакеты продукта **Dr.Web для почтовых серверов UNIX**.

Установка и удаление пакетов также может осуществляться с помощью альтернативных менеджеров (например, **YaST**).

FreeBSD

Установка:

Загрузите архив drweb-maild-meta_current-current~freebsd_all.tar.gz с http://officeshield.drweb.com/drweb/freebsd/ports/, распакуйте в отдельный каталог и выполните команду make install для сборки и установки **Dr.Web для почтовых серверов UNIX**. При установке **Dr.Web для почтовых серверов UNIX** в **FreeBSD** версии 6.1 требуется указать путь к каталогу /usr/ports/Mk с помощью параметра командной строки -I . В этом каталоге располагается дерево портов.

Пример:

```
tar -xzvf drweb-maild-meta_current-current~freebsd_all.tar.gz
make install -I /usr/ports/Mk/
```

Пожалуйста, обратите внимание, что после обновления **Dr.Web для почтовых серверов UNIX** из нативных пакетов требуется перезапустить весь комплекс целиком. Для этого выполните перезагрузку компонента **Dr.Web Monitor**, запустив управляющий скрипт **/etc/init.d/drweb-monitor restart**.

Попытка перезагрузки только модуля **drweb-maild** отправкой ему сигнала SIGHUP приведет к ошибке, если в процессе обновления произошло обновление библиотек подключаемых модулей.

Скрипты настройки

После установки пакетов для подключаемых модулей и MTA можно запустить конфигурационный скрипт configure.pl для базовой настройки компонента **Dr.Web MailD**. Он расположен в каталоге %bin_dir/maild/scripts/.

При запуске `configure.pl`, в общем случае, предложит:

- указать порядок обработки писем конкретным подключаемым модулем (до или после помещения письма в базу данных),
- предпочитаемый язык уведомлений и адрес для их отсылки,
- путь к спискам защищаемых сетей и доменов.

Для настройки почтовой системы следует запустить скрипт `configure_mta.sh`, располагающийся там же. Этот скрипт отвечает за настройку взаимодействия между программным комплексом **Dr.Web для почтовых серверов UNIX** и используемой почтовой системой. При запуске он проверит, установлена ли нужная почтовая система. В случае ее отсутствия скрипт завершит свою работу, а в случае обнаружения предложит ответить в интерактивном режиме на ряд вопросов про отдельные настройки конфигурации используемого MTA. В соответствии с полученными ответами скрипт вносит требуемые изменения в параметры соответствующих конфигурационных файлов.

Этой информации будет достаточно для запуска программного комплекса, но для полноценной работы системы потребуется вручную настроить каждый из компонентов и подключаемых модулей, а также используемую почтовую систему.

Про особенности настройки для разных почтовых систем и подключаемых модулей читайте в соответствующих разделах данного руководства (главы Настройка и запуск, Подключаемые модули, Интеграция с почтовыми системами).

Присутствующие в каталоге `%bin_dir/maild/scripts/` скрипты `plugin_NAME_configure.pl` и `configure_mta.sh` не обеспечивают полноценной настройки работы плагинов и почтовой системы, и могут использоваться только в качестве вспомогательных средств.

Запуск Dr.Web для почтовых серверов UNIX

В данном разделе описана процедура запуска **Dr.Web для почтовых серверов UNIX** в операционных системах **Linux**, **Solaris** и **FreeBSD**.

ОС Linux и Solaris

Для запуска комплекса необходимо:

1. Зарегистрировать продукт.
2. Скопировать или переместить полученный после регистрации лицензионный ключевой файл с расширением `.key` в каталог с исполняемыми файлами программного комплекса **Dr.Web для почтовых серверов UNIX** (по умолчанию `%bin_dir` для UNIX систем). Имя ключевого файла может варьироваться в зависимости от комплекта поставки (подробнее см. в главе Регистрация продукта):
   - Если **Dr.Web для почтовых серверов UNIX** был приобретен как самостоятельный продукт, ключевой файл продукта имеет название `drweb32.key`. В таком случае вы можете скопировать данный файл в каталог `%bin_dir`, не изменяя его имени;
   - В случае приобретения **Dr.Web для почтовых серверов UNIX** в составе программного комплекса **Dr.Web Enterprise Security Suite**, архив содержит 2 файла: ключевой файл для сервера централизованной защиты **Dr.Web Enterprise Server** (`enterprise.key`) и ключевой файл продукта (`agent.key`). Переименуйте `agent.key` как `drweb32.key` и скопируйте его в каталог `%bin_dir`.

   Если вы хотите использовать ключевой файл, расположенный в каком-либо другом каталоге, либо имеющий другое имя (например, `agent.key`), то путь к нему должен быть задан в настройках конфигурационного файла `drweb32.ini` значением параметра **`Key`**. При работе в режиме `Standalone` альтернативный путь к ключу должен быть также задан в настройках конфигурационного файла **Dr.Web Agent** `agent.conf` в значении параметра **`LicenseFile`**.
3. Настроить программный комплекс, внеся все необходимые изменения в конфигурационные файлы. Для настройки компонентов обратитесь к соответствующим разделам документации.
4. Вручную исправить `enable`-файл `drwebd`, присвоив переменной `ENABLE` значение `1`. Это позволит запустить **Dr.Web Daemon**. Если запускать **Dr.Web Daemon** не нужно (используется **Dr.Web Daemon**, запущенный на другом компьютере в локальной сети), то для переменной `ENABLE` нужно оставить присвоенное по умолчанию значение `0`.
5. Вручную исправить `enable`-файл **Dr.Web Monitor**, присвоив переменной `ENABLE` значение `1`. Это позволит запустить **Dr.Web Monitor**.

Расположение `enable`-файлов может меняться в зависимости от способа установки **Dr.Web для почтовых серверов UNIX**:

- Установка при помощи **универсального пакета для UNIX**:

  Файлы располагаются в каталоге `%etc_dir` и называются
  `drwebd.enable`,
  `drweb-monitor.enable`.

- Установка из **нативных DEB-пакетов**:

  Файлы располагаются в каталоге `%etc_dir/defaults` и называются
  `drwebd`,
  `drweb-monitor`.

- Установка из **нативных RPM-пакетов**:

  Файлы располагаются в каталоге `%etc_dir/sysconfig` и называются
  `drwebd.enable`,
  `drweb-monitor.enable`.

6. Запустить инициализационные скрипты для **Dr.Web Daemon** и **Dr.Web Monitor** либо из консоли, либо воспользовавшись встроенными программными средствами вашей операционной системы. После этого **Dr.Web Monitor** сам автоматически запустит остальные компоненты программного комплекса (**Sender**, **Receiver**, **Notifier**, и т.д.).

В случае установки из нативных пакетов в Solaris:

В процессе установки **Dr.Web для почтовых серверов UNIX** система управления сервисами SMF производит попытку запуска компонента **Dr.Web Monitor**. В случае если **Dr.Web Monitor** не может обнаружить лицензионный ключевой файл (например при первой установке комплекса **Dr.Web для почтовых серверов UNIX**), он завершает свою работу и переводится SMF в состояние *maintenance*.

Чтобы запустить **Dr.Web Monitor**, необходимо сбросить состояние *maintenance*:

- Введите команду

  ```
  # svcs -p <FMRI>
  ```

  где FMRI - уникальный идентификатор управляемого ресурса, в данном случае - компонента **Dr.Web Monitor**.

- Принудительно завершите процессы из списка, выводящегося при исполнии команды `svcs -p`.

  ```
  # pkill -9 <PID>
  ```

  где PID - номер процесса, представленного в списке выше.

- Перезапустите **Dr.Web Monitor** командой

  ```
  # svcadm clear <FMRI>
  ```

При установке **Dr.Web для почтовых серверов UNIX** из нативных пакетов в **Solaris**, запуск комплекса производится с помощью системы управления сервисами SMF:

```
# svcadm enable <drweb-monitor>
# svcadm enable <drweb-daemon>
```

Для остановки сервиса введите:

```
# svcadm disable <название сервиса>
```

Модуль **drwebd** может быть запущен в двух режимах:

1. Стандартный запуск посредством скрипта `init`
2. С помощью **Dr.Web Monitor**

При работе во втором режиме необходимо установить значение параметра `ENABLE` в `enable`-файле равным нулю.

ОС FreeBSD

Для запуска комплекса необходимо:

1. Зарегистрировать продукт.
2. Скопировать или переместить полученный после регистрации лицензионный ключевой файл с расширением `.key` в каталог с исполняемыми файлами программного комплекса **Dr.Web для почтовых серверов UNIX** (по умолчанию `%bin_dir` для UNIX-систем). Имя ключевого файла может варьироваться в зависимости от комплекта поставки (подробнее см. в главе Регистрация продукта):
   - Если **Dr.Web для почтовых серверов UNIX** был приобретен как самостоятельный продукт, ключевой файл продукта имеет название `drweb32.key`. В таком случае вы можете скопировать данный файл в каталог `%bin_dir`, не изменяя его имени;
   - В случае приобретения **Dr.Web для почтовых серверов UNIX** в составе программного комплекса **Dr.Web Enterprise Security Suite**, архив содержит 2 файла: ключевой файл для сервера централизованной защиты **Dr.Web Enterprise Server** (`enterprise.key`) и ключевой файл продукта (`agent.key`). Переименуйте `agent.key` как `drweb32.key` и скопируйте его в каталог `%bin_dir`.

   Если вы хотите использовать ключевой файл, расположенный в каком-либо другом каталоге, либо имеющий другое имя (например, `agent.key`), то путь к нему должен быть задан в настройках конфигурационного файла `drweb32.ini` значением параметра **`Key`**. При работе в режиме `Standalone` альтернативный путь к ключу должен быть также задан в настройках конфигурационного файла **Dr.Web Agent** `agent.conf` в значении параметра **`LicenseFile`**.
3. Настроить программный комплекс, внеся все необходимые изменения в конфигурационные файлы. Для настройки компонентов обратитесь к соответствующим разделам документации.
4. Вручную исправить файл `/etc/rc.conf`, добавив в него следующие строки:
   - `drweb_monitor_enable="YES"` – для получения возможности запуска **Dr.Web Monitor**.
   - `drwebd_enable="YES"` – для получения возможности запуска **Dr.Web Daemon**. Если запускать **Dr.Web Daemon** не нужно (используется **Dr.Web Daemon**, запущенный на другом компьютере в локальной сети), то указанную строку можно просто не добавлять в `rc.conf`.
5. Запустить инициализационные скрипты для **Dr.Web Daemon** и **Dr.Web Monitor** либо из консоли, либо воспользовавшись встроенными программными средствами вашей операционной системы. После этого **Dr.Web Monitor** сам автоматически запустит остальные компоненты программного комплекса (**Sender**, **Receiver**, **Notifier**, и т.д.).

Каждый из компонентов можно запускать и отдельно, но при этом модуль **Dr.Web Agent** должен быть запущен самым первым, так как через него остальные компоненты получают свои настройки.

Настройка политик безопасности SELinux

Если используемый вами дистрибутив **Linux** оснащен подсистемой безопасности **SELinux** (`Security-Enhanced Linux` – **Linux** с улучшенной безопасностью), то для того, чтобы антивирусные компоненты (сканирующий демон **Dr.Web Daemon** и консольный сканер **Dr.Web Console Scanner**) работали корректно после установки компонентов приложения, вам потребуется внести изменения в политики безопасности, используемые **SELinux.**

Кроме того, при включенном **SELinux** установка продукта из универсальных пакетов (`.run`) может окончиться неудачей, поскольку будет заблокирована попытка создания пользователя `drweb`, от имени которого работают модули **Dr.Web для почтовых серверов UNIX**.

Перед началом установки рекомендуется проверить режим работы **SELinux**, для этого выполните команду **`getenforce`**. Эта команда выводит на экран текщий режим зашиты:

- **`Permissive`** – защита активна, но используется разрешающая стратегия: действия, нарушающие политики безопасности, не запрещаются, а только фиксируются в журнале аудита.
- **`Enforced`** – защита активна, используется запрещающая стратегия: действия, нарушающие политики безопасности, регистрируются в журнале аудита и блокируются.
- **`Disabled`** – **SELinux** установлен, но неактивен.

Если **SELinux** работает в режиме `Enforced`, следует временно (на период установки продукта и последующей настройки политик безопасности) перевести ее в режим `Permissive`. Для этого выполните команду **`setenforce 0`**, которая временно (до первой перезагрузки системы) переведет **SELinux** в режим `Permissive`. Чтобы вернуть систему в режим `Enforced`, следует выполнить команду
`setenforce 1`.

Обратите внимание, что, какой бы режим защиты вы не установили при помощи команды **`setenforce`**, после перезагрузки операционной системы **SELinux** вернется в режим защиты, заданный в ее настройках (обычно файл настроек **SELinux** находится в каталоге `/etc/selinux`).

В общем случае, при использовании в системе демона **`audit`**, файл журнала аудита располагается в `/var/log/audit/audit.log`. В противном случае сообщения о запрете операции записываются в общий файл журнала `/var/log/messages`.

Чтобы при работающем **SELinux** антивирусные компоненты могли успешно функционировать, необходимо скомпилировать специальные политики безопасности сразу после установки программного продукта, по завершении работы инсталлятора или установщика нативных пакетов.

Пожалуйста, обратите внимание, что в некоторых дистрибутивах **Linux** указанные ниже утилиты могут быть по умолчанию не установлены. В этом случае вам возможно потребуется дополнительно установить содержащие их пакеты.

Чтобы создать необходимые политики:

1. Создайте новый файл с исходным кодом политики **SELinux** (.te файл). Данный файл определяет ограничения, относящиеся к описываемому модулю. Исходный файл политики может быть создан двумя способами:

   1) **С помощью утилиты `audit2allow`**. Это наиболее простой способ. Данная утилита генерирует разрешающие правила на основе сообщений об отказе в доступе в файлах системных журналов. Возможно задать автоматический поиск сообщений в файлах журналов или указать путь к файлу журнала вручную.

Утилита **audit2allow** находится в пакете `policycoreutils-python` или `policycoreutils-devel` (для ОС **RedHat Enterprise Linux, CentOS, Fedora,** в зависимости от версии) или в пакете `python-sepolgen` (для ОС **Debian, Ubuntu**).

Пример использования:

```
# audit2allow -M drweb -i /var/log/audit/audit.log
```

ИЛИ

```
# cat /var/log/audit/audit.log | audit2allow -M drweb
```

В данном примере утилита **audit2allow** производит поиск сообщений об отказе в доступе в файле `audit.log`.

```
# audit2allow -a -M drweb
```

В данном примере утилита **audit2allow** ищет сообщения об отказе в доступе в файлах журналов автоматически.

В обоих случаях в результате работы утилиты создаются два файла: исходный файл политики `drweb.te` и готовый к установке модуль политики `drweb.pp`.

В большинстве случаев вам не потребуется вносить изменения в созданный утилитой файл политики. Поэтому рекомендуется сразу переходить к пункту 4 для установки полученного модуля политики `drweb.pp`. Обратите внимание, что по умолчанию утилита **audit2allow** в качестве результата своей работы выводит на экран готовый вызов команды **semodule**. Скопировав его в командную строку и выполнив, вы выполните пункт 4. Перейдите к пункту 2, только если вы хотите внести изменения в политики, автоматически сформированные для компонентов **Dr.Web для почтовых серверов UNIX**.

2) **С помощью утилиты** `policygentool`. Для этого укажите в качестве параметров имя модуля, работу с которым вы хотите настроить, и полный путь к его исполняемому файлу.

Обратите внимание, что утилита **policygentool**, входящая в состав пакета `selinux-policy` для ОС **RedHat Enterprise Linux** и **CentOS Linux,** может работать некорректно. В таком случае воспользуйтесь утилитой **audit2allow**.

Пример создания политик при помощи policygentool:

- Для модуля **Dr.Web Console Scanner**:

```
# policygentool drweb-scanner /opt/drweb/drweb.real
```

- Для сканирующего демона **Dr.Web Daemon**:

```
# policygentool drweb-daemon /opt/drweb/drwebd.real
```

Вам будет предложено указать несколько общих характеристик домена, после чего для каждого модуля будут созданы три файла, определяющих политику:

`[module_name].te`, `[module_name].fc` и `[module_name].if`.

2. При необходимости отредактируйте сгенерированный исходный файл политики `[module_name].te`, а затем, используя утилиту **checkmodule**, создайте бинарное представление (`.mod` файл) исходного файла локальной политики.

Обратите внимание, что для успешной работы этой команды в системе должен быть установлен пакет `checkpolicy`.

Пример использования:

```
# checkmodule -M -m -o drweb.mod drweb.te
```

3. Создайте устанавливаемый модуль политики (`.pp` файл) с помощью утилиты **`semodule_package`**.

   Пример:

   ```
   # semodule_package -o drweb.pp -m drweb.mod
   ```

4. Для установки созданного модуля политики воспользуйтесь утилитой **`semodule`**.

   Пример:

   ```
   # semodule -i drweb.pp
   ```

После перезагрузки операционной системы подсистема безопасности **SELinux** будет настроена для корректной работы **Dr.Web для почтовых серверов UNIX**.

Для получения дополнительной информации о принципах работы и настройки **SELinux** обратитесь к документации по используемому вами дистрибутиву **Linux.**

Регистрация продукта

Права на использование программного комплекса **Dr.Web для почтовых серверов UNIX** регулируются при помощи специального файла, называемого ключевым файлом. В ключевом файле содержится, в частности, следующая информация:

- перечень компонентов, которые разрешено использовать данному пользователю;
- период, в течение которого разрешено использование продукта;
- список подключаемых модулей программного комплекса **Dr.Web для почтовых серверов UNIX**, которые разрешено использовать данному пользователю (работа некоторых из них, например, **Dr.Web HeadersFilter**, не требует их упоминания в ключевом файле);
- другие ограничения (например, количество писем, которое могут проверять подключаемые модули **Dr.Web MailD** за сутки).

Ключевой файл имеет расширение `key` и при работе комплекса по умолчанию должен находиться в одном каталоге с исполняемыми файлами продукта.

Ключевой файл защищен от редактирования при помощи механизма электронной подписи. Редактирование файла делает его недействительным. Поэтому не следует открывать ключевой файл в текстовых редакторах во избежание его случайной порчи.

Коммерческие пользователи, приобретающие **Dr.Web для почтовых серверов UNIX** у авторизованных поставщиков продукта, получают лицензионный ключевой файл. Параметры этого ключевого файла, регулирующие права пользователя, установлены в соответствии с лицензионным договором. В такой файл также заносится информация о пользователе и продавце продукта.

Для целей ознакомления с программным комплексом **Dr.Web для почтовых серверов UNIX** может быть получен демонстрационный ключевой файл. Такие ключевые файлы обеспечивают полную функциональность основных компонентов комплекса, но имеют ограниченный срок действия и не предполагают оказания поддержки пользователю.

Ключевые файлы поставляются пользователю:

- в виде ключевого файла для рабочей станции `drweb32.key` или в виде ZIP-архива, содержащего этот файл, в случае приобретения **Dr.Web для почтовых серверов UNIX** в качестве отдельного продукта.
- в виде zip-архива, содержащего ключевой файл для сервера **Dr.Web Enterprise Server** (`enterprise.key`) и ключевой файл для рабочей станции (`agent.key`) в случае приобретения **Dr.Web для почтовых серверов UNIX** в составе программного комплекса **Dr.Web Enterprise Security Suite**.

Ключевой файл может быть получен пользователем:

- по электронной почте в виде ZIP-архива, содержащего файл с расширением `key` (обычно после регистрации на веб-сайте, см. ниже). Необходимо извлечь файл при помощи архиватора данного формата и скопировать/переместить его в каталог с исполняемыми файлами программного комплекса **Dr.Web для почтовых серверов UNIX** (по умолчанию `%bin_dir` для UNIX систем);
- в составе дистрибутива продукта;
- на отдельном носителе в виде файла с расширением `key`. В этом случае его необходимо скопировать в вышеуказанный каталог.

Лицензионный ключевой файл высылается пользователям по электронной почте, как правило, после регистрации на специальном веб-сайте (адрес сайта регистрации указан в регистрационной карточке, прилагаемой к продукту). Для получения лицензионного ключевого файла необходимо зайти на указанный сайт, заполнить форму со сведениями о покупателе и ввести в соответствующее поле регистрационный серийный номер (находится на

регистрационной карточке). Это процедура активации лицензии, в результате которой для данного серийного номера создается лицензионный ключевой файл. Затем этот файл высылается на указанный при регистрации адрес электронной почты.

Рекомендуется сохранять лицензионный ключевой файл до истечения срока его действия и использовать его при переустановке или восстановлении программы. В случае утраты лицензионного ключевого файла можно использовать ту же процедуру, что и при активации лицензии: повторно ввести регистрационный серийный номер и адрес электронной почты — и робот вышлет соответствующий указанному серийному номеру ключевой файл.

Регистрация с одним и тем же регистрационным серийным номером допускается не более 25 раз. При необходимости восстановить утерянный лицензионный ключевой файл после 25 регистраций следует разместить запрос на восстановление ключевого файла по адресу в Интернете http://support.drweb.com/request/, указать данные, введенные при регистрации, адрес электронной почты и подробно описать ситуацию. Запрос будет рассмотрен специалистами службы технической поддержки. В случае положительного решения ключевой файл будет либо выдан через автоматизированную систему поддержки пользователей, либо выслан по электронной почте.

Путь к ключу для соответствующего компонента должен быть задан в настройках конфигурационного файла `drweb32.ini` значением параметра **`Key`**.

Пример:

```
Key = %bin_dir/drweb32.key
```

Если ключевой файл, указанный в параметре **`Key`**, не удается прочитать (неверный путь, нет прав), истек срок действия, файл заблокирован или недействителен, то соответствующий компонент завершит свою работу.

Если до истечения срока действия ключевого файла осталось менее двух недель, **Dr.Web Scanner** предупредит об этом при запуске. **Dr.Web Daemon** в такой ситуации может извещать пользователя по электронной почте. Сообщения отправляются для каждого установленного ключевого файла при каждом запуске, перезапуске или перезагрузке **Dr.Web Daemon**, если до истечения срока действия лицензионного ключевого файла осталось менее двух недель. Чтобы воспользоваться этой возможностью, следует настроить параметр **`MailCommand`** в секции `[Daemon]` файла `drweb32.ini`.

Если требуется расположить ключевой файл в каталоге, отличном от стандартного, то следует также указать его новое расположение в параметре **`LicenseFile`** секции `[StandaloneMode]` конфигурационного файла компонента **Dr.Web Agent** (см. раздел Секция [StandaloneMode]).

В программном комплексе **Dr.Web для почтовых серверов UNIX** предусмотрена возможность одновременного использования нескольких ключевых файлов. Список подключаемых модулей, разрешенных к использованию, составляется из всех подключаемых модулей, упомянутых в ключевых файлах (хотя бы в одном из них). Ограничения на работу определенного подключаемого модуля складываются из ограничений, установленных для него в разных ключевых файлах.

При функционировании программного комплекса на работу всех подключаемых модулей должны накладываться одинаковые ограничения, поэтому, когда для разных модулей устанавливаются разные лицензионные ограничения, общее ограничение для работы **Dr.Web для почтовых серверов UNIX** устанавливается по минимальной границе.

Пример:

Используются три ключевых файла. В одном указан антивирусный подключаемый модуль **Drweb**, а также ограничение на проверку 10 000 писем ежесуточно. Во втором указан подключаемый модуль проверки на спам **Vaderetro** и ограничение на проверку 15 000 писем ежесуточно. В третьем снова указан антивирусный подключаемый модуль **Drweb**, а также

ограничение на проверку 10 000 писем ежесуточно.

В итоге при использовании таких лицензионных ключей программный комплекс **Dr.Web для почтовых серверов UNIX** может работать с подключаемыми модулями **Drweb** и **Vaderetro**, поскольку оба они указаны в ключевых файлах. При этом ограничение на проверку писем устанавливается по минимальной границе в размере 15 000 писем ежесуточно, установленной ключевым файлом для модуля **Vaderetro**, несмотря на то, что подключаемый модуль **Drweb** в результате сложения ограничений из разных ключевых файлов может обрабатывать 20 000 писем в сутки.

Модуль обновления Dr.Web Updater

Для автоматизации получения и установки обновлений вирусных баз **Dr.Web** используется модуль обновления **Dr.Web Updater**. Модуль обновления представляет собой написанный на **Perl** скрипт **update.pl** и находится в каталоге, содержащем исполняемые файлы программного комплекса **Dr.Web для почтовых серверов UNIX**.

Модуль обновления **Dr.Web Updater** требует наличия установленного **Perl** 5.8.0 и выше.

Настройки модуля обновления **Dr.Web Updater** хранятся в в секции `[Updater]` главного конфигурационного файла (`drweb32.ini` по умолчанию), который находится в каталоге `%etc_dir`. Для использования другого конфигурационного файла полный путь к нему необходимо указать параметром командной строки при запуске скрипта обновления.

Для запуска скрипта обновления используйте команду:

```
$ %bin_dir/update.pl [параметры]
```

Перечень параметров, которые можно использовать, см. в разделе Параметры командной строки.

В штатном режиме обновления выполняются автоматически, с правами пользователя `drweb`.

Не следует запускать обновление с правами суперпользователя `root`, т.к. в этом случае все последующие попытки автоматического обновления будут завершаться ошибкой из-за попыток доступа к файлам, которые в результате предыдущего обновления сменили своего владельца на суперпользователя `root`.

Обновление антивируса и вирусных баз

Компоненты программного комплекса **Dr.Web для почтовых серверов UNIX** нуждаются в регулярном обновлении баз данных вирусов.

Вирусные базы **Dr.Web для почтовых серверов UNIX** состоят из нескольких файлов с расширением `vdb`. На серверах **Всемирной Системы Обновлений Dr.Web** (**ВСО Dr.Web**) эти файлы могут храниться также в lzma-архивах. При появлении новых вирусов выпускаются небольшие, размером в один или несколько килобайт, файлы (дополнения), которые содержат фрагменты баз, описывающие эти вирусы.

Дополнения являются едиными для всех поддерживаемых платформ и делятся на два вида:

- ежедневные "горячие" обновления (`drwtoday.vdb`);
- еженедельные регулярные обновления (`drwXXXYY.vdb`), где `XXX` – номер версии антивируса, а `YY` – порядковый номер обновления, начиная с номера `00` (например, файл первого регулярного обновления для версии 6.0.1 именуется `drw60100.vdb`).

"Горячие" обновления выпускаются ежедневно или несколько раз в день для оперативной реакции на новые вирусные угрозы. Особенность установки "горячих" дополнений связана с тем, что в промежутке между выходом регулярных (нумерованных) дополнений файл `drwtoday.vdb` пополняется новыми записями, т.е. его необходимо устанавливать вместо имевшегося ранее. В момент выхода очередного регулярного дополнения все записи из этого файла переписываются в регулярное дополнение, а сам он очищается (выпускается файл `drwtoday.vdb`, не содержащий ни одной записи базы данных).

Следовательно, при обновлении баз вручную необходимо устанавливать все отсутствующие у пользователя регулярные дополнения, после чего переписывать файл "горячего" дополнения вместо имевшегося ранее.

Чтобы подключить дополнение к основным вирусным базам, соответствующий файл должен быть помещен в каталог программного комплекса **Dr.Web для почтовых серверов UNIX** (по умолчанию в `%var_dir/bases/`) или иной каталог, определенный в конфигурационном файле.

Сигнатуры, позволяющие обнаруживать и предотвращать распространение вирусоподобных вредоносных программ (рекламных, программ дозвона, программ взлома и т.п.), поставляются в виде двух отдельных вирусных баз с аналогичной структурой – `drwrisky.vdb` и `drwnasty.vdb`. К этим базам также поставляются регулярные обновления `dwrXXYYY.vdb` и `dwnXXYYY.vdb`, а также "горячие" обновления `dwrtoday.vdb` и `dwntoday.vdb`.

Периодически (в частности, в связи с появлением радикально новых вирусных и антивирусных технологий) выпускаются новые версии пакета с обновленными алгоритмами, заложенными в Антивирусное ядро **Dr.Web Engine**. Одновременно с этим сводятся воедино все ранее выпущенные дополнения баз, и новая версия пакета комплектуется новейшими вирусными базами, содержащими описания всех известных на момент ее выхода вирусов. Как правило, при переходе на новую версию пакета сохраняется преемственность формата баз, т.е. новые вирусные базы могут быть подключены к старому Антивирусному ядру. Однако при этом не гарантируется обнаружение или излечение новых вирусов, для борьбы с которыми потребовались обновленные алгоритмы Антивирусного ядра.

При регулярном получении дополнений вирусные базы пакета приобретает следующую структуру:

- `drwebase.vdb` – основная база, получаемая вместе с новой версией пакета;
- `drwXXXYY.vdb` – еженедельные регулярные дополнения вирусных баз;
- `drwtoday.vdb` – "горячие" дополнения;
- `drwnasty.vdb` – основная база вредоносных программ, получаемая вместе с новой версией пакета;
- `dwnXXXYY.vdb` – еженедельные регулярные дополнения базы вредоносных программ;
- `dwntoday.vdb` – "горячие" дополнения базы вредоносных программ;
- `drwrisky.vdb` – основная база потенциально опасных программ, получаемая вместе с новой версией пакета;
- `dwrXXXYY.vdb` – еженедельные регулярные дополнения базы потенциально опасных программ;
- `dwrtoday.vdb` – "горячие" дополнения базы потенциально опасных программ.

Вирусные базы могут быть автоматически обновлены, используя модуль обновления компонентов **Dr.Web Updater** (`%bin_dir/update.pl`).

После установки **Dr.Web для почтовых серверов UNIX** автоматически создаётся файл расписания **cron** (`/etc/cron.d/drweb-update`) для запуска **Dr.Web Updater** каждые 30 минут. Это обеспечивает регулярное обновление и наилучшую защиту.

Настройка cron

Для Linux: при установке компонентов программного комплекса в каталоге `/etc/cron.d/` будет создан пользовательский файл расписания для настройки взаимодействия **cron** с **Dr.Web Updater**.

В создаваемом задании для **crond** используется наиболее распространённый синтаксис *vixie cron*. Если в вашей системе используется другой демон **cron**, например **dcron**, необходимо вручную создать задание для автоматического запуска модуля обновления **Dr.Web Updater**.

Для FreeBSD и Solaris: необходимо вручную настроить **cron** для работы с **Dr.Web Updater**.

Например, при работе с **FreeBSD** можно добавить в `crontab` пользователя `drweb` следующую строку:

```
*/30 * * * * /usr/local/drweb/update.pl
```

При работе с **Solaris** можно использовать следующий набор команд:

```
# crontab -e drweb
# 0,30 * * * * /opt/drweb/update.pl
```

Обратите внимание, что по умолчанию демон **cron** будет запускать модуль **Dr.Web Updater** с периодичностью раз в 30 минут (в 0 и 30 минут каждого часа). Это может вызывать повышенную нагрузку на сервера **BCO Dr.Web** и приводить к задержке обновления. Чтобы избежать подобной ситуации, рекомендуется изменить моменты запуска, заданные по умолчанию, на произвольные.

Параметры командной строки

Параметр `--help` используется для вывода краткой справки о ключах программы.

Для использования другого конфигурационного файла, полный путь к нему необходимо указать параметром командной строки `--ini`. Если имя конфигурационного файла не задано, используется `%etc_dir/drweb32.ini`.

Пример:

```
$ /opt/drweb/update.pl --ini=/path/to/conf_file
```

Параметр командной строки `--what` позволяет временно переопределить значение параметра **`Section`** при запуске модуля обновления. Значение параметра будет действовать до следующего запуска скрипта. Возможные значения: `scanner` или `daemon`.

Пример:

```
$ /opt/drweb/update.pl --what=Scanner
```

Чтобы просмотреть список всех компонентов продукта, доступных для обновления, нужно указать параметр `--components`.

Пример:

```
$ /opt/drweb/update.pl --components
```

В качестве параметра командной строки также может быть указан `--not-need-reload`. Возможны три варианта его использования:

- Если данный параметр не задан, то по завершении работы скрипта обновления **`update.pl`** будут перезагружаться все демоны (**Dr.Web Daemon** для программного комплекса **Dr.Web для почтовых серверов UNIX**), для которых в процессе обновления был изменен/удален/добавлен хотя бы один компонент;
- Если указать параметр `--not-need-reload`, не задав значения, то по завершении работы модуля обновления **`update.pl`** ни один из демонов перезагружаться не будет;
- Если при задании параметра `--not-need-reload` в качестве его значения были указаны названия демонов (через запятую, без пробелов, регистр не важен), то соответствующие демоны перезагружаться не будут, а все остальные — будут при наличии обновлений.

Пример:

```
$ /opt/drweb/update.pl --not-need-reload=drwebd
```

Блокирование обновлений для компонентов

Вы можете заблокировать обновления для определенных компонентов **Dr.Web для почтовых серверов UNIX**.

Чтобы получить список доступных компонентов, запустите **Dr.Web Updater** с параметром командной строки `--components`.

Пример:

```
# ./update.pl --components

Available Components:
    agent
    drweb          (frozen)
    icapd          (frozen)
    vaderetro_lib
```

Если обновления для компонента заблокированы, такой компонент будет отмечен как замороженный (*frozen*). Замороженные компоненты не будут обновляться при запуске **Dr.Web Updater**.

Блокирование обновлений

Чтобы заблокировать обновления для определенных компонентов, запустите **Dr.Web Updater** с параметром командной строки `--freeze=<components>`, где `<components>` – список имен компонентов, разделенных запятыми.

Пример:

```
# ./update.pl --freeze=drweb
Updates for component 'drweb' are frozen.
Run command './updater --unfreeze=drweb' to start updates again.
```

Разблокирование обновлений

Чтобы вновь разрешить обновления для замороженных компонентов, запустите **Dr.Web Updater** с параметром командной строки `--unfreeze=<components>`, где `<components>` – список имен компонентов, разделенных запятыми.

Пример:

```
# ./update.pl --unfreeze=drweb
Updates for component 'drweb' are no longer frozen.
```

Обратите внимание, что размораживание компонента само по себе не приведет к его обновлению.

Восстановление компонентов

При обновлении компонентов **Dr.Web для почтовых серверов UNIX, Dr.Web Updater** сохраняет в рабочем каталоге их резервные копии. Это позволяет вернуть компонент к предыдущему состоянию в случае каких-либо проблем с обновлением.

Чтобы восстановить компонент к предыдущему состоянию, следует запустить **Dr.Web Updater** с параметром командной строки `--restore=<components>`, где `<components>` – это список

имен компонентов, разделенных запятыми.

Пример:

```
# ./update.pl --restore=drweb

Restoring backup for component 'drweb'...
Updates for component 'drweb' are frozen.
Run command './updater --unfreeze=drweb' to start updates again.

Backup for component 'drweb' has been restored!
Dr.Web (R) restore details:

Following files has been restored:
     /var/drweb/bases/drwtoday.vdb
     /var/drweb/bases/dwntoday.vdb
     /var/drweb/bases/dwrtoday.vdb
     /var/drweb/bases/timestamp
     /var/drweb/updates/timestamp
```

При восстановлении компонент будет автоматически заморожен. Чтобы возобновить обновления для восстановленного компонента, его необходимо разморозить.

Настройки

Настройки модуля обновления компонентов **Dr.Web Updater** хранятся в секции `[Updater]` конфигурационного файла программы (по умолчанию `drweb32.ini`), который размещается в каталоге `%etc_dir`.

Устройство конфигурационного файла и краткое описание правил задания параметров конфигурации приведены в разделе Конфигурационные файлы.

Секция [Updater]

UpdatePluginsOnly = {логический}	Значение `Yes` предписывает модулю не производить обновление **Dr.Web Daemon** и **Dr.Web Scanner**, а ограничиться только обновлением подключаемых модулей. Значение по умолчанию: **UpdatePluginsOnly** = No
Section = {Daemon | Scanner}	Указывает, из какой секции конфигурационного файла **Dr.Web Updater** берёт настройки, такие как путь к ключевому файлу, путь к вирусным базам и т.п. Возможные значения параметра: `Scanner` или `Daemon`. Значение параметра возможно временно переопределить при запуске модуля обновления с помощью параметра командной строки `--what`. Измененное таким образом значение параметра будет действовать до следующего запуска скрипта. Значение по умолчанию: **Section** = Daemon
ProgramPath = {путь к файлу}	Путь к исполняемому файлу компонента, который будет обновляться. Требуется модулю обновления для получения информации о версии компонента. Значение по умолчанию: **ProgramPath** = %bin_dir/drwebd

SignedReader = {путь к файлу}	Путь к файлу программы чтения подписанных файлов. Значение по умолчанию: **SignedReader** = %bin_dir/read_signed
LzmaDecoderPath = {путь к каталогу}	Путь к каталогу, в котором располагается утилита lzma, используемая для распаковывания lzma-архивов. Значение по умолчанию: **LzmaDecoderPath** = %bin_dir/
LockFile = {путь к файлу}	Путь к файлу, предназначенному для предотвращения совместного использования некоторых файлов на время их обработки модулем обновления. Значение по умолчанию: **LockFile** = %var_dir/run/update.lock
CronSummary = {логический}	Значение Yes предписывает модулю обновления выдавать отчет сессии обновления на стандартный вывод (stdout). Данный режим используется для посылки уведомлений администратору по электронной почте при запуске модуля обновления демоном **cron**. Значение по умолчанию: **CronSummary** = Yes
DrlFile = {путь к файлу}	Путь к специальному файлу, содержащему список серверов обновления **BCO Dr.Web**. Модуль обновления выбирает сервера обновления из этого списка случайным образом. Подробнее об алгоритме выбора сервера для обновления см. в разделе Процедура обновления Данный файл подписан компанией **«Доктор Веб»**, не подлежит редактированию пользователем и обновляется автоматически. Значение по умолчанию: **DrlFile** = %var_dir/bases/update.drl
CustomDrlFile = {путь к файлу}	Путь к файлу, содержащему альтернативный список серверов обновления **BCO Dr.Web**. Модуль обновления выбирает сервера обновления из этого списка случайным образом. Подробнее об алгоритме выбора сервера для обновления см. в разделе Процедура обновления Данный файл подписан компанией **«Доктор Веб»**, не подлежит редактированию пользователем и обновляется автоматически. Значение по умолчанию: **CustomDrlFile** = %var_dir/bases/custom.drl
FallbackToDrl = {логический}	Разрешение использовать файл **DrlFile** в том случае, если не удалось подключиться ни к одному из серверов, заданных в файле **CustomDrlFile**. В случае если значение параметра No, файл **DrlFile** не используется. В случае если файл **CustomDrlFile** не существует,

	обращение к файлу **DrlFile** производится вне зависимости от значения параметра **FallbackToDrl**. Подробнее об алгоритме выбора сервера для обновления см. в разделе Процедура обновления Значение по умолчанию: **FallbackToDrl** = Yes
DrlDir = {путь к каталогу}	Путь к каталогу, содержащему подписанные **«Доктор Веб»** drl-файлы со списками серверов обновления **BCO Dr.Web** для каждого из подключаемых модулей. Значение по умолчанию: **DrlDir** = %var_dir/drl/
Timeout = {числовое значение}	Максимальное время ожидания для загрузки обновлений с **BCO Dr.Web** в секундах. Значение по умолчанию: **Timeout** = 90
Tries = {числовое значение}	Количество попыток установки соединения модулем обновления **Dr.Web Updater** с серверами **BCO Dr.Web** Значение по умолчанию: **Tries** = 3
ProxyServer = {IP-адрес | имя хоста}	Имя или IP-адрес используемого прокси-сервера. Если здесь указано пусто значение, прокси-сервер не используется. Значение по умолчанию: **ProxyServer** =
ProxyLogin = {текст}	Имя пользователя прокси-сервера (если сервер требует аутентификации). Значение по умолчанию: **ProxyLogin** =
ProxyPassword = {текст}	Пароль пользователя прокси-сервера (если сервер требует аутентификации). Значение по умолчанию: **ProxyPassword** =
LogFileName = {syslog | путь к файлу}	Имя файла журнала или syslog, если журнал будет вестись средствами системного сервиса **syslog** Значение по умолчанию: **LogFileName** = syslog
SyslogFacility = {метка syslog}	Метка записи при использовании системного сервиса **syslog** Значение по умолчанию: **SyslogFacility** = Daemon
LogLevel = {уровень подробности}	Уровень подробности ведения журнала. Допускается использование следующих уровней: • Quiet • Error

	• Warning • Info • Debug • Verbose Значение по умолчанию: **LogLevel** = Info
MaildPidFile = {путь к файлу}	Путь к PID-файлу для **Dr.Web MailD**. Значение по умолчанию: **MaildPidFile** = %var_dir/run/drweb-maild.pid
AgentConfPath = {путь к файлу}	Путь к конфигурационному файлу **Dr.Web Agent**. Значение по умолчанию: **AgentConfPath** = %var_dir/agent.conf
PathToVadeRetro = {путь к файлу}	Путь к библиотеке libvaderetro.so (используется подключаемым модулем **Vaderetro**) Значение по умолчанию: **PathToVadeRetro** = %var_dir/lib/libvaderetro.so
ExpiredTimeLimit = {числовое значение}	Количество дней до истечения срока действия лицензии, в течение которых **Dr.Web Updater** будет пытаться обновить лицензионный ключевой файл. Значение по умолчанию: **ExpiredTimeLimit** = 14
ESLockfile = {путь к файлу}	Путь к блокирующему файлу. Если данный файл существует, то **Dr.Web Updater** перестает использовать расписания **cron** для обновления. Значение по умолчанию: **ESLockfile** = %var_dir/run/es_updater.lock

Процедура обновления

Обновление происходит следующим образом:

1. Модуль обновления **Dr.Web Updater** читает конфигурационный файл (по умолчанию – drweb32.ini, или тот, который указан при помощи аргумента командной строки --ini).
2. Из конфигурационного файла используются параметры, находящиеся в секции [Updater] (описание параметров см. выше), а также параметры **EnginePath**, **VirusBase**, **UpdatePath** и **PidFile**.
3. **Dr.Web Updater** выбирает сервер **ВСО Dr.Web** для получения обновлений. Выбор сервера обновления происходит следующим образом:
   - Производится чтение файлов со списками серверов, указанных в параметрах **DrlFile** и **CustomDrlFile** конфигурационного файла;
   - Если оба файла отсутствуют, то обновление не происходит;
   - Если существует только один из файлов (указанный в **DrlFile** или **CustomDrlFile**), то используется существующий, вне зависимости от значения,

указанного в параметре **FallbackToDrl**;

- Если существуют оба файла, то в первую очередь проверяются сервера из файла, указанного в **CustomDrlFile**;
- Если не получилось подключиться ни к одному из серверов, заданных в файле, указанном в **CustomDrlFile**, и значение параметра **FallbackToDrl**=Yes, то проверяются сервера из файла, указанного в **DrlFile**. В противном случае обновление не происходит.

4. Производятся попытки подключения к случайно выбираемым серверам из списка, содержащегося в файле, до тех пор, пока попытка подключения к серверу не окажется успешной (при подключении **Dr.Web Updater** ожидает ответ от выбранного сервера в течение периода времени, указанного в параметре **Timeout**).
5. Модуль запрашивает с сервера **BCO Dr.Web**, к которому удалось подключиться, список обновлений, а затем lzma-архивы соответствующих баз. В случае отсутствия последних базы скачиваются в виде vdb-файлов. Для распаковывания lzma-архивов используется утилита **lzma**, путь к которой (точнее, к каталогу, в котором она располагается) задается значением параметра **LzmaDecoderPath**.
6. Обновления раскладываются по каталогам, как описано в разделе Обновление антивируса и вирусных баз.

Консольный сканер Dr.Web Scanner

Консольный сканер **Dr.Web Scanner** служит для обнаружения и лечения вирусов на локальной машине. Консольный сканер представлен исполняемым модулем **drweb**.

Dr.Web Scanner проверяет указанные при запуске файлы и загрузочные записи указанных дисков. Для антивирусной проверки и лечения **Dr.Web Scanner** использует Антивирусное ядро **Dr.Web Engine** и вирусные базы, но не использует резидентный модуль **Dr.Web Daemon** (работа производится независимо от него).

Запуск

Запуск **Dr.Web Scanner** осуществляется командой:

```
$ %bin_dir/drweb
```

В том случае, если каталог `%bin_dir` внесен в переменную окружения командной оболочки `PATH`, запуск осуществляется из произвольного каталога. Следует учесть, что последний вариант не рекомендуется из соображений безопасности, равно как и создание символической ссылки на исполняемый файл **drweb** в каком-либо из каталогов типа `/bin/`, `/usr/bin/` и т.д.

Dr.Web Scanner может быть запущен как с правами администратора, так и с правами обычного пользователя. Разумеется, в последнем случае проверка будет выполняться только в тех каталогах, к которым пользователь имеет доступ на чтение, а лечение зараженных файлов будет производиться только в каталогах, в которых он имеет право на запись (обычно это домашний каталог пользователя, `$HOME`). Существуют и другие ограничения при запуске **Dr.Web Scanner** в пользовательском режиме, например, на перемещение и переименование зараженных файлов.

После запуска **Dr.Web Scanner** на экран выводится заставка с названием программы и ее целевой платформы, номером версии и датой ее выпуска, контактными координатами.

Далее выводится сообщение о регистрационных данных пользователя и загрузке вирусных баз **«Доктор Веб»**, включая их обновления, если они были установлены:

```
Dr.Web (R) Сканер для Linux v6.0.1 (19 февраля 2010)
Copyright (c) Игорь Данилов, 1992-2010
"Доктор Веб", Москва, Российская Федерация.
Техподдержка: http://support.drweb.com/
Отдел продаж: http://buy.drweb.com/
Версия оболочки: 6.0.1.10060 <API:2.2>
Антивирусное ядро: 6.0.1.9170 <API:2.2>
Загрузка /var/drweb/bases/drwtoday.vdb - Ok, вирусных записей: 1533
Загрузка /var/drweb/bases/drw60012.vdb - Ok, вирусных записей: 3511
----------------------------------------
Загрузка /var/drweb/bases/drw60000.vdb - Ok, вирусных записей: 1194
Загрузка /var/drweb/bases/dwn60001.vdb - Ok, вирусных записей: 840
Загрузка /var/drweb/bases/drwebase.vdb - Ok, вирусных записей: 78674
Загрузка /var/drweb/bases/drwrisky.vdb - Ok, вирусных записей: 1271
Загрузка /var/drweb/bases/drwnasty.vdb - Ok, вирусных записей: 4867
Вирусных записей: 538681
Ключевой файл: /opt/drweb/drweb32.key
Номер лицензионного ключа: XXXXXXXXXX
Дата активации лицензионного ключа: XXXX-XX-XX
Дата истечения действия лицензионного ключа: XXXX-XX-XX
```

После этого возвращается приглашение командной оболочки.

При запуске **Dr.Web Scanner** с параметрами по умолчанию не осуществляется лечение зараженных файлов. Не предусмотрены также действия в отношении неизлечимых файлов и подозрительных файлов. Все эти действия требуют указания дополнительных параметров командной строки (параметров действия).

Наборы параметров действия могут различаться в каждом конкретном случае, однако обычно представляются целесообразными следующие:

- **cu** – лечение зараженных файлов и системных областей, без удаления, перемещения или переименования зараженных файлов;
- **icd** – удаление неизлечимых файлов;
- **spr** – переименование подозрительных файлов.
- **spm** – перемещение подозрительных файлов;

Запуск **Dr.Web Scanner** с параметром лечения **cu** означает, что программа предпримет попытку восстановить состояние зараженного объекта. Это возможно только тогда, когда обнаружен известный вирус, причем необходимые инструкции по излечению имеются в вирусных базах, однако и в этих случаях попытка излечения может не быть успешной, например, если зараженный файл уже серьезно поврежден.

Если при проверке архивов в их составе были обнаружены зараженные файлы, лечение последних, как и удаление, перемещение или переименование, не производится. Для уничтожения вирусов в таких объектах архивы должны быть вручную распакованы соответствующими программными средствами, желательно, в отдельный каталог, который и будет указан как аргумент при повторном запуске **Dr.Web Scanner**.

При запуске с параметром удаления **icd** программа уничтожит зараженный файл на диске. Этот параметр целесообразен для неизлечимых (необратимо поврежденных вирусом) файлов.

Параметр переименования **spr** вызывает замену расширения имени файла на некое установленное (по умолчанию «`*.#??`», т.е. первый символ расширения заменяется символом «#»). Этот параметр целесообразно применять для файлов других ОС (например, DOS/Windows), выявленных при эвристическом анализе как подозрительные. Переименование сделает невозможным случайный запуск исполняемых модулей в этих системах, загрузку документов **Word** или **Excel** без дальнейшей проверки и таким образом предотвратит заражение возможным вирусом и дальнейшее его распространение.

Параметр перемещения **spm** переместит зараженный (или подозрительный) файл в предназначенный для этого каталог **Карантина** (по умолчанию `%var_dir/infected/`). Пока он имеет чисто теоретическое значение: для файлов других ОС перемещение не имеет смысла, т.к они не могут нанести вреда UNIX-системе, перемещение же подозрительных файлов самой UNIX-системы может вызвать ошибки в работе системы, вплоть до полного ее отказа.

В результате форма запуска **Dr.Web Scanner** для повседневного использования представляется следующей:

```
$ drweb <путь> -cu -icd -spm -ar -ha -fl- -ml -sd
```

Такая команда может быть сохранена в виде текстового файла, который затем с помощью команды:

```
# chmod a+x [имя файла]
```

может быть оформлен как сценарий командной оболочки или серия сценариев для различных ситуаций.

Параметры командной строки

Общий формат запуска программы следующий:

```
$ %bin_dir/drweb <путь> [параметры командной строки]
```

где `<путь>` – путь или пути к проверяемым каталогам или маска проверяемых файлов. Если путь задан с префиксом: `disk://<путь к файлу устройства>` (файлы устройств размещаются в каталоге `/dev`), то будет проверен загрузочный сектор соответствующего устройства и при

необходимости произведено его лечение. Путь может быть предварен необязательным ключом `path`.

Запущенный без параметров, только с указанием пути в качестве аргумента, консольный сканер **Dr.Web Scanner** осуществляет проверку указанного каталога, используя набор параметров по умолчанию (см. ниже). В следующем примере проверяется домашний каталог пользователя:

```
$ %bin_dir/drweb ~
```

По окончании проверки, в случае обнаружения зараженных или подозрительных файлов, **Dr.Web Scanner** выводит информацию обо всех таких файлах в следующем виде:

```
/path/file инфицирован [вирусом] ИМЯ_ВИРУСА
```

После вывода информации о зараженных и подозрительных файлах, если таковые были обнаружены, **Dr.Web Scanner** выдает отчет примерно следующего вида:

```
Отчет для "/opt/drweb/tmp":
Проверено    : 34/32   Исцелено    : 0
Инфицировано : 5/5     Удалено     : 0
Модификаций  : 0/0     Переименовано: 0
Подозрительных: 0/0     Перемещено  : 0
Время проверки: 00:00:02  Скорость    : 5233 KB/s
```

Числа, разделенные символом "/", означают: первое – общее количество файлов, второе – количество файлов в архивах.

Для того, чтобы пользователь имел возможность проверить работоспособность антивируса, в состав дистрибутива продукта входит специальный тестовый файл `readme.eicar.rus`. С помощью текстового редактора из него легко изготовить программу `eicar.com` (см. указания внутри самого файла), которая ведет себя подобно вирусу, вызывая сообщение вида:

```
%bin_dir/doc/eicar.com инфицирован Eicar Test File (Not a Virus!)
```

Этот файл не является вирусом и используется исключительно для тестирования. С этой целью все современные антивирусные программы включают информацию о нем в свои вирусные базы.

Dr.Web Scanner может быть настроен с помощью многочисленных параметров командной строки. В соответствии с соглашениями UNIX-систем, праметры должны быть отделены от указанного путя для проверки пробелои и начинаться с дефиса ("-"). Полный список параметров командной строки для консольного сканера **Dr.Web Scanner** можно получить, запустив программу **drweb** с параметрами `-?`, `-h` или `--help`.

Основные параметры консольного сканера **Dr.Web Scanner** можно сгруппировать следующим образом:

- Параметры области проверки;
- Параметры диагностики;
- Параметры действий;
- Параметры интерфейса.

Параметры области проверки

Эти параметры указывают, где следует проводить проверку на вирусы:

Параметр	Описание
-path [=] {путь}	Задает пути для сканирования. В одном параметре может быть задано несколько путей. Символ '=' можно опустить, в этом случае путь для сканирования отделяется от ключа пробелом. Можно несколько раз указать ключ `path` с разными путями, в этом случае они будт объединены в один список. Кроме того, пути можно задавать, не используя ключ

Параметр	Описание
	path. Если в параметрах запуска путь задан с префиксом: disk://<путь к файлу устройства>, то будет проверен загрузочный сектор (MBR) соответствующего устройства и при необходимости произведено его лечение. Файл устройства – это специальный файл, расположенный в каталоге файлов устройств /dev и имеющий имя вида **sdX** или **hdX**, где **X** – латинская буква (a, b, c, ...). Например: **hda**, **sda**. Таким образом, чтобы проверить, например, загрузочную запись диска **sda**, следует указать путь: disk:///dev/sda
-@[+] {файл}	Задает проверку объектов, перечисленных в указанном файле. Символ «+» (плюс) предписывает не удалять файл со списком объектов по окончании проверки. Этот файл может содержать пути к периодически проверяемым каталогам или просто список файлов, подлежащих регулярной проверке.
--	Указывает, что список объектов для сканирования следует считать из стандартного потока ввода stdin.
-sd	Задает рекурсивный поиск и проверку файлов во вложенных каталогах.
-fl	Указывает следовать символическим ссылкам как для файлов, так и для каталогов. Ссылки, приводящие к «зацикливанию», игнорируются.
-mask	Игнорировать маски имен файлов.

Параметры диагностики

Эти параметры определяют, какие типы объектов и каким образом должны проверяться на вирусы:

Параметр	Описание
-al	Указывает, что по заданным путям необходимо проверять все файлы вне зависимости от их расширения и внутреннего формата. Этот параметр противоположен по действию параметру **-ex**.
-ex	Указывает, что по заданным путям необходимо проверять только файлы заданного типа (разрешения). Разрешения указываются в конфигурационном файле (задается параметром **-ini**) в переменной **FileTypes**. По умолчанию осуществляется проверка файлов со следующими расширениями: EXE, COM, DLL, SYS, VXD, OV?, BAT, BIN, DRV, PRG, BOO, SCR, CMD, 386, FON, DO?, XL?, WIZ, RTF, CL*, HT*, VB*, JS*, INF, PP?, OBJ, LIB, PIF, HLP, MD?, INI, MBR, IMG, CSC, CPL, MBP, SH, SHB, SHS, SHT*, CHM, REG, XML, PRC, ASP, LSP, MSO, OBD, THE*, NWS, SWF, MPP, OCX, VS*, DVB, CPY, BMP, RPM, ISO, DEB, AR?, ZIP, R??, GZ, Z, TGZ, TAR, TAZ, CAB, LHA, LZH, BZ2, MSG, EML, 7Z, CPIO. Этот параметр противоположен по действию параметру **-al**.
-ar[d|m|r][n]	Задает проверку файлов в архивах (ARJ, CAB, GZIP, RAR, TAR, ZIP и др.). Под архивами в данном случае понимаются не только собственно архивы (например, вида *.tar), но и их сжатые формы (например, сжатые TAR–архивы вида *.tar.bz2 и *.tbz). Если параметр указан без дополнительных модификаторов d, m или r, то в случае обнаружения архива с вредоносными или подозрительными файлами, производится только информирование пользователя. Если параметр дополняется модификатором d, m или r, то применяются соответствующие действия для устранения обнаруженной угрозы.
-cn[d|m|r][n]	Задает проверку файлов в контейнерах (HTML, RTF, PowerPoint). Если параметр указан без дополнительных модификаторов d, m или r, то в случае обнаружения контейнера с вредоносными или подозрительными объектами,

Параметр	Описание
	производится только информирование пользователя. Если параметр дополняется модификатором `d`, `m` или `r`, то применяются соответствующие действия для устранения обнаруженной угрозы.
-ml[d|m|r][n]	Задает проверку файлов почтовых программ. Если параметр указан без дополнительных модификаторов `d`, `m` или `r`, то в случае обнаружения файла с вредоносными или подозрительными элементами, производится только информирование пользователя. Если параметр дополняется модификатором `d`, `m` или `r`, то применяются соответствующие действия для устранения обнаруженной угрозы.
-upn	Проверка исполняемых файлов, упакованных LZEXE, DIET, PKLITE, EXEPACK без вывода имен утилит упаковки (в противном случае имя утилиты-упаковщика будет выводиться на экран).
-ha	Задает использование *эвристического анализа* для поиска неизвестных угроз.

Для некоторых параметров доступны также следующие дополнительные модификаторы:

- `d` – использовать удаление объекта для устранения угрозы;
- `m` – использовать перемещение объекта в **Карантин** для устранения угрозы;
- `r` – использовать переименование объекта для устранения угрозы (первый символ расширения заменяется на символ «#»);
- `n` – не указывать в отчете типы архиваторов, контейнеров, почтовых файлов или упаковщиков.

При обнаружении вредоносных элементов в составных объектах (архивах, контейнерах, упакованных или почтовых файлах), указанное действие применяется ко всему составному объекту целиком, а не только к вредоносному элементу.

Параметры действия

Эти параметры определяют, какие действия должны быть выполнены в отношении зараженных (или подозрительных) объектов:

Параметр	Описание
-cu[d|m|r]	Задает действие для инфицированных файлов и загрузочных секторов дисков. Если параметр указан без дополнительных модификаторов, то производится лечение излечимых объектов и удаление неизлечимых файлов (если другое не задано параметром **-ic**). Дополнительные модификаторы позволяют задать иное действие взамен лечения, но оно применяется только для инфицированных файлов. Действие для неизлечимых файлов в таком случае должно быть задано параметром **-ic**.
-ic[d|m|r]	Задает действие для неизлечимых файлов. Если параметр указан без дополнительных модификаторов, то производится только информировании об угрозе.
-sp[d|m|r]	Задает действие для подозрительных файлов. Если параметр указан без дополнительных модификаторов, то производится только информировании об угрозе.
-adw[d|m|r|i]	Задает действие для файлов, содержащих рекламные программы. Если параметр указан без дополнительных модификаторов, то производится только информировании об угрозе.
-dls[d|m|r|i]	Задает действие для файлов, содержащих программы дозвона. Если параметр указан без дополнительных модификаторов, то производится только информировании об угрозе.
-jok[d|m|r|i]	Задает действие для файлов, содержащих программы-шутки. Если параметр указан без дополнительных модификаторов, то производится только информировании об угрозе.
-rsk[d|m|r|i]	Задает действие для файлов, содержащих потенциально опасные программы. Если параметр указан без дополнительных модификаторов, то производится только информировании об угрозе.

Параметр	Описание
-hck[d|m|r|i]	Задает действие для файлов, содержащих программы, используемые для взлома. Если параметр указан без дополнительных модификаторов, то производится только информировании об угрозе.

Дополнительные модификаторы задают действие, необходимое для устранения угрозы:

- d – удаление файла;
- m – перемещение файла в **Карантин**;
- r – переименование файла (первый символ расширения заменяется на символ «#»);
- i – игнорирование (доступно только для незначительных угроз, например, рекламных программ); при использовании этого модификатора объект пропускается без каких-либо действий и оповещение сообщение об угрозе не выводится.

При обнаружении вредоносных элементов в составных объектах (архивах, контейнерах, упакованных или почтовых файлах), указанное действие применяется ко всему составному объекту целиком, а не только к вредоносному элементу.

Параметры интерфейса

Эти параметры определяют условия вывода результатов работы консольного сканера **Dr.Web Scanner**:

Параметр	Описание
-v, -version, --version	Задает вывод информации о версии продукта и версии антивирусного ядра и завершение работы консольного сканера **Dr.Web Scanner**.
-ki	Задает вывод информации о лицензии и ее владельце (только в кодировке UTF8).
-go	Задает пакетный режим работы консольного сканера **Dr.Web Scanner**. Все вопросы, подразумевающие ожидание ответа от пользователя, пропускаются; решения, требующие выбора, принимаются автоматически. Этот режим полезно использовать для автоматической проверки файлов, например, при ежедневной или еженедельной проверке жесткого диска.
-ot	Переключает вывод информации на стандартный вывод (stdout).
-oq	Отключает вывод информации на экран.
-ok	Задает вывод полного списка сканируемых объектов, сопровождая безопасные объекты пометкой **Ok**.
-log =[+] {путь к файлу}	Включает протоколирование работы консольного сканера **Dr.Web Scanner** в указанном файле. При отсутствии имени файла отчет записываться на будет. Символ «+» (плюс) предписывает не перезаписывать файл отчета, а добавлять новую информацию.
-ini = {путь к файлу}	Задает использование указанного конфигурационного файла. По умолчанию консольный сканер **Dr.Web Scanner** использует конфигурационный файл drweb32.ini (этот файл совместно используется компонентами **Dr.Web Daemon**, **Dr.Web Scanner** и **Dr.Web Updaer**). Компонент использует параметры, расположенные в секции [Scanner]. Перечень параметров, задаваемых в секции и их назначение аналогичны параметрам, указанным в секции [Daemon].
-lng = {путь к файлу}	Задает использование указанного альтернативного языкового файла. По умолчанию используется английский язык.
-a = {адрес Агента}	Запустить консольный сканер **Dr.Web Scanner** в режиме централизованной защиты под управлением выбранного **Dr.Web Agent**.
-ni	Отключает использование конфигурационного файла для настройки консольного сканера **Dr.Web Scanner**. Настройка сканирования в данном случае будет осуществляться только с использованием параметров из командной строки.

Параметр	Описание
-ns	Запрещает возможность прерывания проверки, в том числе при получении сигнала остановки процесса (SIGINT).
--only-key	При запуске от **Dr.Web Agent** будет получен только ключевой файл.

Некоторые из параметров отменяют соответствующее им действие, если оканчиваются символом минуса (без пробела). К ним относятся следующие параметры:

-ar -cu -ha -ic -fl -ml -ok -sd -sp

Например, при запуске консольного сканера **Dr.Web Scanner** командой вида:

```
$ drweb <путь> -ha-
```

проверка будет производиться без использования *Эвристического анализа*, который обычно по умолчанию включен.

Для параметров **-cu**, **-ic** и **-sp** «отрицательная» форма отменяет выполнение любых действий, указанных в их описании. Это означает, что информация о зараженных и подозрительных объектах будет фиксироваться в отчете, но никаких действий под устранению представляемых ими угроз предприниматься не будет.

Для параметров **-al** и **-ex** «отрицательная» форма не предусмотрена, однако задание одного из них отменяет действие другого.

Если не производились действия по перенастройке программы, то по умолчанию (то есть без отдельного указания параметров) **Dr.Web Scanner** запускается с параметрами:

-ar -ha -fl- -ml -sd -al -ok

Этот набор параметров по умолчанию (включающий проверку архивов и упакованных файлов, файлов почтовых программ, рекурсивный поиск, эвристический анализ и т.д.) достаточно целесообразен для целей диагностики и может использоваться в большинстве типичных случаев. Если какой-либо из параметров по умолчанию не нужен в конкретной ситуации, его можно отключить, указав после него минус, как это было показано выше на примере параметра **-ha** (использование *Эвристического анализа*).

Следует добавить, что отключение проверки архивированных и упакованных файлов резко снижает уровень антивирусной защиты, т.к. именно в виде архивов (часто самораспаковывающихся) распространяются файловые вирусы в виде почтовых вложений. Документы прикладных программ, потенциально подверженные заражению макровирусами (**Word, Excel** и др.), также обычно пересылаются по электронной почте в архивированном и упакованном виде.

При запуске **Dr.Web Scanner** с параметрами по умолчанию не осуществляется лечение зараженных файлов. Не предусмотрены также действия в отношении неизлечимых файлов и подозрительных файлов. Все эти действия требуют указания дополнительных параметров командной строки – параметров действия.

Настройки

Можно использовать **Dr.Web Scanner** с настройками по умолчанию, но значительно удобнее настроить его для соответствия конкретным требованиям и условиям эксплуатации. Настройки **Dr.Web Scanner** хранятся в конфигурационном файле программы (по умолчанию `drweb32.ini`), который размещается в каталоге `%etc_dir`. Для использования другого конфигурационного файла полный путь к нему необходимо указать параметром командной строки при запуске **Dr.Web Scanner**, например:

```
$ %bin_dir/drweb -ini=%bin_dir/etc/drweb.ini
```

Устройство конфигурационного файла и краткое описание правил задания параметров конфигурации приведены в разделе Конфигурационные файлы.

Секция [Scanner]

EnginePath = {путь к файлу}	Расположение модуля **drweb32.dll** (Антивирусное ядро **Dr.Web Engine**). Этот параметр также используется модулем обновления **Dr.Web Updater**. Значение по умолчанию: **EnginePath** = %bin_dir/lib/drweb32.dll
VirusBase = {список масок файлов}	Маски для подключаемых вирусных баз. Этот параметр также используется модулем обновления **Dr.Web Updater**. Допустимо перечисление нескольких масок через запятую. По умолчанию вирусные базы хранятся в файлах с расширением .vdb Значение по умолчанию: **VirusBase** = %var_dir/bases/*.vdb
UpdatePath = {путь к каталогу}	Этот параметр используется модулем обновления **Dr.Web Updater** и должен быть задан обязательно. Значение по умолчанию: **UpdatePath** = %var_dir/updates/
TempPath = {путь к каталогу}	Этот каталог используется Антивирусным ядром **Dr.Web Engine** для создания временных файлов. При нормальной работе каталог практически не используется, он нужен для распаковки некоторых видов архивов, или когда в системе не хватает памяти. Значение по умолчанию: **TempPath** = /tmp/
LngFileName = {путь к файлу}	Расположение файла языковых ресурсов. По умолчанию файлы языковых ресурсов имеют расширение .dwl Значение по умолчанию: **LngFileName** = %bin_dir/lib/ru_scanner.dwl
Key = {путь к ключевому файлу}	Расположение ключевого файла (лицензионного или демонстрационного). По умолчанию ключевой файл имеет расширение .key Значение по умолчанию: **Key** = %bin_dir/drweb32.key
OutputMode = {Terminal | Quiet}	Режим вывода информации при запуске: • Terminal – вывод на консоль, • Quiet – отменяет вывод. Значение по умолчанию: **OutputMode** = Terminal
HeuristicAnalysis = {логический}	Включение/отключение использования *Эвристического анализа.* *Эвристический анализ* делает возможным обнаружение

	неизвестных вирусов по априорным соображениям об устройстве вирусного кода. Особенностью этого типа поиска вирусов является вероятностный характер обнаружения заражения, что позволяет говорить не о зараженных, а о подозрительных объектах. При отключении этого режима осуществляется только поиск известных вирусов по вирусным базам **«Доктор Веб»**. Целый класс программ ввиду использования сходного с вирусами кода может вызывать ложные срабатывания *Эвристического анализа*. Кроме того, данный режим может незначительно увеличить время проверки. Данные обстоятельства могут быть доводами в пользу отключения использования *Эвристического анализа*. Вместе с тем, включение этого типа анализа увеличивает надежность антивирусной защиты. Все файлы, обнаруженные методом *Эвристического анализа*, лучше всего отправить разработчикам через сайт http://vms.drweb.com/sendvirus/. Отправку подозрительных файлов рекомендуется производить следующим образом: запаковать файл в архив с паролем, пароль сообщить в теле письма, при этом желательно приложить отчет **Dr.Web Scanner**. Значение по умолчанию: **HeuristicAnalysis** = Yes
ScanPriority = {числовое значение}	Приоритет работы **Dr.Web Scanner**. Значение параметра должно быть в диапазоне от высшего значения (-20) до низшего (19 для **Linux**, 20 для остальных ОС). Значение по умолчанию: **ScanPriority** = 0
FilesTypes = {список расширений файлов}	Список типов файлов, которые будут проверяться при сканировании по типу, т.е. когда параметр **ScanFiles** (см. ниже) имеет значение ByType. Допускаются символы маски '*' и '?'. Значение по умолчанию: FilesTypes = EXE, COM, SYS, OV?, BAT, BIN, DRV, PRG, BOO, SCR, CMD, VXD, 386, DLL, FON, DO?, XL?, WIZ, RTF, CL*, HT*, VB*, JS*, INF, AR?, ZIP, R??, PP?, OBJ, LIB, HLP, MD?, INI, MBR, IMG, CSC, CPL, MBP, SHS, SHB, PIF, SO, CHM, REG, XML, PRC, ASP, LSP, MSO, OBD, THE*, NWS, SWF, BMP, MPP, OCX, DVB, CPY, MSG, EML
FilesTypesWarnings = {логический}	Выводить ли предупреждение о файлах неизвестных типов. Значение по умолчанию: **FilesTypesWarnings** = Yes
ScanFiles = {All | ByType}	Дополнительное ограничение на файлы, подлежащие проверке. При задании значения ByType учитываются расширения файлов, значения которых заданы или по умолчанию, или в параметре (параметрах) **FilesTypes**. В противном случае проверяются все файлы. Внутри почтовых файлов всегда действует режим All. Значение ByType может быть использовано только в режиме

	локального сканирования. Значение по умолчанию: **ScanFiles** = All
ScanSubDirectories = {логический}	Проверка содержимого вложенных подкаталогов. Значение по умолчанию: **ScanSubDirectories** = Yes
CheckArchives = {логический}	Проверка файлов, содержащихся в архивах. Поддерживаются архивы форматов ZIP (WinZip, InfoZIP и др.), RAR, ARJ, TAR, GZIP, CAB и др. Значение по умолчанию: **CheckArchives** = Yes
CheckEMailFiles = {логический}	Проверка файлов в почтовых (e-mail) форматах. Значение по умолчанию: **CheckEMailFiles** = Yes
ExcludePaths = {список путей (масок)}	Маски для тех файлов, которые не должны проверяться. Значение по умолчанию: **ExcludePaths** = /proc,/sys,/dev
FollowLinks = {логический}	Следование символическим ссылкам при сканировании. Значение по умолчанию: **FollowLinks** = No
RenameFilesTo = {маска}	Маска для переименования файлов, если сработало действие Rename. Значение по умолчанию: **RenameFilesTo** = #??
MoveFilesTo = {путь к каталогу}	Путь к каталогу **Карантина**. Значение по умолчанию: **MoveFilesTo** = %var_dir/infected/
EnableDeleteArchiveAction = {логический}	Разрешение применения действия Delete для составных объектов (архивов, почтовых ящиков, писем, HTML-страниц и прочих контейнеров), если они содержат зараженные объекты. Важно понимать, что при наличии данного разрешения будет удален весь составной объект целиком, а не только содержащийся в нем вредоносный элемент. Значение по умолчанию: **EnableDeleteArchiveAction** = No
InfectedFiles = {действие}	Задает реакцию на обнаружение файла, зараженного известным вирусом. Допустимые значения параметра: Report, Cure, Delete, Move, Rename, Ignore. Удаление и перемещение, заданное в связи с обнаружением зараженных объектов в архивах и других контейнерах, применяется к соответствующему контейнеру целиком.

	Значение по умолчанию: **InfectedFiles** = Report

Далее указаны параметры, аналогичные параметру **InfectedFiles** и задающие реакцию программы на обнаружение тех или иных объектов. Для них предусмотрены те же возможные значения, что и для параметра **InfectedFiles**, кроме значения Cure:

SuspiciousFiles = {действие}	Действие, которое нужно выполнить в случае, если файл заражен неизвестным вирусом или представляет собой потенциальную угрозу (если сработал *Эвристический анализ*). Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **SuspiciousFiles** = Report
IncurableFiles = {действие}	Действие, которое нужно выполнить в случае, если зараженный файл не может быть вылечен (параметр имеет смысл, только если **InfectedFiles** = Cure) Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **IncurableFiles** = Report
ActionAdware = {действие}	Действие, которое нужно выполнить в случае, если файл содержит программу для показа рекламы (adware). Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **ActionAdware** = Report
ActionDialers = {действие}	Действие, которое нужно выполнить в случае, если файл содержит программу автоматического дозвона. Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **ActionDialers** = Report
ActionJokes = {действие}	Действие, которое нужно выполнить в случае, если файл содержит программу-шутку, которая может пугать или раздражать пользователя. Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **ActionJokes** = Report
ActionRiskware = {действие}	Действие, которое нужно выполнить в случае, если файл содержит потенциально опасную программу, которая может быть использована не только ее владельцем, но и злоумышленниками. Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **ActionRiskware** = Report

ActionHacktools = {действие}	Действие, которое нужно выполнить в случае, если файл содержит программу, которая используется для взлома компьютеров. Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **ActionHacktools** = Report
ActionInfectedMail = {действие}	Действие, которое нужно выполнить в случае, если почтовое сообщение или почтовый ящик содержат зараженный объект. Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **ActionInfectedMail** = Report
ActionInfectedArchive = {действие}	Действие, которое нужно выполнить в случае, если архив (ZIP, TAR, RAR и др.) содержит зараженный файл. Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **ActionInfectedArchive** = Report
ActionInfectedContainer = {действие}	Действие, которое нужно выполнить в случае, если файл контейнер (OLE, HTML, PowerPoint и др.) содержит зараженный объект. Допустимые значения параметра: Report, Delete, Move, Rename, Ignore. Значение по умолчанию: **ActionInfectedContainer** = Report

Параметры регистрации событий:

LogFileName = {syslog | путь к файлу}	Имя файла журнала или syslog, если нужно использовать системный сервис **syslog**. Значение по умолчанию: **LogFileName** = syslog
SyslogFacility = {метка syslog}	Метка записи при использовании системного сервиса **syslog**. Значение по умолчанию: **SyslogFacility** = Daemon
SyslogPriority = {уровень подробности}	Уровень подробности ведения журнала при использовании системного сервиса **syslog**. Допускается использование следующих уровней: • Error • Alert • Warning • Info • Notice Значение по умолчанию: **SyslogPriority** = Info

LimitLog = {логический}	Ограничение размера файла журнала, если не используется **syslog**. Ограничение размера файла отчета реализуется следующим образом: при запуске **Dr.Web Scanner** проверяет размер файла журнала, и если он превышает значение, заданное в параметре **MaxLogSize**, файл журнала стирается и ведение журнала начинается с нуля. Значение по умолчанию: **LimitLog** = No
MaxLogSize = {числовое значение}	Максимальный размер файла журнала в килобайтах, если не используется **syslog** и **LimitLog** = Yes. Если указано значение 0, размер файла журнала проверяться не будет. Значение по умолчанию: **MaxLogSize** = 512
LogScanned = {логический}	Вывод в журнал информации обо всех проверяемых объектах, независимо от того, обнаружены вирусы или нет. Значение по умолчанию: **LogScanned** = Yes
LogPacked = {логический}	Вывод в журнал дополнительной информации о файлах, упакованных утилитами DIET, PKLITE и др. Значение по умолчанию: **LogPacked** = Yes
LogArchived = {логический}	Вывод в журнал дополнительной информации об архиваторах. Значение по умолчанию: **LogArchived** = Yes
LogTime = {логический}	Вывод в журнал времени каждой записи. Параметр игнорируется, если используется **syslog** Значение по умолчанию: **LogTime** = Yes
LogStatistics = {логический}	Запись в журнал суммарной статистики задания для сканирования. Значение по умолчанию: **LogStatistics** = Yes
RecodeNonprintable = {логический}	Перекодировка при выводе в журнал символов, не являющихся отображаемыми для данного терминала (см. следующие два параметра). Значение по умолчанию: **RecodeNonprintable** = Yes
RecodeMode = {Replace | QuotedPrintable}	При **RecodeNonprintable** = Yes задает метод перекодировки неотображаемых символов. При **RecodeMode** = Replace все такие символы заменяются на значение параметра **RecodeChar** (см. ниже). При **RecodeMode** = QuotedPrintable производится перекодировка неотображаемых символов в формат Quoted

	Printable. Значение по умолчанию: **RecodeMode** = QuotedPrintable
RecodeChar = {"?" | "_" | ...}	При **RecodeMode** = Replace задает символ, на который будут заменены все неотображаемые символы. Значение по умолчанию: **RecodeChar** = "?"

Следующие параметры могут быть использованы для уменьшения времени проверки архивов за счет отказа от проверки некоторых объектов в архиве.

MaxCompressionRatio = {числовое значение}	Максимальный коэффициент сжатия, т.е. отношение длины файла в распакованном виде к длине файла в запакованном виде (внутри архива). Если коэффициент превышает данное значение, файл не будет извлечен и, соответственно, не будет проверен. Письмо с таким файлом воспринимается программой как *"почтовая бомба"*. Параметр может принимать только натуральные значения. Если указано значение 0, проверка коэффициента сжатия проводиться не будет. Значение по умолчанию: **MaxCompressionRatio** = 5000
CompressionCheckThreshold = {числовое значение}	Минимальный размер файла внутри архива (в килобайтах), начиная с которого будет производиться проверка коэффициента сжатия (если это предписано параметром **MaxCompressionRatio**). Значение по умолчанию: **CompressionCheckThreshold** = 1024
MaxFileSizeToExtract = {числовое значение}	Максимальный размер файла, извлекаемого из архива, в килобайтах. Если размер файла внутри архива превышает это значение, он будет пропущен. Письмо с таким файлом воспринимается программой как *"почтовая бомба"*. Значение по умолчанию: **MaxFileSizeToExtract** = 500000
MaxArchiveLevel = {числовое значение}	Максимальный уровень вложенности архивов (когда архив вложен в архив, который тоже вложен в архив и т.д.). При превышении этого уровня архив будет пропущен (не будет проверен). Письмо с таким файлом воспринимается программой как *"почтовая бомба"*. Если указано значение 0, уровень вложенности проверяемых архивов проверяться не будет. Значение по умолчанию: **MaxArchiveLevel** = 8
MaximumMemoryAllocationSize = {числовое значение}	Максимальный размер памяти в мегабайтах, выделяемой **Dr.Web Scanner** при сканировании одного файла. Если установлено значение 0, размер выделяемой памяти не ограничен.

	Значение по умолчанию: **MaximumMemoryAllocationSize** = 0
ScannerScanTimeout = {числовое значение}	Максимальное время сканирования одного файла (в секундах). Если установлено значение 0, время сканирования одного файла не ограничено. Значение по умолчанию: **ScannerScanTimeout** = 0
MaxBasesObsolescencePeriod = {числовое значение}	Максимальный период времени (в часах) с момента последнего обновления, в течение которого вирусные базы считаются "свежими". По истечении этого времени в консоли выводится уведомление о том, что базы устарели. Если установлено значение 0, "свежесть" вирусных баз не проверяется. Значение по умолчанию: **MaxBasesObsolescencePeriod** = 24
ControlAgent = {адрес}	Адрес сокета **Dr.Web Agent**. **Пример**: **ControlAgent** = inet:4040@127.0.0.1,local:/var/drweb/ipc/.agent **Dr.Web Scanner** получает от **Dr.Web Agent** ключ и конфигурационный файл (если в качестве значения параметра **OnlyKey** задано No). Значение по умолчанию: **ControlAgent** = local:%var_dir/ipc/.agent
OnlyKey = {логический}	Подключение возможности запросить только ключевой файл от **Dr.Web Agent**, не запрашивая конфигурацию. При этом будет использоваться локальный конфигурационный файл. Если указан адрес сокета **Dr.Web Agent** и значение параметра **OnlyKey** установлено в No, **Dr.Web Agent** будет отправляться статистика работы **Dr.Web Scanner** (после сканирования каждого файла **Dr.Web Scanner** будет отправлять информацию **Dr.Web Agent**). Значение по умолчанию: **OnlyKey** = No

Коды возврата

По окончании работы **Dr.Web Scanner** возвращает код возврата, по которому можно определить, с каким результатом завершено сканирование.

Код возврата всегда образуется как комбинация (сумма) кодов, сопоставленных определенным событиям в процессе сканирования. Возможные значения кодов и соответствующие им события следующие:

Код	Событие
1	Обнаружены известные вирусы
2	Обнаружены модификации известных вирусов
4	Обнаружены подозрительные на вирус объекты

Код	Событие
8	В архиве, контейнере или почтовом ящике обнаружены известные вирусы
16	В архиве, контейнере или почтовом ящике обнаружены модификации известных вирусов
32	В архиве, контейнере или почтовом ящике обнаружены подозрительные на вирус объекты
64	Успешно выполнено лечение хотя бы одного зараженного вирусом объекта
128	Выполнено удаление/переименование/перемещение хотя бы одного зараженного файла

Результирующий код возврата, формируемый по завершению проверки, равен сумме кодов тех событий, которые произошли во время проверки (и его слагаемые могут однозначно быть по нему восстановлены). Например, код возврата 9 = 1 + 8 означает, что во время проверки обнаружены известные вирусы (вирус), в том числе в архиве; обезвреживание не проводилось; больше никаких «вирусных» событий не было.

Если в процессе сканирования ни одного из указанных инцидентов не было, **Dr.Web Scanner** возвращает 0.

Одна из известных особенностей поведения **Dr.Web Scanner** состоит в том, что в случае отсутствия инцидентов при сканировании он может вернуть код 128. Возврат этого кода равносилен возврату кода 0.

Антивирусный модуль Dr.Web Daemon

Dr.Web Daemon – основной компонент безопасности. Он представляет собой постоянно загруженный (резидентный) антивирусный модуль **drwebd**, который позволяет по запросу от других компонентов комплекса проверять файлы на диске или данные, переданные ему через сокет. Запросы на антивирусную проверку осуществляются по специальному протоколу через UNIX- или TCP-сокеты. **Dr.Web Daemon** использует то же Антивирусное ядро **Dr.Web Engine** и вирусные базы, что и **Dr.Web Scanner**, и способен обнаруживать и лечить все вирусы, известные Антивирусному ядру **Dr. Engine**.

Dr.Web Daemon всегда готов к выполнению своих функций и имеет понятный и доступный протокол для запросов сканирования, что делает его подходящим компонентом для создания антивирусного фильтра для файловых серверов. Программный комплекс **Dr.Web для почтовых серверов UNIX** является готовым решением по интеграции **Dr.Web Daemon** с почтовыми серверами UNIX.

Обратите внимание, что **Dr.Web Daemon** не может проверять содержимое зашифрованных файлов, поскольку для анализа их содержимого требуется знание пароля. Поэтому такие файлы пропускаются без проверки, а **Dr.Web Daemon** возвращает специальный код ответа вызвавшему его клиентскому приложению.

Параметры командной строки

Для запуска **Dr.Web Daemon** используется следующая команда:

```
drwebd [параметры]
```

Dr.Web Daemon допускает использование следующих параметров:

Краткий вариант	Расширенный вариант	Аргументы
-h, -?	-help, --help	
Описание: Вывод на экран краткой справки по имеющимся параметрам командной строки и завершение работы модуля		
-a		<адрес Агента>
Описание: Запуск **Dr.Web Daemon** в режиме центральной защиты под управлением указанного **Dr.Web Agent**		
-ini		<путь к файлу>
Описание: Использование указанного конфигурационного файла		
	--foreground	<yes|no>
Описание: Задание режима работы **Dr.Web Daemon** при запуске. Если выбрано значение yes, то **Dr.Web Daemon** будет работать как приоритетная задача. При значении no **Dr.Web Daemon** будет работать в фоновом режиме		
	--check-only	<параметры командной строки для проверки>
Описание: Проверка правильности конфигурации **Dr.Web Daemon** при запуске. Если указаны какие-либо параметры командной строки, то правильность задаваемых с их помощью значений также будет проверена		
	--only-key	

Краткий вариант	Расширенный вариант	Аргументы
Описание: При запуске **Dr.Web Daemon** получит от **Dr.Web Agent** только лицензионный ключевой файл		

Запуск

В процессе загрузки **Dr.Web Daemon** выполняются следующие действия:

1. Поиск и загрузка конфигурационного файла. Если конфигурационный файл не найден, загрузка **Dr.Web Daemon** прекращается. Путь к конфигурационному файлу может быть задан при запуске параметром командной строки `-ini: {путь/к/drweb32.ini}`, иначе будет использовано значение `%etc_dir/drweb32.ini`, заданное по умолчанию. При загрузке проверяется допустимость некоторых параметров и, если значение параметра недопустимо, берется значение по умолчанию;
2. Создается файл отчета. Каталог с файлом отчета должен быть доступен на запись пользователю, с правами которого работает **Dr.Web Daemon**. Каталог `/var/log/`, используемый по умолчанию, недоступен пользователям на запись. Поэтому, если задано значение параметра **`User`**, необходимо также указать путь к альтернативнойму каталогу для хранения отчётов в значении параметра **`LogFileName`**;
3. Производится загрузка ключевого файла по пути, указанному в конфигурационном файле. Если ключевой файл не найден, загрузка **Dr.Web Daemon** прекращается;
4. Если задан параметр **`User`**, **Dr.Web Daemon** пытается изменить свои права;
5. Производится загрузка антивирусного ядра **Dr.Web Engine** (файл **`drweb32.dll`**). Если Антивирусное ядро не найдено (ошибки в конфигурационном файле) или повреждено, загрузка **Dr.Web Daemon** прекращается;
6. Загружаются вирусные базы. Поиск вирусных баз осуществляется по заданным в конфигурационном файле путям, порядок загрузки вирусных баз не регламентирован. Если вирусные базы повреждены или отсутствуют, загрузка **Dr.Web Daemon** продолжается;
7. **Dr.Web Daemon** отключается от терминала, поэтому сообщения о дальнейших проблемах не могут быть выведены на терминал и выводятся только в файл отчета;
8. Создается сокет, в случае использования TCP-сокетов, возможно, не один. Если какой-либо TCP-сокет создать не удалось, загрузка **Dr.Web Daemon** продолжается. В случае использования UNIX-сокета следует убедиться, что каталог, содержащий его, доступен на запись и чтение пользователю, с чьими правами работает **Dr.Web Daemon**. Для пользователей, с правами которых будут работать интеграционные модули, каталог должен быть доступен на выполнение, а сам файл сокета — на запись и чтение. Каталог по умолчанию (`/var/run/`) недоступен пользователям на запись и выполнение. Поэтому, если задано значение параметра **`User`**, необходимо также указать путь к альтернативному каталогу для сокетов в значении параметра **`Socket`**. Если UNIX-сокет создать не удалось, загрузка **Dr.Web Daemon** прекращается;
9. После этого создается PID-файл, в котором хранится информация об идентификаторе процесса **Dr.Web Daemon** и о транспортных адресах, по которым доступен **Dr.Web Daemon**. Каталог с PID-файлом также должен быть доступен на запись пользователю, с правами которого работает **Dr.Web Daemon**. Используемый по умолчанию каталог `/var/run/` недоступен пользователям на запись и выполнение. Поэтому, если задано значение параметра **`User`**, необходимо также указать путь к альтернативному каталогу для PID-файла в значении параметра **`PidFile`**. Если создать PID-файл не удалось, загрузка **Dr.Web Daemon** прекращается.

Проверка работоспособности Dr.Web Daemon

Если в ходе загрузки не возникло проблем, **Dr.Web Daemon** готов к работе. Для проверки корректности загрузки **Dr.Web Daemon** можно узнать, созданы ли необходимые для его работы сокеты. Для этого используется команда:

```
$ netstat -a
```

В случае TCP-сокетов:

```
. . .
Active Internet connections (servers and established)
Proto   Recv-Q   Send-Q   Local Address   Foreign Address   State
. . .
tcp     0        0        localhost:3000  *:*               LISTEN
. . .
```

В случае UNIX-сокетов:

```
. . .
Active UNIX domain sockets (servers and established)
Proto   RefCnt   Flags   Type    State      I-Node   Path
. . .
unix    0       [ ACC ]  STREAM  LISTENING  1127     %var_dir/.daemon
. . .
```

Если созданные сокеты не появились в списке, значит, имеются проблемы загрузки.

Для проверки работоспособности **Dr.Web Daemon** можно использовать **Консольный клиент Dr.Web Daemon** **drwebdc**, запустив его для получения служебной информации о **Dr.Web Daemon**. Если запустить **drwebdc**, он выдаст список всех поддерживаемых параметров.

В случае TCP-сокета:

```
$ drwebdc -nHOSTNAME -pPORTNUM -sv -sb
```

В случае UNIX-сокета:

```
$ drwebdc -uSOCKETFILE -sv -sb
```

На консоли появится информация, подобная следующей:

```
- Version: DrWeb Daemon 6.00
- Loaded bases:
Base /var/drweb/bases/drwtoday.vdb
contains 5 records.
Base /var/drweb/bases/drw60003.vdb
contains 409 records.
Base /var/drweb/bases/drw60002.vdb
contains 543 records.
Base /var/drweb/bases/drwebase.vdb
contains 51982 records.
Base /var/drweb/bases/drw60001.vdb
contains 364 records.
Total 53303 virus-finding records.
```

Если этого не произошло, следует провести расширенную диагностику:

В случае TCP-сокета:

```
$ drwebdc -nHOSTNAME -pPORTNUM -sv -sb -v
```

В случае UNIX-сокета:

```
$ drwebdc -uSOCKETFILE -sv -sb -v
```

Более подробный вывод может прояснить ситуацию:

```
dwlib: fd: connect() failed - Connection refused
dwlib: tcp: connecting to 127.0.0.1:3300 - failed
dwlib: cannot create connection with a DrWeb daemon
ERROR: cannot retrieve daemon version
Error -12
```

Проверить работоспособность **Dr.Web Daemon** можно с помощью программы `eicar.com`, получаемой из входящего в дистрибутив файла `readme.eicar.rus` с помощью любого текстового редактора (см. указания об этом внутри самого файла).

Для TCP-сокета:

```
$ drwebdc -n<ИМЯ_УЗЛА> -p<НОМЕР_ПОРТА> eicar.com
```

Для UNIX-сокета:

```
$ drwebdc -u<ФАЙЛ_СОКЕТА> eicar.com
```

Результатом команды должно быть сообщение:

```
Results: daemon return code 0x20
(known virus is found)
```

Если его не появилось, проверьте в файле отчета **Dr.Web Daemon** наличие записи о проверке этого файла. Если файл так и не был проверен, проведите расширенную диагностику (см. выше).

Если проверка файла прошла успешно, **Dr.Web Daemon** находится в рабочем состоянии.

Обратите внимание, что **Dr.Web Daemon** не может сканировать файлы размером больше 2 гигабайт. Такие файлы не будут отправляться на сканирование клиентами **Dr.Web Daemon**.

При сканировании архивов больших размеров могут возникать ошибки, связанные с истечением времени ожидания. При возникновении таких ошибок увеличьте значения, указанные в параметрах `FileTimeout` и `SocketTimeout`.

Режимы проверки

Dr.Web Daemon имеет два основных режима проверки:

- проверка фрагмента данных, полученного из сокета (**удаленное сканирование**);
- проверка файла на диске (**локальное сканирование**).

При использовании первого режима **Dr.Web Daemon** получает данные для проверки из сокета — фактически, это некоторый фрагмент данных. Данный фрагмент может быть поименованным или нет, что отразится исключительно на форме записи в журнале **Dr.Web Daemon**. Пример работы **Dr.Web Daemon** в этом режиме приведен в предыдущем пункте: клиент читает файл и отсылает его **Dr.Web Daemon** для проверки. **Dr.Web Daemon** может проверять любой фрагмент данных, не обязательно файл.

Более эффективен режим, в котором **Dr.Web Daemon** проверяет указанный файл на диске — локальное сканирование. Клиент сообщает **Dr.Web Daemon** лишь путь к файлу, а не передает весь файл. Путь к проверяемому файлу задается относительно **Dr.Web Daemon** (т.к. клиенты могут находиться на других машинах и т.д.). Этот режим обеспечивает большую производительность и упрощает создание рабочих схем с лечением (например, на файловых серверах).

Режим локального сканирования требует более тщательной настройки прав, т.к. **Dr.Web Daemon** проверяемый файл должен быть доступен на чтение, а в случае почтовых файлов и использования действий `Cure` и `Delete` – необходимы и права на запись.

На это стоит обратить особое внимание при использовании **Dr.Web Daemon** с почтовыми системами, т.к. фильтры, как правило, работают от имени почтовой системы (которая также не использует прав `root`). В наиболее выгодном режиме фильтр создает файл с письмом (получая его от почтовой системы) и сообщает **Dr.Web Daemon** о его местоположении. На этом этапе нужно правильно распределить права на каталог, в котором фильтрами будут создаваться файлы. Можно порекомендовать либо включить в группу почтовой подсистемы пользователя, чьи права используются **Dr.Web Daemon**, либо сразу запускать **Dr.Web Daemon** с правами пользователя, с которыми запускается почтовая система.

В корректно настроенной системе **Dr.Web Daemon** в большинстве случаев не требуется прав администратора (суперпользователя `root`).

При необходимости, имя пользователя, от имени которого должен работать **Dr.Web Daemon**, задается при помощи параметра конфигурации **`User`** в настройках **Dr.Web Daemon**. Кроме того, вы можете настроить пользователя и группу, используемые для запуска модуля, отредактировав соответствующий `mmc`-файл у компонента **Dr.Web Monitor**, если он используется для управления работой компонентов **Dr.Web для почтовых серверов UNIX**.

Обрабатываемые сигналы

Dr.Web Daemon может принимать и обрабатывать следующие сигналы:

- `SIGHUP` — перезагрузка конфигурационного файла;
- `SIGTERM` — корректное завершение работы **Dr.Web Daemon**;
- `SIGKILL` — принудительное завершение работы **Dr.Web Daemon** (в случае проблем);
- `SIGUSR1` — инициирует сохранение в журнал статистики пула процессов.

Обратие внимание, что сигнал `SIGUSR1` должен посылаться только родительскому процессу, поскольку для дочерних процессов `SIGUSR1` приведет к завершению процесса.

Журнал работы и статистика пула процессов

Журнал работы

Поскольку **Dr.Web Daemon** является резидентной программой, информация о его работе может быть получена только из журнала (лога). Журнал содержит подробности обработки каждого запроса на сканирование, полученного **Dr.Web Daemon**. Имя файла журнала указывается в значении параметра конфигурационного файла **`LogFileName`**.

Dr.Web Daemon может выводить данные об обработке запросов на сканирование в разные файлы, в зависимости от клиента, который выслал запрос. В параметре **`ClientsLogs`** конфигурационного файла можно указать отдельные файлы журнала (или назначить службу журналирования **syslog**) для каждого из клиентских приложений **Dr.Web** (например, **Dr.Web для почтовых серверов UNIX**).

Вне зависимости от параметра **`ClientsLogs`**, если клиентское приложение было распознано **Dr.Web Daemon**, результаты сканирования будут отмечены специальным префиксом при выводе в файл журнала. Возможны следующие префиксы:

- `<web>` – **Dr.Web ICAPD**;
- `<smb_spider>` – **Dr.Web Samba SpIDer**;
- `<mail>` – **Dr.Web MailD**;

- <drwebdc> – консольный клиент **Dr.Web Daemon**;
- <kerio> – **Dr.Web для интернет-шлюзов Kerio**;
- <lotus> – **Dr.Web для IBM Lotus Domino**.

В операционной системе **FreeBSD** вывод на консоль **Dr.Web Daemon** может быть перехвачен системной службой **syslog** и выведен в файл отчета посимвольно. Эта проблема проявляется, если в конфигурационном файле службы **syslog** `syslog.conf` установлен уровень подробности журналирования `*.info`.

Статистика пула процессов

Статистика текущего состояния пула процессов, который используется для обработки запросов на сканирование, может быть выведена в файл журнала по получению модулем **Dr.Web Daemon** сигнала `SIGUSR1` (сигнал должен посылаться только родительскому процессу, поскольку для дочерних процессов получение сигнала `SIGUSR1` приведет к завершению процесса). Накоплением статистики по пулу процессов управляет соответствующее значение **`stat`** (`yes` или `no`) в параметре **`ProcessesPool`**. Статистика не суммируется. В каждом случае выводится состояние пула, накопленное между двумя последовательными сохранениями статистики.

Пример вывода записи со статистикой пула процессов:

```
Fri Oct 15 19:47:51 2010 processes pool statistics: min = 1 max = 1024 (auto)
freetime = 121 busy max = 1024 avg = 50.756950 requests for new process = 94
(0.084305 num/sec) creating fails = 0 max processing time = 40000 ms; avg = 118646
ms curr = 0 busy = 0
```

где:

- `min` – минимальное количество процессов в пуле;
- `max` – максимальное количество процессов в пуле;
- `(auto)` – выводится, если ограничения пула процессов определяются автоматически;
- `freetime` – максимальное время бездействия процесса в пуле;
- `busy max` – максимальное количество одновременно занятых процессов, `avg` – среднее количество одновременно занятых процессов;
- `requests for new process` – количество запросов на создание дополнительных процессов (в скобках приводится частота запросов в секунду);
- `creating fails` – количество неудачных попыток создания процесса (обычно, по причине нехватки системных ресурсов);
- `max processing time` – максимальное время обработки одного запроса в миллисекундах;
- `avg` – среднее время обработки одного запроса в миллисекундах;
- `curr` – текущее общее количество процессов в пуле;
- `busy` – текущее количество занятых процессов.

Настройки

Можно запустить **Dr.Web Daemon** с настройками по умолчанию, но предпочтительнее настроить его в соответствии с требованиями и условиям эксплуатации. Конфигурационный файл `drweb32.ini` читается **Dr.Web Daemon** из каталога `%etc_dir`. Для использования другого конфигурационного файла полный путь к нему необходимо указать параметром командной строки при запуске **Dr.Web Daemon**.

Устройство конфигурационного файла и краткое описание правил задания параметров конфигурации приведены в разделе Конфигурационные файлы.

Секция [Daemon]

EnginePath = {путь к файлу}	Расположение модуля **drweb32.dll** (Антивирусное ядро **Dr.Web Engine**). Этот параметр также используется модулем обновления **Dr.Web Updater**. Значение по умолчанию: **EnginePath** = %bin_dir/lib/drweb32.dll
VirusBase = {список масок файлов}	Маски для подключаемых вирусных баз. Этот параметр также используется модулем обновления **Dr.Web Updater**. Допустимо перечисление нескольких масок через запятую. По умолчанию вирусные базы хранятся в файлах с расширением .vdb Значение по умолчанию: **VirusBase** = %var_dir/bases/*.vdb
UpdatePath = {путь к каталогу}	Каталог хранения обновлений. Этот параметр используется модулем обновления **Dr.Web Updater** и должен быть задан обязательно. Значение по умолчанию: **UpdatePath** = %var_dir/updates/
TempPath = {путь к каталогу}	Этот каталог используется Антивирусным ядром **Dr.Web Engine** для создания временных файлов. При нормальной работе каталог практически не используется, он нужен для распаковки некоторых видов архивов или когда в системе не хватает памяти. Значение по умолчанию: **TempPath** = %var_dir/spool/
Key = {список путей к файлам}	Расположение ключевых файлов. По умолчанию ключевой файл имеет расширение .key Ключевой файл может быть различным для **Dr.Web Daemon** и для **Dr.Web Scanner**. Соответственно, при необходимости нужно изменить настройки данного параметра. Параметр может задаваться несколько раз, указывая несколько лицензионных ключевых файлов. В таком случае **Dr.Web Daemon** пытается объединить права, предоставляемые различными лицензиями. Значение по умолчанию: **Key** = %bin_dir/drweb32.key
MailAddressesList = {путь к файлу}	Параметр используется только в случае адресной лицензии на менее чем 50 адресов. В задаваемом параметром файле должен быть задан список адресов (но не более количества, заданного в лицензии), которые будут проверяться (входящая и исходящая корреспонденция). Формат файла – один адрес на строке. Алиасы любого вида считаются отдельными адресами. Значение по умолчанию: **MailAddressesList** = %etc_dir/email.ini

OutputMode = {Terminal | Quiet}	Режим вывода информации при запуске: • Terminal – вывод на консоль, • Quiet – отменяет вывод. Значение по умолчанию: **OutputMode** = Terminal
RunForeground = {логический}	Значение Yes запрещает **Dr.Web Daemon** переходить в режим демона, т.е. становиться фоновым процессом без управляющего терминала. Эта возможность может быть использована некоторыми средствами мониторинга (например, **Dr.Web Monitor**). Значение по умолчанию: **RunForeground** = No
User = {строка}	Пользователь, с правами которого работает **Dr.Web Daemon**. Рекомендуется завести в системе специального пользователя drweb, который будет использоваться **Dr.Web Daemon** и некоторыми фильтрами. Использовать **Dr.Web Daemon** с правами root нежелательно, хотя такое решение значительно проще настраивается. Значение этого параметра не изменяется во время процедуры перечитывания конфигурации «на лету» (обработки сигнала SIGHUP). Значение по умолчанию: **User** = drweb
PidFile = {путь к файлу}	Имя файла, в который при запуске **Dr.Web Daemon** записывается информация об идентификаторе его процесса (pid), а также сокет (если параметр **Socket** задает использование UNIX-сокета) или номер порта (если параметр **Socket** задает использование TCP-сокета). Если задано более одного параметра **Socket**, в данном файле будет присутствовать информация обо всех заданных сокетах (по одному в строке). Значение по умолчанию: **PidFile** = %var_dir/run/drwebd.pid
BusyFile = {путь к файлу}	Данный файл сигнализирует о занятости **Dr.Web Daemon**: он создается сканирующей "копией" **Dr.Web Daemon** при получении команды и уничтожается после передачи результата ее выполнения. Имя файла, создаваемого каждой "копией" **Dr.Web Daemon**, дополняется точкой и ASCII-представлением pid (например, /var/run/drwebd.bsy.123456). Значение по умолчанию: **BusyFile** = %var_dir/run/drwebd.bsy
ProcessesPool = {настройки пула процессов}	Настройки динамического пула процессов. Первым определяется количество процессов в пуле: • auto – количество процессов определяется автоматически в зависимости от загрузки системы; • N – целое неотрицательное число. Как минимум N

	процессов в пуле будут активны, а новые процессы будут создаваться по мере надобности; • `N-M` – целые положительное значения, и `M>=N`. Как минимум `N` процессов в пуле будут активны, а новые процессы будут создаваться по мере надобности, пока число процессов не достигнет значения `M`. Далее определяются дополнительные параметры: • **`timeout`** = `{время в секундах}` – если процесс не становится активным в течение заданного периода времени, процесс закрывается. Этот параметр не влияет на первые `N` процессов (ожидающих запросов бесконечно). • **`stat`** = `{yes|no}` – собирать ли статистику по процессам в пуле. В случае если этот параметр равен `yes`, при получении системного сигнала `SIGUSR1` **Dr.Web Daemon** сохранит текущую накопленную статистику в файл журнала. В противном случае учет и сохранение статистики не производится. • **`stop_timeout`** = `{время в секундах}` – время ожидания остановки работающего процесса. Значение по умолчанию: **`ProcessesPool`** = `auto`,**`timeout`** = `120`, **`stat`** = `no`, **`stop_timeout`** = `1`
`OnlyKey` = `{логический}`	Подключение возможности запросить только ключевой файл от **Dr.Web Agent**, не запрашивая конфигурацию. При этом будет использоваться локальный конфигурационный файл. Если указан адрес сокета **Dr.Web Agent** и значение параметра **`OnlyKey`** установлено в `No`, то **Dr.Web Agent** будет отправляться статистика работы **Dr.Web Daemon** (информация будет отправляться **Dr.Web Agent** после сканирования каждого файла). Значение по умолчанию: **`OnlyKey`** = `No`
`ControlAgent` = `{адрес}`	Адрес сокета **Dr.Web Agent**. **Пример**: **`ControlAgent`** = `inet:4040@127.0.0.1,local:/var/drweb/ipc/.agent` **Dr.Web Daemon** получает через этот сокет от **Dr.Web Agent** лицензионный ключ (и конфигурационный файл, если в качестве значения параметра **`OnlyKey`** задано `No`. Кроме того, в этом случае через этот сокет **Dr.Web Daemon** отправляет **Dr.Web Agent** статистику проверки файлов). Значение по умолчанию: **`ControlAgent`** = `local:%var_dir/ipc/.agent`
`MailCommand` = `{строка}`	Команда shell, вызываемая **Dr.Web Daemon** и модулем обновления **Dr.Web Updater** для отсылки уведомлений пользователю (администратору) по электронной почте. **Dr.Web Daemon** использует этот механизм при каждом запуске (перезапуске, перезагрузке), если до истечения срока действия ключевого файла (одного из ключевых файлов) осталось менее дней, чем указано в параметре **`NotifyPeriod`**. Модуль обновления **Dr.Web Updater** использует этот

	механизм для рассылки пользователям информационных материалов, подготовленных компанией **Dr.Web**, в том числе по вопросам, связанным с обновлениями файлов программы. Значение по умолчанию: **MailCommand** = "/usr/sbin/sendmail -i -bm -f drweb -- root"
NotifyPeriod = {числовое значение}	Значение данного параметра определяет, за сколько дней до окончания срока действия ключевого файла рассылаются уведомления о необходимости продления лицензии. Если указано значение 0, уведомления рассылаются сразу после окончания действия ключа. Значение по умолчанию: **NotifyPeriod** = 14
NotifyFile = {путь к файлу}	Путь к файлу с меткой времени последнего уведомления о продлении лицензии. Значение по умолчанию: **NotifyFile** = %var_dir/.notify
NotifyType = {Ever | Everyday | Once}	Регулярность отправления уведомления о продлении лицензии: • Once – уведомление посылается единожды. • Everyday – уведомление посылается каждый день. • Ever – уведомление посылается при каждой перезагрузке **Dr.Web Daemon** или обновлении баз. Значение по умолчанию: **NotifyType** = Ever
FileTimeout = {числовое значение}	Максимальное разрешенное время проверки одного файла в секундах. Если указано значение 0, время проверки файла не ограничивается. Значение по умолчанию: **FileTimeout** = 30
StopOnFirstInfected = {логический}	Прекращение проверки письма после первого обнаруженного вируса. Установка значения Yes может резко сократить нагрузку на почтовый сервер и время проверки писем. Значение по умолчанию: **StopOnFirstInfected** = No
ScanPriority = {числовое значение}	Приоритет сканирующих процессов **Dr.Web Daemon**. Значение параметра должно быть в диапазоне от высшего значения (-20) до низшего (19 для **Linux**, 20 для остальных ОС). Значение по умолчанию: **ScanPriority** = 0
FilesTypes = {список расширений файлов}	Типы файлов, которые будут проверяться при сканировании по типу, т.е. когда параметр **ScanFiles** (см. ниже) имеет значение ByType.

	Допускаются символы маски '*' и '?'. Значение по умолчанию: **FilesTypes** = EXE, COM, SYS, OV?, BAT, BIN, DRV, PRG, BOO, SCR, CMD, VXD, 386, DLL, FON, DO?, XL?, WIZ, RTF, CL*, HT*, VB*, JS*, INF, AR?, ZIP, R??, PP?, OBJ, LIB, HLP, MD?, INI, MBR, IMG, CSC, CPL, MBP, SHS, SHB, PIF, SO, CHM, REG, XML, PRC, ASP, LSP, MSO, OBD, THE*, NWS, SWF, BMP, MPP, OCX, DVB, CPY, MSG, EML
FilesTypesWarnings = {логический}	Предупреждение о файлах неизвестных типов. Значение по умолчанию: **FilesTypesWarnings** = Yes
ScanFiles = {All | ByType}	Дополнительное ограничение на файлы, подлежащие проверке. При задании значения ByType учитываются расширения файлов, значения которых заданы или по умолчанию, или в параметре (параметрах) **FilesTypes**. Внутри почтовых файлов всегда действует режим All. Значение ByType может быть использовано только в режиме **локального сканирования**. **Важно!** В случае если для антивирусного подключаемого модуля **Drweb** параметр **ScanType** имеет значение local или auto, то включение режима **ScanFiles** = ByType приведет к тому, что этот модуль будет пропускать письма без проверки на вирусы! Значение по умолчанию: **ScanFiles** = All
CheckArchives = {логический}	Проверка файлов, содержащихся в архивах. Поддерживаются архивы форматов ZIP (WinZip, InfoZIP и др.), RAR, ARJ, TAR, GZIP, CAB и др. Значение по умолчанию: **CheckArchives** = Yes
CheckEMailFiles = {логический}	Проверка файлов в почтовых (e-mail) форматах. Значение по умолчанию: **CheckEMailFiles** = Yes
ExcludePaths = {список путей (масок)}	Маски для тех файлов, которые не должны проверяться. Значение по умолчанию: **ExcludePaths** = /proc,/sys,/dev
FollowLinks = {логический}	Следование символическим ссылкам при сканировании. Значение по умолчанию: **FollowLinks** = No
RenameFilesTo = {маска}	Маска для переименования файлов, если сработало действие Rename. Значение по умолчанию: **RenameFilesTo** = #??

MoveFilesTo = {путь к каталогу}	Путь к каталогу **Карантина**. Значение по умолчанию: **MoveFilesTo** = %var_dir/infected/
BackupFilesTo = {путь к каталогу}	Каталог для сохранения зараженных файлов, которые были вылечены. Значение по умолчанию: **BackupFilesTo** = %var_dir/infected/

Параметры регистрации событий:

LogFileName = {syslog | путь к файлу}	Имя файла журнала или syslog, если нужно использовать системный сервис **syslog**. Значение по умолчанию: **LogFileName** = syslog
SyslogFacility = {метка syslog}	Метка записи при использовании системного сервиса **syslog**. Значение по умолчанию: **SyslogFacility** = Daemon
SyslogPriority = {уровень подробности}	Уровень подробности ведения журнала при использовании системного сервиса **syslog**. Допускается использование следующих уровней: • Error • Alert • Warning • Info • Notice Значение по умолчанию: **SyslogPriority** = Info
LimitLog = {логический}	Ограничение размера файла журнала. Игнорируется при использовании системного сервиса **syslog**. Ограничение размера файла журнала реализуется следующим образом: при запуске или получении сигнала HUP **Dr.Web Daemon** проверяет размер файла журнала, и если он превышает значение, заданное в параметре **MaxLogSize**, файл журнала стирается и ведение журнала начинается с нуля. Значение по умолчанию: **LimitLog** = No
MaxLogSize = {числовое значение}	Максимальный размер файла журнала в килобайтах. Имеет смысл только если не используется **syslog** и **LimitLog** = Yes. Если указано значение 0, размер файла журнала проверяться не будет. Значение по умолчанию: **MaxLogSize** = 512
LogScanned = {логический}	Вывод в файл журнала информации обо всех проверяемых объектах, независимо от того, обнаружены вирусы или нет.

	Значение по умолчанию: **LogScanned** = Yes
LogPacked = {логический}	Вывод в файл журнала дополнительной информации о файлах, упакованных утилитами DIET, PKLITE и др. Значение по умолчанию: **LogPacked** = Yes
LogArchived = {логический}	Вывод в файл журнала дополнительной информации об архиваторах. Значение по умолчанию: **LogArchived** = Yes
LogTime = {логический}	Вывод в файл журнала времени каждой записи. Параметр не имеет смысла при использовании системного сервиса **syslog** Значение по умолчанию: **LogTime** = Yes
LogProcessInfo = {логический}	Вывод в файл журнала перед каждой записью данных о pid сканирующего процесса и адресе фильтра (имени хоста или IP-адресе), с которого инициирована проверка. Значение по умолчанию: **LogProcessInfo** = Yes
RecodeNonprintable = {логический}	Перекодировка при выводе в файл журнала символов, не являющихся отображаемыми для данного терминала (см. следующие два параметра). Значение по умолчанию: **RecodeNonprintable** = Yes
RecodeMode = {Replace | QuotedPrintable}	При **RecodeNonprintable** = Yes задает метод перекодировки неотображаемых символов. При **RecodeMode** = Replace все такие символы заменяются на значение параметра **RecodeChar** (см. ниже). При **RecodeMode** = QuotedPrintable производится перекодировка неотображаемых символов в формат Quoted Printable. Значение по умолчанию: **RecodeMode** = QuotedPrintable
RecodeChar = {"?" | "_" | ...}	При **RecodeMode** = Replace задает символ, на который будут заменены все неотображаемые символы. Значение по умолчанию: **RecodeChar** = "?"
Socket = {список адресов}	Описание сокета, который будет использован для связи с **Dr.Web Daemon**. **Пример**: **Socket** = inet:3000@127.0.0.1,local:%var_dir/.daemon Также можно адрес каждого из сокетов указывать в отдельном параметре в формате ПОРТ [интерфейсы] | ФАЙЛ

	[доступ]. Соответственно, для TCP-сокета: ПОРТ - десятичный номер порта, интерфейсы - список имен интерфейсов или IP-адресов, на которых **Dr.Web Daemon** будет принимать запросы. **Пример**: **Socket** = 3000 127.0.0.1, 192.168.0.100 Для UNIX-сокета: ФАЙЛ - имя сокета, доступ - восьмеричное значение прав доступа к нему. **Пример**: **Socket** = %var_dir/.daemon 0660 Количество значений в списке **Socket** не ограничено, **Dr.Web Daemon** будет работать со всеми из описанных сокетов. Чтобы **Dr.Web Daemon** принимал запросы через все доступные интерфейсы, для параметра следует задать значение 3000 0.0.0.0. Значение по умолчанию: **Socket** = %var_dir/run/.daemon
SocketTimeout = {числовое значение}	Время в секундах, отведенное для приема/передачи всех данных через сокет (время сканирования файла не учитывается). Если указано значение 0, время не будет ограничено. Значение по умолчанию: **SocketTimeout** = 10

Следующие параметры могут быть использованы для уменьшения времени проверки архивов (за счет отказа от проверки некоторых объектов в архиве). Если объект подпадает под ограничения, созданные этими параметрами, то к нему применяется действие **ArchiveRestriction**, которое задано в файлах конфигурации различных фильтров.

MaxCompressionRatio = {числовое значение}	Максимальный коэффициент сжатия, т.е. отношение длины файла в распакованном виде к длине файла в запакованном виде (внутри архива). Если коэффициент превышает данное значение, файл не будет извлечен и, соответственно, не будет проверен. Параметр может принимать только натуральные значения и не может быть меньше 2. Значение по умолчанию: **MaxCompressionRatio** = 5000
CompressionCheckThreshold = {числовое значение}	Минимальный размер файла внутри архива в килобайтах, начиная с которого будет производиться проверка коэффициента сжатия (если это предписано параметром **MaxCompressionRatio**). Должен быть указан размер больше 0. Значение по умолчанию: **CompressionCheckThreshold** = 1024
MaxFileSizeToExtract = {числовое значение}	Максимальный размер файла в килобайтах, извлекаемого из архива. Если размер файла внутри архива превышает это значение, он будет пропущен. Значение по умолчанию: **MaxFileSizeToExtract** = 40960

MaxArchiveLevel = {числовое значение}	Максимальный уровень вложенности архивов (когда архив вложен в архив, который тоже вложен в архив и т.д.). При превышении этого уровня архив будет пропущен (не будет проверен). Значение по умолчанию: **MaxArchiveLevel** = 8
ClientsLogs = {список строк}	Параметр разделения файлов журнала. Если при обращении к **Dr.Web Daemon** клиент передает в расширенных опциях свой идентификатор, файл журнала клиента заменяется на тот, который указан в параметре ClientsLogs. Описания логов разделяются запятыми или пробелами. В случае задания в параметре больше шести файлов журнала строка конфигурационного файла считается неверной. Файлы отчета клиентов задаются в виде: ClientsLogs=<имя клиента1>:<путь к файлу>,<имя клиента2>:<путь к файлу>. Имя клиента может быть одним из следующих: • web — **Dr.Web ICAPD**; • smb_spider — **Dr.Web Samba SpIDer**; • mail — **Dr.Web MailD**; • drwebdc — консольный клиент **Демона Dr.Web**; • kerio — **Dr.Web для интернет-шлюзов Kerio**; • lotus — **Dr.Web для IBM Lotus Domino**. **Пример**: **drwebdc:**/var/drweb/log/drwebdc.log, **smb:**syslog, **mail:**/var/drweb/log/drwebmail.log Значение по умолчанию: **ClientsLogs** =
MaxBasesObsolescencePeriod = {числовое значение}	Максимальный период времени (в часах) с момента последнего обновления, в течение которого вирусные базы считаются "свежими". По истечении этого времени, в консоли выводится уведомление о том, что базы устарели. Если установлено значение 0, то актуальность вирусных баз не проверяется. Значение по умолчанию: **MaxBasesObsolescencePeriod =** 24
MessagePatternFileName = {путь к файлу}	Путь к файлу шаблона сообщения об истечении срока действия лицензии. Позволяет пользователю определить сообщение об истечении срока действия лицензии в удобном для него виде. В шаблоне сообщения могут быть использованы следующие переменные, вместо которых будут автоматически подставлены следующие значения: • $EXPIRATIONDAYS — количество дней до истечения срока лицензии;

	• $KEYFILENAME — путь к лицензионному ключевому файлу; • $KEYNUMBER — номер лицензии; • $KEYACTIVATES — дата активации лицензии; • $KEYEXPIRES — дата завершения срока действия лицензии. Если пользовательский шаблон отсутствует, используется сообщение по умолчанию на английском языке.
	Значение по умолчанию: **MessagePatternFileName =** %etc_dir/templates/drwebd/msg.tmpl
MailTo = {адрес электронной почты}	Почтовый адрес администратора для отправки сообщений об истечении срока действия лицензии, устаревании вирусных баз и пр.
	Значение по умолчанию: **MailTo =**

Dr.Web Agent

Компонент **Dr.Web Agent** представлен модулем **drweb-agent**. Это постоянно загруженный модуль, который управляет настройками модулей программного комплекса **Dr.Web для почтовых серверов UNIX**, определяет политику работы комплекса в зависимости от установленной лицензии и собирает статистику вирусных инцидентов. Эта статистика, в зависимости от режима работы **Dr.Web Agent**, отсылается с заданной периодичностью либо на публичный сервер статистики компании **Dr.Web**, либо на сервер централизованной защиты, под управлением которого работает **Dr.Web Agent**. Когда происходит запуск компонентов **Dr.Web для почтовых серверов UNIX**, или происходит изменение настроек, **Dr.Web Agent** шлет компонентам необходимые настройки.

Обратите внимание, что модуль **drweb-agent** в режиме централизованной защиты (**enterprise mode**) предназначен для работы только с **Dr.Web ESS** версии 6. Если вы хотите обеспечить подключение к серверу централизованной защиты **Dr.Web ESS** версии 10, вам следует установить и настроить новую версию агента, реализованную в виде модуля **drweb-agent10**. Об установке и настройке версии **drweb-agent10** см. в разделе Переход на использование Dr.Web ESS версии 10.

В ходе работы **Dr.Web Agent** может взаимодействовать с другими модулями программного комплекса, обмениваясь с ними различными управляющими сигналами.

Поскольку все компоненты **Dr.Web для почтовых серверов UNIX** (кроме **Dr.Web Monitor**) получают свои конфигурационные данные через модуль **drweb-agent**, он должен запускаться перед другими компонентами, непосредственно после **Dr.Web Monitor**.

Пожалуйста, обратите внимание, что если в конфигурационном файле компонента указано несколько параметров с одним именем, то **Dr.Web Agent** их объединяет через запятую. При задании значений параметров в конфигурационных файлах можно использовать обратный слэш "\". В этом случае **Dr.Web Agent** объединит в одну строку все строки, разделённые с помощью обратного слэша. Обратите внимание, что использование пробела после символа слэша не допускается.

Это может оказаться важным при задании правил обработки писем – вместо того, чтобы писать одно большое правило, можно разбить его на несколько отдельных правил.

Пример:

```
GlobalRules = select message, append_html "lookup:file:/maild-files/
somehtml.html"
```

Это правило можно также задать следующим образом (см. формат правил в описании работы подключаемого модуля **Dr.Web Modifier**):

```
GlobalRules = select message
GlobalRules = append_html "lookup:file:/maild-files/somehtml.html"
```

Обратите внимание, что разделение одной строки на части при помощи слэшэй позволяет вставлять комментарии, которые будут проигнорированы анализатором правила после чтения конфигурации.

Пример:

```
to:user@host cont \
modifier/LocalRules=select mime.headers "X-Spam-Level" "\\*\\*\\*", \
# 3 и более звездочек <- комментарий
if found,\
select mime.headers Subject ".*",\
replace "[SPAM]" "^",\
endif
```

(см. формат правил в секции `[Rules]` конфигурационного файла **Dr.Web MailD**).

Режимы работы

При необходимости продукты компании **«Доктор Веб»** могут быть подключены к корпоративной или частной **Антивирусной сети**, управляемой комплексом **Dr.Web Enterprise Security Suite** (далее **Dr.Web ESS**). Работа в режиме централизованной защиты не требует установки дополнительного программного обеспечения или удаления **Dr.Web для почтовых серверов UNIX**.

Для обеспечения этой возможности, **Dr.Web Agent** может работать в одном из двух режимов:

- Одиночном (**standalone mode**) режиме, когда защищаемый компьютер не включен в **Антивирусную сеть** и управляется локально. В этом режиме конфигурационные и ключевые файлы находятся на локальных дисках, а **Dr.Web Agent** полностью управляется с защищаемого компьютера. Статистика вирусных инцидентов отсылается на сервер статистики компании **«Доктор Веб»**.
- Режиме централизованной защиты (**enterprise mode**), когда защитой компьютера управляет сервер централизованной защиты. В этом режиме некоторые функции и настройки **Dr.Web для почтовых серверов UNIX** могут быть изменены или заблокированы в соответствии с общей (корпоративной) стратегией антивирусной защиты. В этом режиме используется лицензионный ключевой файл с сервера централизованной защиты. Персональный лицензионный ключевой файл на локальном компьютере не используется. Статистика вирусных инцидентов отсылается на управляющий сервер централизованной защиты.

Обратите внимание, что модуль **drweb-agent** в режиме централизованной защиты (**enterprise mode**) предназначен для работы только с **Dr.Web ESS** версии 6. Если вы хотите обеспечить подключение к серверу централизованной защиты **Dr.Web ESS** версии 10, вам следует установить и настроить новую версию агента, реализованную в виде модуля **drweb-agent10**. Об установке и настройке версии **drweb-agent10** см. в разделе Переход на использование Dr.Web ESS версии 10.

Чтобы использовать режим централизованной защиты:

1. Свяжитесь с системным администратором вашей сети, чтобы получить файл с открытым ключом и параметры соединения с сервером централизованной защиты.
2. В конфигурационном файле **Dr.Web Agent** (по умолчанию `%etc_dir/agent.conf`) установите значения следующих параметров в секции `[EnterpriseMode]` :
   - Укажите путь к файлу с открытым ключом, полученному от администратора сети, в параметре **PublicKeyFile** (обычно `%var_dir/drwcsd.pub`). Этот файл содержит открытый ключ, используемый для зашифрованного соединения с сервером **Dr.Web ESS** (далее – **Dr.Web Enterprise Server**). Если вы – администратор сети, то вы можете найти этот файл в соответствующем каталоге на **Dr.Web Enterprise Server**.
   - Укажите IP-адрес или имя узла **Dr.Web Enterprise Server** в параметре **ServerHost**.
   - Укажите номер порта для связи с **Dr.Web Enterprise Server** параметре **ServerPort**.
3. Чтобы включить режим централизованной защиты, установите `Yes` в качестве значения параметра **UseEnterpriseMode**.

В режиме централизованной защиты некоторые функции и настройки **Dr.Web для почтовых серверов UNIX** могут быть изменены или заблокированы в соответствии с общей (корпоративной) стратегией антивирусной защиты. В этом режиме используется лицензионный ключевой файл с сервера централизованной защиты. Персональный лицензионный ключевой файл на локальном компьютере не используется.

Для работы **Dr.Web Agent** в режиме централизованной защиты должен быть установлен пакет `drweb-agent-es`.

Чтобы **Dr.Web для почтовых серверов UNIX** полностью поддерживал режим централизованной защиты, **Dr.Web Monitor** также должен работать в режиме централизованной защиты. Для подробностей обратитесь к разделу Режимы работы **Dr.Web Monitor**.

Чтобы настройки модулей интеграции с MTA `drweb-courier`, `drweb-cgp-receiver` вступили в силу после обновления на сервере централизованной защиты, требуется вручную отправить им сигнал `SIGHUP` или `STOP-START`.

Чтобы использовать одиночный (standalone) режим

1. Убедитесь, что все параметры в секции `[StandaloneMode]` конфигурационного файла **Dr.Web Agent** (по умолчанию, `%etc_dir/agent.conf`) установлены корректно.
2. Установите `No` в качестве значения параметра **`UseEnterpriseMode`** секции `[EnterpriseMode]` конфигурационного файла **Dr.Web Agent**.

При включении этого режима все настройки **Dr.Web для почтовых серверов UNIX** будут разблокированы и вы вновь получите доступ ко всем функциям и настройкам **Dr.Web для почтовых серверов UNIX**.

Для работы в одиночном режиме **Dr.Web для почтовых серверов UNIX** необходим действующий лицензионный ключ. Ключевые файлы, полученные с сервера централизованной защиты, не могут быть использованы в этом режиме.

Совместное использование Dr.Web для почтовых серверов UNIX и Антивируса Dr.Web для Linux в режиме централизованной защиты

Ввиду особенностей реализации, одновременное использование в режиме централизованной защиты **Dr.Web для почтовых серверов UNIX** и **Антивируса Dr.Web для Linux**, установленных на одном компьютере, невозможно. Для включения режима централизованной защиты **Dr.Web для почтовых серверов UNIX** необходимо перевести **Антивирус Dr.Web для Linux** в режим автономной работы, после чего удалить или переместить в другой каталог файлы `%etc_dir/agent/drweb-cc.amc` и `%etc_dir/agent/drweb-spider.amc`.

Рекомендуется сохранить эти файлы в качестве резервной копии в каталоге, отличном от `%etc_dir/agent`, если в дальнейшем вы планируете перевести **Антивирус Dr.Web для Linux** в режим централизованной защиты. В таком случае, отключите режим централизованной защиты **Dr.Web для почтовых серверов UNIX**, копируйте резервные копии файлов `drweb-cc.amc` и `drweb-spider.amc` в каталог `%etc_dir/agent/` и следуйте инструкциям, представленным в руководстве пользователя **Антивируса Dr.Web для Linux**.

Параметры командной строки

Для запуска **Dr.Web Agent** используется следующая команда:

```
drweb-agent [параметры]
```

Dr.Web Agent допускает использование следующих параметров:

Краткий вариант	Расширенный вариант	Аргументы
-h	--help	
Описание: Вывод на экран краткой справки по имеющимся параметрам командной строки и завершение работы модуля		
-v	--version	
Описание: Вывод на экран информации о текущей версии **Dr.Web Agent** и завершение работы модуля		
-u	--update-all	
Описание: Запуск процесса обновления для всех компонентов **Dr.Web для почтовых серверов UNIX**		
-f	--update-failed	
Описание: Запуск процесса обновления для тех компонентов **Dr.Web для почтовых серверов UNIX**, которые не удалось обновить в штатном режиме		
-C	--check-only	
Описание: Проверка корректности конфигурации модуля **Dr.Web Agent**. Данный параметр командной строки не может быть использован при наличии в системе уже запущенного экземпляра **Dr.Web Agent**		
-c	--conf	<путь к файлу>
Описание: Использование при запуске указанного конфигурационного файла		
-d	--droppwd	
Описание: Сбросить регистрационную информацию (имя пользователя и пароль), используемую **Dr.Web Agent** для доступа к **Dr.Web Enterprise Server**. При следующей попытке соединения с **Dr.Web Enterprise Server** будет запущен процесс регистрации новой станции		
-p	--newpwd	
Описание: Смена имени пользователя и пароля на используемом сервере централизованной защиты **Dr.Web Enterprise Server**		
-s	--socket	<путь к файлу>
Описание: Использование компонентом для коммуникации с управляемыми модулями сокета, указанного в аргументе		
-P	--pid-file	<путь к файлу>
Описание: Использование в качестве PID-файла **Dr.Web Agent** файла, указанного в аргументе		
-e	--export-config	<имя приложения>

Краткий вариант	Расширенный вариант	Аргументы
Описание: Экспорт конфигурации приложения, имя которого указано в аргументе, на **Dr.Web Enterprise Server**. В качестве аргумента следует использовать имя приложения, указанное в заголовке секции `Application "<имя приложения>"` соответствующего `amc`-файла (см. раздел Взаимодействие с компонентами программного комплекса). Данный параметр командной строки не может быть использован при наличии в системе уже запущенного экземпляра **Dr.Web Agent**. Также он не может быть использован для экспорта конфигурации **Антивируса Dr.Web для Linux**		

Конфигурационный файл

Настройки компонента **Dr.Web Agent** задаются отдельным конфигурационным файлом `%etc_dir/agent.conf`.

Общие принципы устройства конфигурационных файлов компонентов **Dr.Web для почтовых серверов UNIX** и краткое описание правил задания параметров конфигурации приведены в разделе Конфигурационные-файлы.

Секция [Logging]

В секции `[Logging]` собраны параметры, управляющие ведением журналов работы компонента **Dr.Web Agent** программного комплекса **Dr.Web для почтовых серверов UNIX:**

Секция [Logging]

Level = {уровень подробности}	Уровень подробности сохранения в журнал работы компонента общих событий. Допускается использование следующих уровней: • Quiet • Error • Alert • Info • Debug Значение по умолчанию: **Level** = Info
IPCLevel = {уровень подробности}	Уровень подробности сохранения в журнал работы компонента событий подсистемы IPC. Допускается использование следующих уровней: • Quiet • Error • Alert • Info • Debug Значение по умолчанию: **IPCLevel** = Error
SyslogFacility = {метка syslog}	Метка записи при использовании системного сервиса **syslog** Значение по умолчанию: **SyslogFacility** = Daemon
FileName = {syslog | путь к файлу}	Имя файла журнала или `syslog`, если нужно использовать системный сервис **syslog**

	Значение по умолчанию: **FileName** = syslog

Секция [Agent]

В секции [Agent] собраны основные настройки компонента **Dr.Web Agent**:

Секция [Agent]

MetaConfigDir = {путь к каталогу}	Расположение файлов мета-конфигурации **Dr.Web Agent**. В файлах мета-конфигурации описываются особенности взаимодействия **Dr.Web Agent** с другими модулями программного комплекса. Содержание файлов мета-конфигурации задается разработчиками **«Доктор Веб»** и не требует редактирования. Значение по умолчанию: **MetaConfigDir** = %etc_dir/agent/
UseMonitor = {логический}	Значение Yes данного параметра, указывает **Dr.Web Agent**, что в составе программного комплекса используется **Dr.Web Monitor**. Значение по умолчанию: **UseMonitor** = Yes
MonitorAddress = {адрес}	Сокет, через который **Dr.Web Agent** взаимодействует с **Dr.Web Monitor** (значение параметра должно совпадать со значением параметра **Address** конфигурационного файла **Dr.Web Monitor**). Значение по умолчанию: **MonitorAddress** = local:%var_dir/ipc/.monitor
MonitorResponseTime = {числовое значение}	Максимальное время отклика **Dr.Web Monitor** в секундах. Если в течение этого времени от **Dr.Web Monitor** не поступает реакции, то предполагается, что он не запущен, и **Dr.Web Agent** больше не предпринимает попыток взаимодействия с **Dr.Web Monitor**. Значение по умолчанию: **MonitorResponseTime** = 5
PidFile = {путь к файлу}	Путь к файлу, в который записывается PID исполняемого модуля drweb-agent при запуске. Значение по умолчанию: **PidFile** = %var_dir/run/drweb-agent.pid

Секция [Server]

В этой секции располагаются параметры, управляющие взаимодействием **Dr.Web Agent** с другими модулями программного комплекса **Dr.Web для почтовых серверов UNIX:**

Секция [Server]

Address = {адрес}	Сокет, через который **Dr.Web Agent** взаимодействует с другими модулями программного комплекса. Допускается несколько сокетов, перечисленных через запятую.

	Значение по умолчанию: **Address** = local:%var_dir/ipc/.agent, inet:4040@127.0.0.1
Threads = {числовое значение}	Количество одновременных потоков drweb-agent. Параметр управляет максимальным количеством одновременных подключений к модулям, передающим **Dr.Web Agent** вирусную статистику. Этот параметр не может быть изменен при перезапуске по сигналу SIGHUP. Если указано значение 0, количество одновременных потоков не ограничивается (не рекомендуется). Значение по умолчанию: **Threads** = 2
Timeout = {числовое значение}	Максимальное время (в секундах) установления соединения между **Dr.Web Agent** и другими компонентами программного комплекса. Если указано значение 0, время установления соединения не ограничивается. Значение по умолчанию: **Timeout** = 15

Секция [EnterpriseMode]

В этой секции расположены параметры, управляющие работой **Dr.Web Agent** в режиме **Enterprise:**

Секция [EnterpriseMode]

UseEnterpriseMode = {логический}	При значении Yes данного параметра **Dr.Web Agent** работает в режиме **Enterprise**, при значении No – в режиме **Standalone**. Значение по умолчанию: **UseEnterpriseMode** = No
ComputerName = {текст}	Название этого компьютера в **Антивирусной сети**. Значение по умолчанию: **ComputerName** =
VirusbaseDir = {путь к каталогу}	Путь к каталогу вирусных баз. Значение по умолчанию: **VirusbaseDir** = %var_dir/bases
PublicKeyFile = {путь к файлу}	Путь к файлу открытого ключа для доступа к **Dr.Web Enterprise Server**. Значение по умолчанию: **PublicKeyFile** = %bin_dir/drwcsd.pub
ServerHost = {IP-адрес}	IP-адрес **Dr.Web Enterprise Server**. Значение по умолчанию: **ServerHost** = 127.0.0.1
ServerPort =	Номер порта доступа к **Dr.Web Enterprise Server**.

	Значение по умолчанию: **ServerPort** = 2193
CryptTraffic = {Yes | Possible | No}	Шифрование трафика, передаваемого между **Dr.Web Enterprise Server** и **Dr.Web Agent**: • Yes – обязательно шифровать • Possible – если возможно • No – не шифровать Значение по умолчанию: **CryptTraffic** = possible
CompressTraffic = {Yes | Possible | No}	Сжатие трафика, передаваемого между **Dr.Web Enterprise Server** и **Dr.Web Agent**: • Yes – обязательно сжимать • Possible – если возможно • No – не сжимать Значение по умолчанию: **CompressTraffic** = possible
CacheDir = {путь к каталогу}	Путь к каталогу, в котором хранятся служебные файлы: конфигурационные файлы компонентов и файлы, содержащие информацию о правах каждого из приложений, на случай, если **Dr.Web Enterprise Server** по какой-либо причине окажется недоступен, файлы с регистрационной информацией на **Dr.Web Enterprise Server** и т.п. Значение по умолчанию: **CacheDir** = %var_dir/agent

Секция [StandaloneMode]

Настройки **Dr.Web Agent** для одиночного режима работы.

Секция [StandaloneMode]

StatisticsServer = {текст}	URL сервера вирусной статистики. Если URL сервера не указан, то статистика не будет отправляться. Значение по умолчанию: **StatisticsServer** = stat.drweb.com:80/update
StatisticsUpdatePeriod = {числовое значение}	Период обновления статистической информации в минутах. Не может быть меньше 5. Значение по умолчанию: **StatisticsUpdatePeriod** = 10
StatisticsProxy = {IP-адрес | имя хоста}	IP-адрес или имя хоста прокси-сервера для вирусной статистики. Обратите вниманиме, что если значение параметра не задано, используется значение переменной окружения http_proxy. **Пример**: **StatisticsProxy** = localhost:3128 Значение по умолчанию: **StatisticsProxy** =

StatisticsProxyAuth = {текст}	Строка аутентификации (<имя пользователя>:<пароль>) для доступа к прокси-серверу. **Пример**: **StatisticsProxyAuth** = test:testpwd Значение по умолчанию: **StatisticsProxyAuth** =
UUID = {текст}	Личный идентификатор пользователя на сервере статистики http://stat.drweb.com/. Данный параметр является обязательным для передачи статистики — соответственно, если вы желаете подключить эту возможность, вы должны указать в его значении персональный UUID (в качестве которого обычно используется md5-сумма лицензионного ключевого файла). Значение по умолчанию: **UUID** =
LicenseFile = {список путей к файлам}	Расположение ключевых файлов программного комплекса **Dr.Web для почтовых серверов UNIX** (лицензионных или демонстрационных). Пути в списке разделяются запятой Значение по умолчанию: **LicenseFile** = %bin_dir/drweb32.key

Секция [Update]

В этой секции собраны параметры, относящиеся к процессу обновления компонентов программного комплекса **Dr.Web для почтовых серверов UNIX** через **Dr.Web Enterprise Server** (подробнее см. в Руководстве администратора антивирусной сети **Dr.Web ESS**):

Секция [Update]

CacheDir = {путь к каталогу}	Каталог, в котором **Dr.Web Agent** временно сохраняет загруженные файлы обновлений. Значение по умолчанию: **CacheDir** = %var_dir/updates/cache
Timeout = {числовое значение}	Максимальное время обработки **Dr.Web Agent** полученных обновлений в секундах. Если указано значение 0, время обработки не ограничивается. Значение по умолчанию: **Timeout** = 120
RootDir = {путь к каталогу}	Путь к корневому каталогу. Значение по умолчанию: **RootDir** = /

Запуск

Обратите внимание, что в процессе работы установочного скрипта при выборе соответствующей возможности в диалоге все сервисы, включая **Dr.Web Agent**, будут запущены автоматически.

В процессе запуска **Dr.Web Agent** при установках по умолчанию осуществляются следующие действия:

- производится поиск и загрузка конфигурационного файла; если файл не найден, то загрузка прекращается;
- если в файле конфигурации заданы параметры секции `[EnterpriseMode]` (и программный комплекс **Dr.Web для почтовых серверов UNIX** работает в составе **Антивирусной сети**), **Dr.Web Agent** запускается в режиме **Enterprise**. В противном случае, если в файле настроек заданы параметры секции `[Standalone]`, **Dr.Web Agent** запускается в одиночном режиме. Если параметры секции `[Standalone]` также не заданы, то загрузка **Dr.Web Agent** прекращается;
- создается сокет для взаимодействия с другими модулями программного комплекса. В случае TCP-соединения подключений может быть несколько (загрузка продолжается, если удалось создать хотя бы одно из них). Если используется UNIX-сокет, то он может быть создан только тогда, когда каталог, содержащий его, доступен на запись и чтение пользователю, с чьими правами работает модуль **`drweb-agent`**. Если ни один сокет не может быть создан, загрузка **Dr.Web Agent** прекращается.

Дальнейший процесс загрузки **Dr.Web Agent** зависит от того, в каком режиме он работает.

Если **Dr.Web Agent** работает в режиме **Enterprise**:

- производится соединение с **Dr.Web Enterprise Server**, используемым в **Антивирусной сети**. Если при первом подключении сервер недоступен, либо **Dr.Web Agent** не удалось авторизоваться, **Dr.Web Agent** завершает свою работу. Если ранее **Dr.Web Agent** уже работал с данным сервером, но в данный момент он недоступен (например, в случае проблем с соединением), **Dr.Web Agent** использует резервные копии конфигурационных файлов, полученных от сервера во время предыдущего подключения. Данные файлы зашифрованы и не предназначены для правки пользователем. Попытка изменить их вручную приведёт к их неработоспособности;
- если соединение успешно установлено, происходит получение лицензионных ключей и настроек компонентов программного комплекса с сервера централизованной защиты. После завершения этой операции **Dr.Web Agent** готов к работе.

Если **Dr.Web Agent** работает в режиме **Standalone**:

- загружаются файлы мета-конфигурации компонентов программного комплекса (.amc). В файлах мета-конфигурации описываются особенности взаимодействия **Dr.Web Agent** с компонентами. Расположение файлов мета-конфигурации берется из параметра **`MetaConfigDir`** секции настроек `[Agent]` файла конфигурации **Dr.Web Agent**. После завершения этой операции **Dr.Web Agent** готов к работе.

Взаимодействие с компонентами программного комплекса

Взаимодействие с компонентами программного комплекса осуществляется с помощью файлов мета-конфигурации (`.amc`). В этих файлах описывается конфигурация компонентов и параметры, значения которых **Dr.Web Agent** выдает компонентам. Эти файлы располагаются в каталоге, определяемом параметром **`MetaConfDir`** (по умолчанию – `%etc_dir/agent`). Как правило, в одном файле указывается описание конфигурации и параметров одного компонента, а имя файла совпадает с именем компонента **Dr.Web для почтовых серверов UNIX**.

Описание каждого компонента содержится в секции `Application "имя_компонента"`. В конце секции обязательно должно быть поставлено `EndApplication`. В описании компонента должны присутствовать следующие параметры:

- **`id`**: идентификатор компонента на используемом **Dr.Web Enterprise Server**;
- **`ConfFile`**: путь к конфигурационному файлу компонента;

- **Components**: описание компонентов. В конце описания ставится **EndComponents**. Для каждого из компонентов указываются: его название и через пробел — список секций конфигурационного файла и параметров в них, которые требуются компоненту для нормальной работы. Секции и параметры перечисляются через запятую.
  Для описания параметров необходимо указывать полный путь к ним (например, `/Quarantine/DBISettings`), а для описания секций достаточно указания имени секции (например, `General`).
  Символ обратного слэша "\" используется для экранирования переводов строки.
  Если компоненту нужны все настройки из конфигурационного файла, достаточно указать вместо перечня секций и/или параметров путь "`/*`".

Пример amc-файла Dr.Web MailD для Linux:

```
Application "MAILD"
 id       40
 ConfFile    "/etc/drweb/maild_smtp.conf"
 Components
 lookup_ldap  LDAP
 lookup_regex  REGEX
 drweb-maild  General, Logging, MailBase, Stat, Maild, Filters,
         Quarantine, /_Rules=Rule*:Rules, /Reports/Send,
         /Reports/SendTimes, /Reports/Names, /Reports/MaxPoolSize,
         /Reports/MaxStoreInDbPeriod,Reports/CheckForRemovePeriod,
         /Notifier/FilterMail, /Notifier/NotifyLangs,
         /Notifier/LngBaseDir
 drweb-notifier General, Logging, Notifier, /Sender/Method, /_Rules,
         Reports, /Filters/BeforeQueueFilters,
         /Filters/AfterQueueFilters, /Quarantine/AccessByEmail,
         /Quarantine/StoredTime
 drweb-sender  General, Logging, Sender
 drweb-receiver General, Logging, /Maild/ProtectedNetworks,
         /Maild/ProtectedDomains, /Maild/IncludeSubdomains,
         SASL, Receiver
 EndComponents
EndApplication
```

Интеграция с Dr.Web Enterprise Security Suite

Возможны следующие ситуации, в которых требуется интегрировать программный комплекс **Dr.Web для почтовых серверов UNIX** с **Антивирусной сетью** под управлением **Dr.Web ESS**:

- первоначальная установка и настройка **Dr.Web для почтовых серверов UNIX** в уже работающей **Антивирусной сети** под управлением **Dr.Web ESS**;
- встраивание работающего UNIX-сервера с установленным и настроенным программным комплексом **Dr.Web для почтовых серверов UNIX** в **Антивирусную сеть** под управлением **Dr.Web ESS**.

Для того, чтобы **Dr.Web для почтовых серверов UNIX** мог работать в составе **Антивирусной сети** под управлением **Dr.Web ESS**, необходимо настроить компоненты **Dr.Web Agent** и **Dr.Web Monitor** для работы в режиме **Enterprise** и зарегистрировать комплекс на сервере централизованной защиты **Dr.Web Enterprise Server**.

В соответствии с политикой подключения новых станций (подробнее см. Руководство администратора Антивирусной сети **Dr.Web ESS**), подключить **Dr.Web для почтовых серверов UNIX** к **Dr.Web Enterprise Server** можно двумя способами:

- создав учетную запись на сервере автоматически;
- создав учетную запись на сервере вручную.

Настройка компонентов для работы в режиме Enterprise

После установки для запуска в режиме **Enterprise** необходимо вручную внести изменения в локальные конфигурационные файлы **Dr.Web Agent** и **Dr.Web Monitor**.

Для Dr.Web Agent

В секции `[EnterpriseMode]` конфигурационного файла **Dr.Web Agent** `%etc_dir/agent.conf` установите следующие значения параметров:

- **`UseEnterpriseMode`** = `Yes`;
- **`PublicKeyFile`** = `%var_dir/drwcsd.pub` (открытый ключ шифрования для доступа к **Dr.Web Enterprise Server**. Администратор должен самостоятельно взять данный файл из соответствующего каталога **Dr.Web Enterprise Server** и разместить его по указанному пути);
- **`ServerHost`** = IP-адрес или имя хоста **Dr.Web Enterprise Server**;
- **`ServerPort`** = порт **Dr.Web Enterprise Server** (`2193` по умолчанию).

Для Dr.Web Monitor

В секции `[Monitor]` конфигурационного файла **Dr.Web Monitor** `%etc_dir/monitor.conf` установите следующие значения параметров:

- **`UseEnterpriseMode`** = `Yes`.

Автоматическое создание учетной записи

При автоматическом создании учетной записи:

- при первом запуске в режиме **Enterprise** **Dr.Web Agent** запрашивает регистрационные данные (идентификатор станции и пароль) у **Dr.Web Enterprise Server**;
- если на **Dr.Web Enterprise Server** установлен режим "**Ручное подтверждение доступа**" (режим по умолчанию, см. Руководство администратора Антивирусной сети **Dr.Web ESS**), то администратору в течение одной минуты с момента запроса необходимо подтвердить регистрацию новой станции через веб-интерфейс **Центра управления Dr.Web**;
- после первого подключения **Dr.Web Agent** записывает хэш идентификатора станции и пароля пользователя в файл с названием `pwd`. Данный файл создаётся в каталоге, заданном значением параметра **`CacheDir`** секции `[EnterpriseMode]` (по умолчанию `%var_dir/agent/`);
- в дальнейшем данные из этого файла используются для подключения программного комплекса **Dr.Web для почтовых серверов UNIX** к **Dr.Web Enterprise Server**;
- удаление файла с паролем приведет к повторному запросу регистрационных данных у **Dr.Web Enterprise Server** при следующем запуске **Dr.Web Agent**.

Создание учетной записи на сервере вручную

Для создания учетной записи на сервере вручную:

- Создайте учетную запись на сервере с указанием идентификатора станции и пароля (см. Руководство администратора Антивирусной сети **Dr.Web ESS**);
- Запустите **Dr.Web Agent** с параметром командной строки `--newpwd` (или `-p`) и введите идентификатор и пароль. Хэш идентификатора станции и пароля пользователя записывается в файл с названием `pwd`. Данный файл создаётся в каталоге, путь к которому задается значением параметра **`CacheDir`** секции `[EnterpriseMode]` (по умолчанию `%var_dir/agent/`);
- В дальнейшем данные из этого файла используются для подключения **Dr.Web для**

почтовых серверов UNIX к **Dr.Web Enterprise Server**;

- Удаление файла с паролем приведет к необходимости повторить процедуру регистрации при следующем запуске **Dr.Web Agent**.

Задание конфигурации компонентов через Центр Управления Dr.Web

Через веб-интерфейс **Центра Управления Dr.Web** можно управлять настройкой конфигурации компонентов **Dr.Web для почтовых серверов UNIX** и **Dr.Web Daemon** (антивирусного модуля, входящего в базовый пакет **Dr.Web**).

В поставку **Dr.Web ESS** включены стандартные конфигурационные файлы компонентов **Dr.Web для почтовых серверов UNIX** и **Dr.Web Daemon** для основных UNIX-платформ: **Linux**, **FreeBSD** и **Solaris**. Соответственно, при настройке компонентов задание значений параметров происходит в этих файлах через веб-интерфейс **Центра Управления Dr.Web**. Затем каждый раз при запуске какого-либо из компонентов **Dr.Web Agent** запрашивает и получает конфигурацию от сервера централизованной защиты **Dr.Web Enterprise Server**.

Экспорт существующей конфигурации на сервер

При помощи **Dr.Web Agent**, работающего в режиме **Enterprise**, возможно автоматически экспортировать конфигурацию компонентов на **Dr.Web Enterprise Server**. Для этого необходимо экспортировать конфигурацию параметром командной строки `--export-config` (или `-e`) с указанием названия компонента (`DAEMON`, `MAILD`).

Пример:

```
# %bin_dir/drweb-agent --export-config MAILD
```

Запуск комплекса

Чтобы запустить комплекс:

- Через веб-интерфейс **Центра Управления Dr.Web** в настройках **Dr.Web Monitor** установите флаги **Daemon** и **Maild** для запуска соответствующих компонентов комплекса;
- Запустите **Dr.Web Monitor** на локальной станции:

  Для **Linux** и **Solaris**:

  ```
  # /etc/init.d/drweb-monitor start
  ```

  Для **FreeBSD**:

  ```
  # /usr/local/etc/rc.d/00.drweb-monitor.sh start
  ```

Интеграция с Dr.Web ESS версии 10

В состав продукта **Dr.Web для почтовых серверов UNIX** версии 6.0.2 входит две версии компонента **Dr.Web Agent**:

- **Dr.Web Agent**, представленный модулем **drweb-agent**, в режиме **enterprise mode** может взаимодействовать только с сервером **Dr.Web ESS** версии 6.
- **Dr.Web Agent**, представленный модулем **drweb-agent10**, в режиме **enterprise mode** может взаимодействовать только с сервером **Dr.Web ESS** версии 10.

Чтобы перейти на использование сервера централизованной защиты **Dr.Web ESS** версии 10, следует, помимо настройки интеграции, выполнить ряд дополнительных настроек.

Продукты, работающие под управлением операционной системы **FreeBSD** 6.x, не могут быть подключены к серверу **Dr.Web ESS** версии 10.

Настройка программного продукта для подключения к Dr.Web ESS версии 10

Так как **Dr.Web ESS** версии 10 не поддерживает управление компонентами **Dr.Web Monitor** и **Dr.Web Daemon**, в дополнение к стандартному файлу конфигурации `%etc_dir/agent.conf`, модуль **drweb-agent10** использует два дополнительных файла конфигурации: `es_monitor.conf` и `es_daemon.conf`, расположенных в том же каталоге. Эти файлы хранят параметры конфигурации модулей **Dr.Web Monitor** и **Dr.Web Daemon**, которые будут использоваться агентом для настройки работы этих модулей в режиме **enterprise mode**.

В каждой строке файла указывается значение некоторого параметра конфигурации соответствующего модуля в формате `<section>/<parameter> <value>`, где `<section>` – имя секции из конфигурационного файла компонента, `<parameter>` – имя параметра, а `<value>` – задаваемое значение параметра.

Пример (файл `es_monitor.conf`, задающий настройки для работы компонента **Dr.Web Monitor** в режиме **enterprise mode**):

```
Monitor/RunAppList DAEMON
```

В этой строке задается значение параметра **RunAppList**, находящегося в секции `[Monitor]` файла конфигурации **Dr.Web Monitor**. Данное значение параметра будет использовано, когда программный комплекс будет запущен в режиме **enterprise mode**. В этом случае **Dr.Web Monitor** запустит только компонент **Dr.Web Daemon**.

Пример (файл `es_daemon.conf`, задающий настройки для работы компонента **Dr.Web Daemon** в режиме **enterprise mode**):

```
Daemon/MaxCompressionRatio 500
```

В этой строке задается значение параметра **MaxCompressionRatio**, находящегося в секции `[Daemon]` файла конфигурации **Dr.Web Daemon**. Данное значение параметра будет использовано, когда программный комплекс будет запущен в режиме **enterprise mode**. В этом случае **Dr.Web Daemon** будет использовать 500 в качестве порогового значения коэффициента сжатия.

Для подключения программного продукта **Dr.Web для почтовых серверов UNIX** к серверу централизованной защиты **Dr.Web ESS** версии 10 следует:

1. Открыть файл мета-конфигурации `agent.mmc` (используется **Dr.Web Monitor** для взаимодействия с **Dr.Web Agent**) и заменить указанное в нем имя бинарного файла **drweb-agent** на **drweb-agent10**.

2. В файле `es_monitor.conf` указать требуемые для запуска компоненты, задав строку **`Monitor/RunAppList`**. Состав запускаемых компонентов должен совпадать составом компонентов, запускаемых программным комплексом в режиме **standalone** (указан непосредственно в параметре **`RunAppList`**, находящегося в секции `[Monitor]` файла конфигурации **Dr.Web Monitor**). В случае если должно быть запущено более одного компонента, они указываются через запятую, причем использование пробелов не допускается. Например:

```
Monitor/RunAppList DAEMON,MAILD
```

   В качестве имен компонентов указываются имена, заданные в секции `Application` mmc-файлов.

3. При необходимости изменить в файле `es_daemon.conf` значения параметров, которые будут использованы **Dr.Web Daemon** в режиме **enterprise mode**.

4. Если ранее использовался режим **standalone**, следует переключить **Dr.Web Agent** и **Dr.Web Monitor** в режим **enterprise mode**, задав соответствующие настройки в файлах конфигурации этих модулей, как показано в разделе Настройка компонентов для работы в режиме Enterprise.

5. Перезапустить модуль **Dr.Web Monitor**, выполнив команду:

```
# service drweb-monitor restart
```

Работа с вирусной статистикой

При работе программного комплекса **Dr.Web для почтовых серверов UNIX** с подключенным антивирусным модулем может производиться сбор сведений о вирусных событиях.

Собранная информация передается на сервер статистики **«Доктор Веб»** (http://stat.drweb.com/), либо на сервер централизованной защиты **Dr.Web Enterprise Server**, если **Dr.Web Agent** работает в режиме **Enterprise**. Обратите внимание, что если на компьютере одновременно установлено несколько антивирусных продуктов **Dr.Web**, работающих под управлением **Dr.Web Agent**, то он будет собирать и отправлять статистику по каждому из работающих продуктов.

Для соединения **Dr.Web Agent** с сервером статистики **«Доктор Веб»** необходим идентификатор пользователя – `UUID`. По умолчанию в качестве `UUID` используется `md5`-хэш от ключевого файла. Также вы можете получить персональный `UUID`, обратившись в службу поддержки. Такой `UUID` указывается в файле конфигурации **Dr.Web Agent** (параметр **`UUID`** в секции `[StandaloneMode]`).

Статистика собирается только для тех модулей **Dr.Web**, которые получают настройки от **Dr.Web Agent**. Информация о том, как настроить получение настроек от **Dr.Web Agent**, приведена в описании каждого модуля.

По адресу http://stat.drweb.com/ можно ознакомиться как с результатами обработки статистических данных по вашему серверу, так и с обобщенной статистической информацией по всем серверам, обслуживаемым антивирусными продуктами **Dr.Web** для ОС **UNIX** либо программным комплексом **Dr.Web для почтовых серверов UNIX** с подключенным антивирусным модулем.

В случае если работа ведется в режиме централизованной защиты, со статистикой можно ознакомиться также и на специальной странице **Центра управления Dr.Web**. Однако и в этом случае вся статистика, собранная сервером централизованной защиты **Dr.Web Enterprise Server**, также передается им на сервер статистики **«Доктор Веб»** в обобщенном виде для всей **Антивирусной сети**.

Результаты обработки содержат сведения о наиболее часто обнаруживаемых вирусах (для обобщенной статистики только в виде процента от общей суммы, а для индивидуальной – и в

виде количества обнаруженных вирусов) за определенный период.

Сведения могут представляться как в формате HTML, так и в виде файла с XML-разметкой. Последний вариант особенно удобен, если предполагается публикация полученных данных на веб-сайте, поскольку позволяет предварительно преобразовать данные в соответствии с дизайном сайта и концепцией представления информации на нем.

Для получения обобщенной статистики по всем обслуживаемым серверам откройте в веб-браузере страницу http://stat.drweb.com/. На странице представлен список обнаруженных вирусов на обслуживаемых серверах (в порядке убывания частоты встречаемости) с указанием для каждого из них количества обнаружений в процентной форме. Внешний вид страницы может различаться в зависимости от используемого веб-браузера.

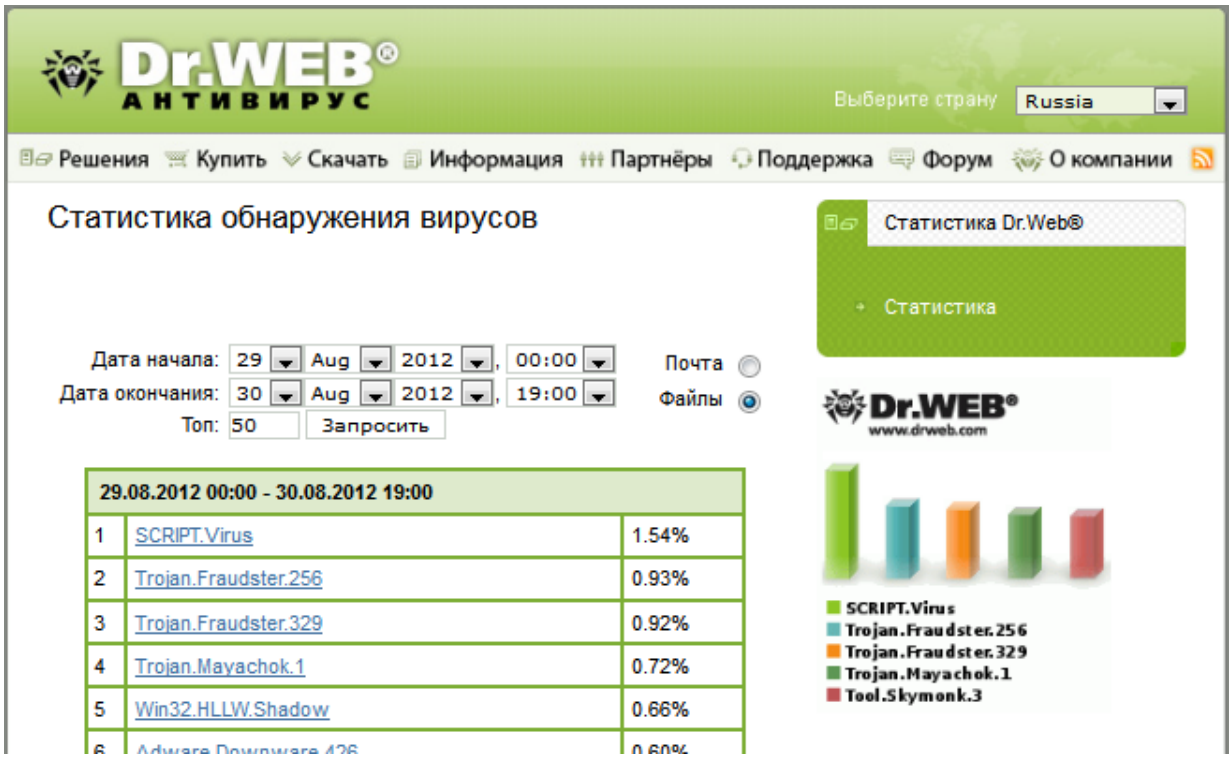


Рис. 16. Вирусная статистика

Вы можете изменить параметры запроса и повторить его:

- Установите переключатель в положение **Почта** или **Файлы** для получения статистики по вирусам, найденным в почтовых сообщениях или файлах.
- В раскрывающихся списках **Дата начала** и **Дата** окончания установите время и дату начала и окончания периода, за который требуется статистика.
- Введите в поле **Top** количество строк в таблице (будут представлены только наиболее часто встречающиеся вирусы).
- Нажмите на кнопку **Запросить**.

Файл с обобщенной статистикой в формате XML находится по адресу http://info.drweb.com/export/xml/top/.

Пример такого файла приведен ниже:

```
<drwebvirustop period="24" top="5"
  vdbaseurl="http://info.drweb.com/virus_description/"
  updatedutc="2009-06-09 09:32:02">
 <item>
  <vname>Win32.HLLM.Netsky</vname>
  <dwvlid>62083</dwvlid>
  <place>1</place>
  <percents>34.201062139103</percents>
 </item>
 <item>
  <vname>Win32.HLLM.MyDoom</vname>
  <dwvlid>9353</dwvlid>
  <place>2</place>
  <percents>25.1303270912579</percents>
 </item>
 <item>
  <vname>Win32.HLLM.Beagle</vname>
  <dwvlid>26997</dwvlid>
  <place>3</place>
  <percents>13.4593034783378</percents>
 </item>
 <item>
  <vname>Trojan.Botnetlog.9</vname>
  <dwvlid>438003</dwvlid>
  <place>4</place>
  <percents>7.86446592583328</percents>
 </item>
 <item>
  <vname>Trojan.DownLoad.36339</vname>
  <dwvlid>435637</dwvlid>
  <place>5</place>
  <percents>7.31494163115527</percents>
 </item>
</drwebvirustop>
```

В данном файле используются следующие атрибуты:

- `period` – продолжительность времени сбора статистики (в часах);
- `top` – количество представленных в таблице наиболее часто встречающихся вирусов;
- `updatedutc` – время последнего обновления статистики;
- `vname` – наименование вируса;
- `place` – место в статистике;
- `percents` – процент от общего числа обнаружений.

Пользователь не может задать продолжительность периода сбора статистики и размер выборки.

Для получения персональной статистики откройте страницу http://stat.drweb.com/view/<UUID>, где `<UUID>` – это md5-хэш ключевого файла пользователя. Страница персональной статистики имеет формат, аналогичный формату страницы обобщенной статистики, за исключением того, что для персональной статистики указывается также количество обнаруженных вирусов, а не только процент от общего количества.

Файл с персональной статистикой в формате XML находится по адресу http://stat.drweb.com/xml/<UUID>, где `<UUID>` – это md5-хэш ключевого файла пользователя.

Ниже приводится сокращенный пример такого файла:

```
<drwebvirustop period="24" top="2" user="<UUID>"
  lastdata="2005-04-12 07:00:00+04">
 <item>
  <caught>69</caught>
  <percents>24.1258741258741</percents>
  <place>1</place>
  <vname>Win32.HLLM.Netsky.35328</vname>
 </item>
 <item>
  <caught>57</caught>
  <percents>19.9300699300699</percents>
  <place>2</place>
  <vname>Win32.HLLM.MyDoom.54464</vname>
 </item>
</drwebvirustop>
```

В данном файле используются следующие атрибуты:

- `period` – продолжительность времени сбора статистики (в часах);
- `top` – количество представленных в таблице наиболее часто встречающихся вирусов;
- `user` – идентификатор пользователя;
- `lastdata` – время последнего получения данных от пользователя;
- `vname` – наименование вируса;
- `place` – место в статистике;
- `caught` – количество обнаружений данного вируса;
- `percents` – процент от общего числа обнаружений.

Как и в случае запроса обобщенной статистики, пользователь не может задать продолжительность периода сбора статистики и размер выборки.

Dr.Web Monitor

Компонент **Dr.Web Monitor** представлен модулем **drweb-monitor** и предназначен для повышения отказоустойчивости всего программного комплекса **Dr.Web для почтовых серверов UNIX**. Он осуществляет запуск всех модулей, подгружая при необходимости их дополнительные компоненты. Если запустить какой-либо модуль не удалось, **Dr.Web Monitor** повторяет попытку. Количество попыток и время между ними определяются настройками компонента.

После того, как все модули были загружены, **Dr.Web Monitor** осуществляет постоянный контроль их работы. **Dr.Web Monitor** может обмениваться с этими модулями различными управляющими сигналами. В случае сбоя какого-либо модуля или одного из его компонентов **Dr.Web Monitor** перезапускает его. Максимальное количество попыток перезапуска и время между ними также определяются настройками **Dr.Web Monitor**. При возникновении неполадок в работе какого-либо модуля **Dr.Web Monitor** одним из доступных ему способов оповещает об этом администратора.

Dr.Web Monitor может взаимодействовать с компонентом **Dr.Web Agent**, обмениваясь с ним управляющими сигналами.

Режимы работы

При необходимости продукты компании **«Доктор Веб»** могут быть подключены к корпоративной или частной **Антивирусной сети**, управляемой комплексом **Dr.Web ESS**. Работа в режиме централизованной защиты не требует установки дополнительного программного обеспечения или удаления **Dr.Web для почтовых серверов UNIX**.

Для обеспечения этой возможности, **Dr.Web Monitor** может работать в одном из двух режимов:

- Одиночном (**standalone mode**) режиме, когда защищаемый компьютер не включен в **Антивирусную сеть** и управляется локально. В этом режиме конфигурационные и ключевые файлы находятся на локальных дисках, **Dr.Web Monitor** полностью управляется с защищаемого компьютера, а все необходимые модули **Dr.Web** запускаются в соответствии с локальными настройками **Dr.Web Monitor**.
- Режиме централизованной защиты (**enterprise mode**), когда защитой компьютера управляет сервер централизованной защиты **Dr.Web Enterprise Server**. В этом режиме некоторые функции и настройки **Dr.Web для почтовых серверов UNIX** могут быть изменены или заблокированы в соответствии с общей (корпоративной) стратегией антивирусной защиты. В этом режиме используется лицензионный ключевой файл, получаемый от сервера централизованной защиты **Dr.Web Enterprise Server**. Персональный лицензионный ключевой файл на локальном компьютере не используется.

Чтобы использовать режим централизованной защиты

1. Свяжитесь с системным администратором вашей сети чтобы получить файл с открытым ключом и параметры соединения с сервером централизованной защиты.
2. В конфигурационном файле **Dr.Web Monitor** (по умолчанию `%etc_dir/monitor.conf`) установите `Yes` в качестве значения параметра **UseEnterpriseMode**.

   В режиме централизованной защиты некоторые функции и настройки **Dr.Web для почтовых серверов UNIX** могут быть изменены или заблокированы в соответствии с общей (корпоративной) стратегией антивирусной защиты. В этом режиме используется лицензионный ключевой файл с сервера централизованной защиты. Персональный лицензионный ключевой файл на локальном компьютере не используется.

Чтобы **Dr.Web для почтовых серверов UNIX** полностью поддерживал режим централизованной защиты, **Dr.Web Agent** также должен работать в режиме централизованной защиты. Для подробностей обратитесь к разделу Режимы работы **Dr.Web Agent**.

Чтобы использовать одиночный (standalone) режим

1. Убедитесь, что все необходимые модули, указанные в параметре **RunAppList** в секции [Monitor] конфигурационного файла **Dr.Web Monitor** (по умолчанию %etc_dir/monitor.conf), установлены и настроены корректно.
2. Установите No в качестве значения параметра **UseEnterpriseMode** секции [Monitor] конфигурационного файла **Dr.Web Monitor**.

   При включении этого режима все настройки **Dr.Web для почтовых серверов UNIX** будут разблокированы, и вы вновь получите доступ ко всем функциям и настройкам **Dr.Web для почтовых серверов UNIX**.

Для работы в одиночном режиме **Dr.Web для почтовых серверов UNIX** необходим действующий лицензионный ключ. Ключевые файлы, полученные от сервера централизованной защиты **Dr.Web Enterprise Server**, не могут быть использованы в этом режиме.

Параметры командной строки

Для запуска **Dr.Web Monitor** используется следующая команда:

```
drweb-monitor [параметры]
```

Dr.Web Monitor допускает использование следующих параметров:

Краткий вариант	Расширенный вариант	Аргументы
-h	--help	
Описание: Вывод на экран краткой справки по имеющимся параметрам командной строки и завершение работы модуля		
-v	--version	
Описание: Вывод на экран информации о текущей версии **Dr.Web Monitor** и завершение работы модуля		
-u	--update	
Описание: Запуск процесса обновления для всех компонентов **Dr.Web для почтовых серверов UNIX**		
-C	--check-only	
Описание: Проверка корректности конфигурации модуля **Dr.Web Monitor**. Данный параметр командной строки не может быть использован при наличии в системе уже запущенного экземпляра **Dr.Web Monitor**		
-A	--check-all	<путь к файлу>
Описание: Проверка корректности конфигурации всех компонентов **Dr.Web для почтовых серверов UNIX**		
-c	--conf	<путь к файлу>
Описание: Использование при запуске указанного конфигурационного файла		
-r	--run	<имя приложения>[,<имя приложения>,...]

Краткий вариант	Расширенный вариант	Аргументы
Описание: Запуск указанных приложений. В качестве аргументов следует использовать имена приложений, указанных в заголовке секции `Application "<имя приложения>"` соответствующего `mmc`-файла (см. раздел Взаимодействие с компонентами программного комплекса). Данный параметр командной строки не может быть использован при наличии в системе уже запущенного экземпляра **Dr.Web Monitor**		

Пример использования:

```
drweb-monitor -r AGENT, MAILD
```

Конфигурационный файл

Настройки компонента **Dr.Web Monitor** задаются отдельным конфигурационным файлом `%etc_dir/monitor.conf`.

Устройство конфигурационного файла и краткое описание правил задания параметров конфигурации приведены в разделе Конфигурационные файлы.

Секция [Logging]

В секции `[Logging]` собраны параметры, управляющие ведением журналов работы компонента **Dr.Web Monitor** программного комплекса **Dr.Web для почтовых серверов UNIX:**

Секция [Logging]

Level = {уровень подробности}	Уровень подробности сохранения в журнал работы компонента общих событий. Допускается использование следующих уровней: • Quiet • Error • Alert • Info • Debug Значение по умолчанию: **Level** = Info
IPCLevel = {уровень подробности}	Уровень подробности сохранения в журнал работы компонента событий подсистемы IPC. Допускается использование следующих уровней: • Quiet • Error • Alert • Info • Debug Значение по умолчанию: **IPCLevel** = Error
SyslogFacility = {метка syslog}	Метка записи при использовании системного сервиса **syslog** Значение по умолчанию: **SyslogFacility** = Daemon

FileName = {syslog | путь к файлу}	Имя файла журнала или `syslog`, если нужно использовать системный сервис **syslog** Значение по умолчанию: **FileName** = syslog

Секция [Monitor]

В секции `[Monitor]` собраны основные настройки компонента **Dr.Web Monitor:**

Секция [Monitor]

RunForeground = {логический}	Значение `Yes` запрещает **Dr.Web Monitor** переходить в режим демона, т.е. становиться фоновым процессом без управляющего терминала. Эта возможность может быть использована некоторыми средствами мониторинга (например, **daemontools**). Значение по умолчанию: **RunForeground** = No
User = {текст}	Имя пользователя, с правами которого запускается **Dr.Web Monitor.** Пожалуйста, обратите внимание, что при работе программного комплекса **Dr.WebMailD** в режиме SMTP-прокси или интеграции его с MTA CGP или MTA Exim значение данного параметра должно быть установлено в `root`. Значение по умолчанию: **User** = drweb
Group = {текст}	Имя пользовательской группы, с правами которой запускается **Dr.Web Monitor.** Пожалуйста, обратите внимание, что при работе программного комплекса **Dr.WebMailD** в режиме SMTP-прокси или интеграции его с MTA CGP или MTA Exim значение данного параметра должно быть установлено в `root`. Значение по умолчанию: **Group** = drweb
PidFileDir = {путь к каталогу}	Имя каталога, содержащего файл, в который при запуске **Dr.Web Monitor** записывается информация об идентификаторе его процесса (`PID`). Значение по умолчанию: **PidFileDir** = %var_dir/run/
ChDir = {путь к каталогу}	Смена активного каталога при запуске **Dr.Web Monitor**. Если значение параметра задано, то при запуске **Dr.Web Monitor** делает активным каталог, указанный в значении этого параметра. Если значение параметра не задано, то смены активного каталога не происходит. Значение по умолчанию: **ChDir** = /
MetaConfigDir = {путь к каталогу}	Путь к каталогу с файлами мета-конфигурации. В этих файлах задаются параметры работы **Dr.Web Monitor** с модулями программного комплекса. Содержание файлов мета-конфигурации задается разработчиками программного

	продукта и не требует редактирования. Значение по умолчанию: **MetaConfigDir** = %etc_dir/monitor/
Address = {адрес}	Сокет, через который **Dr.Web Monitor** взаимодействует с другими модулями антивируса. Значение по умолчанию: **Address** = local:%var_dir/ipc/.monitor
Timeout = {числовое значение}	Максимальное время установления соединения между **Dr.Web Monitor** и другими компонентами программного комплекса в секундах. Значение по умолчанию: **Timeout** = 5
TmpFileFmt = {текст}	Шаблон имени временных файлов **Dr.Web Monitor**. Формат шаблона: путь_к_файлу.XXXXXX, где X - произвольный символ (буква или цифра) в именах создаваемых временных файлов. Значение по умолчанию: **TmpFileFmt** = %var_dir/msgs/tmp/monitor.XXXXXX
RunAppList = {текст}	Список модулей, запускаемых **Dr.Web Monitor**. Названия модулей отделяются друг от друга запятыми. Обратите внимание, что при удалении какого-либо модуля из системы его название не удаляется из списка RunAppList автоматически и должно быть удалено вручную. В противном случае **Dr.Web Monitor** не сможет запуститься сам и запустить остальные компоненты. Значение по умолчанию: **RunAppList** = AGENT
UseEnterpriseMode = {логический}	При значении Yes данного параметра список модулей, запускаемых **Dr.Web Monitor**, берется не из параметра **RunAppList**, а от модуля **Dr.Web Agent**. Значение по умолчанию: **UseEnterpriseMode** = No
RecoveryTimeList = {список числовых значений}	Временные промежутки между попытками перезапуска "зависших" приложений в секундах. Для параметра можно задать несколько значений, перечислив их через запятую. Первая попытка перезагрузки приложения производится через время, указанное первым значением параметра, вторая – через время, указанное вторым и т.д. Значение по умолчанию: **RecoveryTimeList** = 0,30,60
InjectCmd = {текст}	Команда для отсылки отчетов. Обратите внимание, что для отправки сообщений на адрес, отличный от root@localhost, надо в команде указать действительный адрес. Значение по умолчанию: **InjectCmd** = "/usr/sbin/sendmail -t"

AgentAddress = {адрес}	Сокет, через который **Dr.Web Monitor** взаимодействует с **Dr.Web Agent** (значение параметра должно совпадать со значением параметра **Address** конфигурационного файла **Dr.Web Agent**). Значение по умолчанию: **AgentAddress** = local:%var_dir/ipc/.agent
AgentResponseTime = {числовое значение}	Максимальное время отклика модуля **Dr.Web Agent** в секундах Если в течение этого времени от модуля не поступает ответа, то **Dr.Web Monitor** перезапускает его. Если указано значение 0, время отклика не ограничивается. Значение по умолчанию: **AgentResponseTime** = 5

Запуск

В процессе запуска **Dr.Web Monitor** (при установках по умолчанию) осуществляются следующие действия:

- производится поиск и загрузка конфигурационного файла; если файл не найден, то загрузка прекращается;
- **Dr.Web Monitor** переходит в режим демона, поэтому сообщения о дальнейших проблемах не могут быть выведены на терминал и выводятся только в файл журнала;
- создается сокет для взаимодействия с другими модулями программного комплекса **Dr.Web для почтовых серверов UNIX**. В случае использования TCP-соединений, подключений может быть несколько (загрузка продолжится, если удалось создать хотя бы одно из них). Если используется UNIX-сокет, то он может быть создан только тогда, когда содержащий его каталог доступен на запись и чтение пользователю, с чьими привилегиями работает модуль **drweb-monitor**. Если ни один сокет не может быть создан, загрузка прекращается;
- создается PID-файл, в котором хранится информация об идентификаторе процесса **Dr.Web Monitor**. Если создать PID-файл не удалось, то загрузка прекращается;
- модуль **drweb-monitor** запускает остальные модули программного комплекса **Dr.Web для почтовых серверов UNIX**. Если какой-либо из модулей не загружается, **Dr.Web Monitor** пытается запустить его повторно. Если все попытки **Dr.Web Monitor** загрузить модуль окончились неудачей, **Dr.Web Monitor** выгружает все уже загруженные модули и завершает свою работу. Обо всех проблемах с запуском модулей программного комплекса **Dr.Web Monitor** сообщает одним из доступных ему способов (записью в файл журнала, сообщением электронной почты, запуском произвольной программы). Способы оповещения, используемые для разных модулей, задаются в файле мета-конфигурации **Dr.Web Monitor** (.mmc).

Для успешного запуска **Dr.Web Monitor** в автоматическом режиме:

- либо в `enable`-файле **Dr.Web Monitor** переменной `ENABLE` должно быть присвоено значение `1` (для **Linux** и **Solaris**);
- либо строка `drweb_monitor_enable="YES"` должна быть добавлена в файл `/etc/rc.conf` (для **FreeBSD**).

В процессе работы установочного скрипта при выборе соответствующей возможности в диалоге все сервисы, включая **Dr.Web Monitor**, будут запущены автоматически.

Расположение `enable`-файлов зависит от способа установки **Dr.Web для почтовых серверов UNIX**:

- Установка при помощи **универсального пакета для UNIX**:
  Файлы располагаются в каталоге `%etc_dir` и называются
  `drwebd.enable`,
  `drweb-monitor.enable`.
- Установка из **нативных DEB-пакетов**:
  Файлы располагаются в каталоге `/etc/defaults` и называются
  `drwebd`,
  `drweb-monitor`.
- Установка из **нативных RPM-пакетов**:
  Файлы располагаются в каталоге `/etc/sysconfig` и называются
  `drwebd.enable`,
  `drweb-monitor.enable`.

Взаимодействие с компонентами программного комплекса

Взаимодействие с компонентами программного комплекса осуществляется с помощью файлов мета-конфигурации (`.mmc`). Эти файлы включены в пакеты тех продуктов, компоненты которых могут работать под управлением **Dr.Web Monitor**, и располагаются в каталоге, определяемом параметром **`MetaConfDir`** (по умолчанию – `%etc_dir/monitor`). В этих файлах описывается состав компонентов, расположение бинарных файлов, порядок их запуска и параметры запуска. Как правило, в одном файле указывается описание одного компонента, а имя файла совпадает с именем компонента **Dr.Web для почтовых серверов UNIX**.

Описание каждого компонента содержится в секции `Application "имя_компонента"`. В конце секции обязательно должно быть поставлено `EndApplication`.

В описании компонента должны присутствовать следующие параметры:

- **`FullName`**: полное имя приложения;
- **`Path`**: путь к бинарным файлам;
- **`Depends`**: имена компонентов, которые должны запускаться до запуска описываемого компонента. Например, компонент `AGENT` должен запускаться до компонента `DAEMON`, поэтому в `mmc`-файле для **Dr.Web Daemon** параметр **`Depends`** имеет значение `"AGENT"`. Если подобные зависимости отсутствуют, то параметр может быть пропущен;
- **`Components`**: список бинарных файлов компонентов, запускаемых при старте приложения. Компоненты запускаются в том порядке, в котором перечислены. Для каждого из компонентов через пробел указываются:
  - Аргументы командной строки, передаваемые модулю при запуске (могут быть заключены в кавычки);
  - Максимальное время в секундах, отводимое на запуск компонента (StartTimeout);
  - Максимальное время в секундах для остановки (StopTimeout);
  - Тип оповещения и права для запуска.

  Тип оповещения указывает, куда высылать сообщения о сбоях компонента. Он может принимать значения `MAIL` (осуществляется отсылка оповещений по почте) и `LOG` (информация о сбоях только записывается в журнал).

  Права для запуска указывают группу и пользователя, с чьими правами будет запускаться компонент.

Пример mmc-файла для Dr.Web Daemon:

```
Application "DAEMON"
 FullName    "Dr.Web (R) Daemon"
 Path        "/opt/drweb/"
 Depends     "AGENT"
 Components
  # name  args  MaxStartTime  MaxStopTime  NotifyType  User:Group
  drwebd "-a=local:/var/drweb/ipc/.agent --foreground=yes" 30 10 MAIL drweb:drweb
 EndComponents
EndApplication
```

Пример mmc-файла для Dr.Web MailD:

```
Application "MAILD"
 FullName    "Dr.Web (R) MailD"
 Path        "/opt/drweb/"
 Depends     "AGENT"
 Components
  # name  args  MaxStartTime  MaxStopTime  NotifyType  User:Group
  drweb-notifier  local:/var/drweb/ipc/.agent  30   30  MAIL  drweb:drweb
  drweb-sender    local:/var/drweb/ipc/.agent  15   30  LOG   drweb:drweb
  drweb-maild     local:/var/drweb/ipc/.agent  120  30  MAIL  drweb:drweb
  drweb-receiver  local:/var/drweb/ipc/.agent  15   30  MAIL  root:drweb
 EndComponents
EndApplication
```

Dr.Web MailD

Компонент обработки почты **Dr.Web MailD** является комплексными. Он состоит из группы совместно работающих модулей, предназначенных для приема и передачи почтовых сообщений и их антивирусной и антиспам-проверки.

Комплексный компонент **Dr.Web MailD** может исполнять функции:

- Прокси-сервера для протоколов SMTP и LMTP (режим **SMTP/LMTP-прокси**);
- **Фильтра сообщений для почтовых систем**. Поддерживается интеграция со следующими почтовыми системами:
  - **Sendmail**;
  - **Postfix**;
  - **Exim**;
  - **CommuniGate Pro**;
  - **Courier**;
  - **Zmailer**;
  - **Qmail**.
- **Прокси-сервера клиентских почтовых протоколов** (POP3, IMAP), выполняющего роль фильтра-посредника между почтовой системой и почтовым клиентом пользователя.

Настройка параметров функционирования комплексного компонента обработки почты **Dr.Web MailD** (как в целом, так и его отдельных модулей) осуществляется при помощи конфигурационных файлов. Доступно также интерактивное управление работой **Dr.Web MailD** через интерфейс управления наподобие командной строки.

Структура компонента обработки почты

Общая структура программного комплекса **Dr.Web для почтовых серверов UNIX** приведена во Введении.

На рисунке ниже изображена структура комплексного компонента **Dr.Web MailD** и схема его взаимодействия с почтовыми системами (MTA и MTA/MDA).

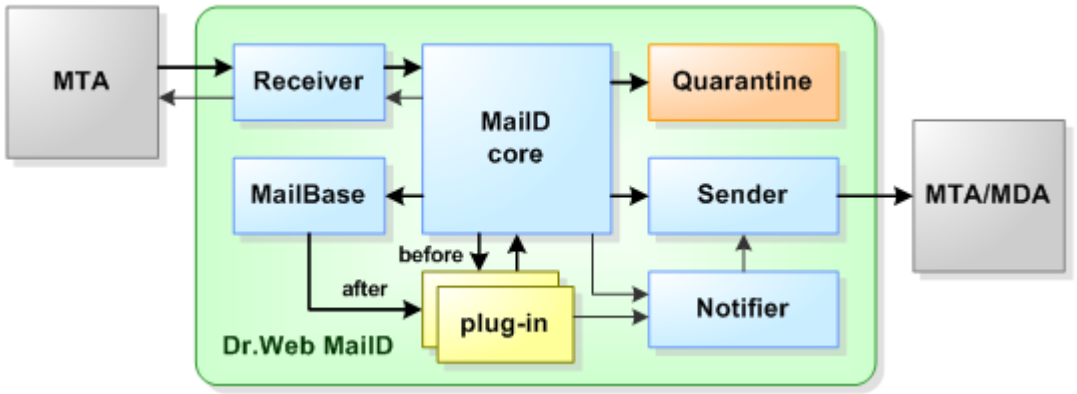


Рис. 17. Структура комплексного компонента Dr.Web MailD

Краткое описание компонентов, входящих в состав комплексный компонент обработки почты **Dr.Web MailD**, приведено в таблице.

Компонент	Описание
Receiver	Компонент предназначен для приема почтовых сообщений от почтовых серверов (MTA) или почтовых систем (MDA) для антивирусной и/или антиспам-проверки. Функции этого

Компонент	Описание
	компонента реализуются разными модулями (**drweb-receiver**, **drweb-milter**, **drweb-cgp-receiver** и т.п.), включаемыми в состав программного комплекса в зависимости от того, с какой почтовой системой интегрирован **Dr.Web для почтовых серверов UNIX**. Поддерживается возможность одновременной работы в составе комплекса нескольких модулей, реализующих функции компонента **Receiver**, что позволяет получать и обрабатывать почту сразу из нескольких источников. Некоторые модули также поддерживают возможность модификации/отправления полученных сообщений, принимая результаты проверки писем от компонента **MailD core** (например, такой возможностью обладает модуль **drweb-milter**, что позволяет ему возвращать почтовой системе **Sendmail** результат проверки писем до окончания SMTP-сессии).
MailBase	Специальная база данных, хранящая принятые письма до момента окончания их проверки подключаемыми модулями и отправки получателю, если они работают в асинхронном режиме. Для хранения писем используется файловое хранилище (письма хранятся в виде файлов). Также может хранить перечни пользователей (адресатов писем), их групп, и связанных с ними индивидуальных настроек обработки писем, а также хранит накопленную статистику обработки писем.
MailD core	Основной компонент системы обработки почты. Он производит MIME-разбор поступивших сообщений, передает письма на обработку подключаемым модулям (**plug-ins**) и отвечает за хранение писем в базе данных. Результаты проверки отправляются либо компоненту **Receiver** (если идет обработка письма в асинхронном режиме и еще не истекло время ожидания результата проверки), либо компоненту **Sender**. Само проверенное письмо, если оно не было отвергнуто в процессе проверки, отправляется компоненту **Sender** для доставки получателю. Функции компонента реализует модуль **drweb-maild**
Quarantine	**Карантин** – специальный каталог файловой системы, временно хранящий письма, не прошедшие проверку подключаемыми модулями (если в результате проверки письмо не было удалено или отвергнуто). Вместо каталога может быть использовано хранилище **DBI**.
Sender	Компонент отвечает за отправление всех исходящих писем, включая уведомления, отчеты и DSN. Письма отправляются либо напрямую в сопряженную почтовую систему, либо по протоколам SMTP/LMTP. В зависимости от используемых почтовых систем и протоколов, функции компонента **Sender** выполняют разные модули (**drweb-sender**, **drweb-cgp-sender** и т.п.). Как правило, модуль, используемый в качестве компонета **Sender**, является парным модулю, используемому в качестве компонента **Receiver**. Компонент **Sender** может получать запросы на отправление писем и уведомлений от компонентов **MailD core**, **Notifier** и **Monitor**
Notifier	Компонент отвечает за создание и отправку служебных писем двух типов: • **Уведомлений MailD** – представляет собой письмо, направленное с адреса, указанного в параметре **FilterMail** настроек **Dr.Web MailD**, и содержащее некоторое сообщение **Dr.Web MailD** (например, о наличии вирусов в полученном письме); • **DSN** (*delivery status notification*) – представляет собой письмо, автоматически формируемое MTA при возникновении тех или иных неполадок и содержащее служебную информацию. В соответствии с требованиями имеет пустой заголовок FROM. Уведомления формируются в процессе работы комплекса. Запрос на отправку уведомлений могут отправлять как подключаемые модули (например, при обнаружении вируса), так и другие компоненты системы в случае их соответствующей настройки. Например, компонент **MailD core** может посылать запрос на создание общего отчета со статистикой работы всех используемых подключаемых модулей, а компонент **Sender** может посылать запрос на формирование DSN о невозможности отправить письмо. Уведомления могут рассылаться отправителям и получателям писем, а также администратору системы.

Компонент	Описание
	Функции компонента реализует модуль **drweb-notifier**
plug-in	Дополнительные модули, используемые для анализа писем, в том числе, на содержание вирусов, вредоносного ПО и наличие признаков спама. Обработка писем подключаемыми модулями осуществляется в порядке, определяемом пользователем системы. Состав используемых подключаемых модулей и порядок их вызова могут быть изменены без перезапуска **Dr.Web MailD**, для этого достаточно изменить соответствующие параметры в конфигурационном файле и отправить сигнал SIGHUP компоненту **MailD core** или компоненту **Dr.Web Monitor**. Основные используемые подключаемые модули (состав зависит от поставки): • **Drweb** – модуль, осуществляющий антивирусную проверку почты с помощью компонента **Dr.Web Daemon**, которому сообщения передаются на проверку уже разобранными на части. • **Dr.Web HeadersFilter** – модуль, осуществляющий фильтрацию сообщений по содержимому заголовков. При задании правил фильтрации можно использовать регулярные выражения (используется синтаксис регулярных выражений **Perl**). • **Vaderetro** – модуль, осуществляющий спам-фильтрацию почты через стороннюю библиотеку **VadeRetro**. Правила распознавания спама, используемые этой библиотекой, динамически обновляются через компонент **Dr.Web Updater**, что позволяет сохранять стабильно высокое качество фильтрации. • **Dr.Web Modifier** – модуль, позволяющий осуществлять модификацию сообщения или какой-либо его части в зависимости от содержимого сообщения и его конверта. С его помощью можно, например, добавлять текстовую подпись в проверенные сообщения или удалять картинки из письма, отмеченного как спам.

Конкретный набор модулей, образующих программный комплекс **Dr.Web для почтовых серверов UNIX**, зависит от поставки, а также от выбранной почтовой системы, с которой он интегрируется и способа интеграции.

Особенности Dr.Web MailD

Dr.Web MailD имеет развитую систему настройки Правил обработки писем (в том числе – их внутренней межмодульной маршрутизации), формирования статистики, отправки отчетов и уведомлений об обработке проверяемых писем, включая как DSN, так и специальные уведомления MailD (о вирусах и т.п.).

В Правилах обработки писем, а также во всех параметрах конфигурации, имеющих специальный тип Lookup, можно использовать информацию, извлекаемую при помощи запросов LDAP, из текстовых файлов, а также из реляционных баз данных. Поддерживаются следующие системы управления базами данных (СУБД):

- **Oracle**
- **MySQL**
- **PostgreSQL**
- **SQLite**
- **Firebird**
- **CDB**
- **Berkeley**

Кроме того, **Dr.Web MailD** может подключиться через механизм **ODBC** к любому источнику данных, для которого имеется ODBC-драйвер. Работа с каждым источником данных производится независимо от других, с использованием настроек подключения, индивидуальных для этого источника. Одновременно в Правилах обработки писем и во всех параметрах конфигурации, имеющих тип Lookup, могут извлекаться и использоваться данные из разных источников.

Также **Dr.Web MailD** позволяет вести внутреннюю базу данных пользователей (адресатов

писем), с заданием для каждого пользователя (а также группы пользователей) индивидуальных Правил обработки их сообщений и формирования индивидуальной или групповой статистики.

Работа с базой данных пользователей ведется только через интерфейс интерактивного управления.

Для борьбы со спамом и атаками типа DHA в **Dr.Web MailD** реализованы технология определения репутации IP-адресов клиентов, технология управления счетом (как соединений, так и каждого письма), а также технология управления SMTP-ограничениями, позволяющая отсеять подозрительного клиента еще на этапе подключения и передачи письма (сразу в рамках SMTP-сессии). Указанные технологии позволяют повысить эффективность борьбы со спамом и снизить нагрузку на защищаемую почтовую систему.

Получатель или отправитель (адресат) сообщения, помещенного в **Карантин**, имеет возможность дистанционного управления сообщением в нем при помощи отправки на **Dr.Web для почтовых серверов UNIX** специальных управляющих писем, отправляемых в ответ на уведомления о помещении письма в **Карантин**, поступившие адресату.

В случае если объем и интенсивность почтового трафика велики, **Dr.Web MailD** позволяет организовать работу комплекса в режиме кластера, с размещением различных компонентов программного комплекса на нескольких серверах с распределением нагрузки для повышения производительности. Для этого в состав комплекса включены специальные **прокси-компоненты** (**Proxy client** и **Proxy server**), обеспечивающие прозрачное удаленное взаимодействие компонентов **Sender** и **Receiver** с центральным компонентом **MailD core**.

При необходимости может быть обеспечено взаимодействие сразу с несколькими MTA/MDA при помощи механизма одновременной параллельной работы нескольких пар компонентов **Sender** и **Receiver** (при этом они могут быть разного типа).

Обработка сообщений

Алгоритм обработки почтовых сообщений

Обработка почтовых сообщений происходит по следующему алгоритму:

1. Сообщения, поступающие от MTA, принимаются компонентом **Receiver**, который передает их компоненту **MailD core**, отвечающему за проверку почтовых сообщений.
2. Компонент **MailD core** производит MIME-разбор сообщений, передает письма на обработку подключаемым модулям и отвечает за хранение писем в базе данных.

   При выборе значений параметров, которые следует применить к обрабатываемому письму, используется следующий алгоритм:

   - Просматриваются Правила, имеющиеся во встроенной базе данных и связанные с **получателем** данного письма (получатель определяется по заданному отправителем `RCPT TO`).
   - Просматриваются Правила, имеющиеся во встроенной базе данных и связанные со всеми группами, к которым относится пользователь-получатель. Просмотр Правил групп производится в обратном порядке: с настроек самой последней группы и до первой в списке группы.
   - Просматриваются Правила, заданные в секции `[Rules]` основного конфигурационного файла.

   Обратите внимание на порядок обхода Правил:

   - Все Правила в текущей просматриваемой группе Правил всегда проверяются в порядке их задания.
   - Для каждого проверяемого Правила проверяется условие `CONDITION` – и если оно

истинно, то значение требуемого параметра ищется среди элементов секции `SETTINGS` этого правила.

- Если условие `CONDITION` оказалось ложно, то просмотр Правила заканчивается и происходит переход к поиску значения в следующем Правиле.
- Если условие `CONDITION` истинно и после него стоит директива `cont`, то происходит переход к проверке следующего Правила. Если же после истинного `CONDITION` стоит директива `stop`, то просмотр Правил заканчивается вне зависимости от того, было найдено значение требуемого параметра или нет.

Значение параметра по результатам просмотра Правил всегда определяется следующим образом:

- Если искомый параметр встретился в одном из сработавшем правил, то используется его значение, извлеченное из части `SETTINGS` (обратите внимание, что при срабатывании нескольких Правил для одного и того же параметра, результирующее значение этого параметра зависит от его семантики. Подробнее об этом см. в разделе Правила обработки писем).
- Если Правила отсутствуют, или ни одно Правило не сработало, или ни в одном из сработавших Правилах параметр не нашелся, то извлекается значение этого параметра, заданное в соответствующей секции конфигурационного файла.
- Если в конфигурационном файле искомый параметр не задан, то используется его значение по умолчанию.

Обратите внимание, что если у письма имеется несколько получателей, то проверяется срабатывание Правил для каждого получателя. Подробнее см. в Описании Правил.

3. Обработка писем может производиться подключаемыми модулями сначала в синхронном режиме (если такие модули указаны), а затем, после сохранения писем в хранилище, – в асинхронном режиме (если такие модули указаны).
4. Если проверка производилась в синхронном режиме, то результаты проверки писем отправляются компоненту **Receiver** (если еще не истекло время ожидания результата проверки).
5. Если письмо не прошло проверку, то оно может быть либо отвергнуто (с уведомлением получателя либо без уведомления, в зависимости от настроенного действия), а также перенаправлено на другой адрес или перемещено в **Карантин**. В случае если письмо было отвергнуто, то формируется уведомление отправителю, и, при необходимости – получателю, для этого производится обращение к компоненту **Notifier**.
6. Компонент **Sender** отвечает за отправление всех исходящих писем в различные почтовые системы. Он отправляет как исходящие проверенные письма, так и все сформированные **Dr.Web MailD** уведомления и отчеты.

Режимы обработки почтовых сообщений

При обработке почтовых сообщений **Dr.Web MailD** использует два режима обработки:

1. **Синхронный режим** ("**before-queue**"): поступившее от отправителя через **Receiver** письмо обрабатывается "на лету", т.е. не сохраняется в хранилище **MailBase**, а сразу же передается на обработку подключаемым модулям, перечисленным в списке **`BeforeQueueFilters`**. При этом **Receiver** не шлет отправителю ответа до окончания обработки или до истечения тайм-аута, отведенного на обработку письма. В случае если письмо не пройдет проверку, отправитель получит от **Receiver** в качестве ответа код ошибки.
2. **Асинхронный режим** ("**after-queue**"): поступившее через **Receiver** письмо сохраняется в локальное хранилище **MailBase**, а **Receiver** отвечает отправителю ответом, что письмо успешно получено. После этого письмо поступает на обработку в подключаемые модули, перечисленные в списке **`AfterQueueFilters`**. Если принятое письмо не пройдет проверку, то уведомление отправителя производится отправкой ему DSN, содержащей отчет об

ошибке.

Если часть подключаемых модулей указана в списке **BeforeQueueFilters**, а часть – в списке **AfterQueueFilters**, то письмо последовательно обрабатывается сначала в синхронном, а затем – в асинхронном режимах.

Обратите внимание, что если используются подключаемые модули **Dr.Web HeadersFilter** и **Dr.Web Modifier**, т.е. если для них заданы локальные правила обработки писем, или же заданы Правила обработки писем, переопределяющие параметры данных модулей, то, если **Dr.Web MailD** работает в режиме **SMTP/LMTP-прокси**, рекомендуется помещать их в список **AfterQueueFilters**. Если же **Dr.Web MailD** интегрирован с какой-либо MTA, то рекомендуется помещать все подключаемые модули в список **BeforeQueueFilters**, но при этом увеличить тайм-аут IPC (параметр **IPCTimeout**, определенный в конфигурационном файле в секции `[General]`). Если эти подключаемые модули вовсе не используются для обработки писем, то рекомендуется исключить их из всех очередей в секции `[Filters]`.

При любом способе проверки, после ее окончания, письмо, если оно не было отвергнуто (т.е. к нему не применялись действия `reject`, `discard` или `tempfail`), отправляется для доставки получателю в компонент **Sender**. При этом, если письмо поступает на отправку из синхронного режима проверки, то отправляться оно так же будет синхронно, т.е. **Receiver** будет ждать результата отправки письма через **Sender** или окончания тайм-аута. Использование тайм-аута нужно для того, чтобы не вызывать ошибки взаимодействия с внешними MTA при ведении протокольных диалогов приема и отправки писем.

Пожалуйста, обратите внимание, что наилучшие режимы использования подключаемых модулей зависят от типа проверяемого почтового трафика и типа MTA, с которым интегрирован **Dr.Web MailD** (включая способ интеграции). Таким образом, при изменении настроек по умолчанию прежде всего прочтите описание способа интеграции с выбранным MTA в соответствующем разделе Руководства.

Особенности взаимодействия компонентов Receiver, MailD core и Sender в различных режимах

1. Максимально допустимое время взаимодействия компонентов **Receiver** и **MailD core** ограничено величиной тайм-аута **IPCTimeout** (определен в конфигурационном файле, в секции `[General]`) В синхронном режиме этот же тайм-аут ограничивает также и время взаимодействия между компонентами **MailD core** и **Sender**. Модуль **drweb-milter** использует наибольшее из значений параметра **IPCTimeout** и параметра **ProcessingTimeout** из своих настроек (секция `[Milter]`).
2. В **асинхронном режиме**:

   Принятое письмо сохраняется во внутренней очереди **MailD core** и **Receiver** сразу же отвечает отправителю кодом `SMTP 250` о том, что письмо находится в очереди и он снял с себя ответственность за его доставку. **MailD core**, передавая обработанное письмо в **Sender** для доставки получателю, также снимает с себя ответственность за доставку. Далее компонент **Sender** либо сразу и без проблем отправляет письмо далее, либо (в случае задержек в канале или невозможности подключиться к целевому почтовому серверу) переходит к отложенной отправке.

   В этом режиме величина тайм-аута **IPCTimeout** должна быть соизмерима со средним временем обработки писем подключаемыми модулями.
3. В **синхронном режиме**:

   Компонент **Receiver** будет ожидать, и не возвращать ответа отправителю, пока письмо не будет обработано всеми подключаемыми модулями и не отправлено далее через **Sender**. В случае если в течение периода времени, меньшего **IPCTimeout** на 1 секунду (модуль **drweb-milter** использует в этом случае наибольшее из значений параметра **IPCTimeout** и параметра **ProcessingTimeout** из своих настроек) компонент **Sender** не успеет отправить письмо, то он пропускает все попытки подключиться к целевому MTA (если их

перечислено несколько в параметрах **Address** или **Router** в секции [Sender]) и сразу переходит к отложенной отправке. При этом компонент **Receiver** вернет отправителю специфический ответ SMTP 250 Maild Error, который означает, что письмо находится в очереди на отправку и будет отправлено, как только исчезнут проблемы с подключением к целевому MTA. Сообщение "Maild Error" говорит о том, что это – непредвиденная для синхронного режима ситуация (режим расчитан на быструю обработку всего входящего трафика и быструю его передачу в целевой MTA).

Также в этом случае в журнале возможны ошибки вида "ERROR Broken pipe", свидетельствующие о том, что компонент **Sender** пытался вернуть **MailD core** отчет об отправке сообщения через соединение, которое уже было закрыто по истечению тайм-аута.

Общие рекомендации:

1. Синхронный режим **не предназначен для работы под интенсивной нагрузкой**. Его рекомендуется использовать только в том случае, если **Dr.Web MailD** выступает в качестве локального почтового фильтра, и **ни в коем случае не использовать**, если **Dr.Web MailD** выступает в качестве высоконагруженного SMTP-шлюза.
2. Если в почтовом трафике, проходящем через **Dr.Web MailD**, имеется большая доля писем с большим количеством вложений (или вложений, больших по размеру), а также если вероятны задержки в канале или какие-либо проблемы на стороне целевого MTA, то лучше выбирать асинхронный режим работы. Также не следует помещать в очередь **before-queue** подключаемые модули, требующие потенциально длительное время для обработки писем.
3. В случае работы в синхронном режиме желательно не уменьшать значение тайм-аута **IPCTimeout**, заданное по умолчанию, а при изменении этого значения соизмерять его со временем обработки письма подключаемыми модулями (величина **IPCTimeout** всегда должна быть больше, поскольку срабатывание тайм-аута прямо во время проверки может привести к потере письма и недоставки его получателю). Также в этом режиме предполагается, что нет никаких задержек при отправке почты через компонент **Sender** к целевому MTA (он должен быть доступен и корректно настроен).

Особенности работы с MTA, интегрированными по протоколу Milter:

1. Если **Dr.Web MailD** подключен к MTA по протоколу Milter (в качестве **Receiver** используется модуль **drweb-milter**), то на порядок возвращения проверенного письма обратно в очередь MTA влияет только значение параметра **CanChangeBody** (секция [Milter] конфигурационного файла);
2. Если имеются подключаемые модули, расположенные в очереди **BeforeQueueFilters** (т.е. идет работа в **синхронном** режиме), параметр **CanChangeBody = Yes** и при обработке писем имеются существенные задержки (например, какой-то из модулей не успевает обработать письмо за период времени, заданный в параметре **ProcessingTimeout**), то в MTA возвращается код ответа SMTP, заданный в параметре **ProcessingError** (все перечисленные параметры расположены в секции [Milter] конфигурационного файла);
3. Если идет работа в **синхронном** режиме, но параметр **CanChangeBody = No**, и в этом случае при обработке писем не отвечает какой-либо из компонентов (**MailD core** или **Sender**), то в MTA возвращается код ответа SMTP 451;
4. Если имеются подключаемые модули, расположенные в очереди **AfterQueueFilters** (т.е. идет работа в **асинхронном** режиме), параметр **CanChangeBody = Yes** и при обработке писем наблюдаются существенные задержки (например, какой-то из модулей не успевает обработать письмо за период времени, заданный в параметре **ProcessingTimeout**), то по истечению периода времени, заданного в параметре **ProcessingTimeout**, в MTA возвращается код ответа SMTP 250 queued, но письмо обрабатывается далее и отправляется через **Sender** (модуль **drweb-sender**). Таким образом, в случае если параметр **CanChangeBody = Yes**, то асинхронный режим нежелателен, т.к. не даст выигрыша ни по скорости (поскольку требуется время на дополнение сохранение служебных файлов сообщений в хранилище), ни по надежности обработки;

5. Если идет работа в **асинхронном** режиме, но параметр `CanChangeBody = No`, и при обработке ни отвечает какой-либо из компонентов (**MailD core** или **Sender**), то письмо теряется, а в MTA возвращается код SMTP `250 queued`, следовательно, **этот вариант настроек крайне нежелателен!**
6. Таким образом, в случае подключения к MTA по протоколу `Milter` асинхронного режима (т.е. размещения подключаемых модулей в очереди **`AfterQueue`**) следует избегать, поскольку он не дает никаких преимуществ в обработке писем.

Кроме того, при необходимости оптимизации процесса обработки сообщений (в случае возникновения больших задержек, очередей на приеме или отправке, зависаний, нехватке системных ресурсов и т.п.) рекомендуется провести комплекс мероприятий по оптимизации работы и использования системных ресурсов.

Используемые модули

Модули, образующие комплексный компонент обработки почты **Dr.Web MailD**, перечислены в таблице ниже:

Модуль	Компонент	Назначение
Обязательные модули		
`drweb-maild`	**MailD core**	Центральный модуль, обеспечивающий работу комплексного компонента обработки почты **Dr.Web MailD**
`drweb-notifier`	**Notifier**	Модуль формирования уведомлений, оповещений, отчетов и DSN
`drweb-receiver`	**Receiver** (SMTP/LMTP)	Модуль взаимодействия с MTA (приема входящих писем) по протоколу SMTP/LMTP
`drweb-sender`	**Sender** (SMTP/LMTP)	Модуль взаимодействия с MTA (отправки исходящих писем) по протоколу SMTP/LMTP Так как этот модуль имеет возможность передавать письма непосредственно локально установленной почтовой системе, он используется практически во всех схемах интеграции **Dr.Web MailD** с MTA
`drweb-proxy-client`	**Proxy client**	Клиентский модуль проксирования для кластеризации **Dr.Web MailD**. Используется в качестве заглушки (stub) на узле кластера, предназначенного для приема и передачи почты (т.е. там, где установлены компоненты **Sender** и **Receiver**)
`drweb-proxy-server`	**Proxy server**	Серверный модуль проксирования для кластеризации **Dr.Web MailD**. Используется в качестве заглушки (stub) на узле кластера, предназначенного для обработки почты (т.е. там, где установлен центральный компонент **MailD core**, встроенная база данных **MailDB** и подключаемые модули)
`drweb-imap`	**IMAP filter**	Модуль проксирования MDA по протоколу IMAP (проверка писем при их передаче от MDA к MUA)
`drweb-pop3`	**POP3 filter**	Модуль проксирования MDA по протоколу POP3 (проверка писем при их передаче от MDA к MUA)
Опциональные (используются для сопряжения со специфическими почтовыми системами, могут отсутствовать)		
`drweb-zmailer`	**Receiver**	Модуль интеграции с почтовой системой **Zmailer** (прием писем для обработки)

Модуль	Компонент	Назначение
	(**Zmailer**)	
drweb-qmail	**Receiver** (**Qmail**)	Модуль интеграции с почтовой системой **Qmail** (прием писем для обработки)
drweb-courier	**Receiver** (**Courier**)	Модуль интеграции с почтовой системой **Courier** (прием писем для обработки)
drweb-cgp-sender	**Sender** (**CommuniGate Pro**)	Модуль интеграции с почтовой системой **CommuniGate Pro** (отправка обработанных писем)
drweb-cgp-receiver	**Receiver** (**CommuniGate Pro**)	Модуль интеграции с почтовой системой **CommuniGate Pro** (прием писем для обработки)
drweb-milter	**Receiver** (`Milter`)	Модуль интеграции с почтовыми системами через протокол `Milter`
Утилиты		
drweb-inject		Утилита принудительной отправки писем
drweb-lookup		Утилита проверки корректности Lookup
drweb-qcontrol		Утилита управления **Карантином**
drweb-qp		Утилита управления **Карантином** (для работы с DBI). Не предназначена для запуска вручную

Все перечисленные в таблице модули располагаются в каталоге `%bin_dir`.

Параметры командной строки

Параметры командной строки

Как и для любых UNIX-программ, для всех модулей, входящих в состав **Dr.Web MailD**, предусмотрены параметры командной строки. Формат командной строки для запуска модулей программного комплекса следующий:

```
<название модуля> [<параметры>] <сокет Агента>
```

где:

- **<название модуля>** – название модуля;
- `<параметры>` – необязательные параметры командной строки;
- `<сокет Агента>` – сокет, через который модуль при запуске будет получать конфигурационную информацию от компонента **Dr.Web Agent**.

Общие параметры

Все текущей версии все модули, входящие в состав **Dr.Web MailD**, поддерживают следующие параметры командной строки:

Краткий вариант	Расширенный вариант	Аргументы
`-h`	`--help`	
Описание: Вывод на экран консоли краткой справки по имеющимся параметрам командной строки и завершение работы модуля		
`-v`	`--version`	
Описание: Вывод на экран консоли информации о текущей версии модуля и завершение работы модуля		

Краткий вариант	Расширенный вариант	Аргументы
-l	--level	<уровень>
Описание: Уровень детализации ведения журнала запуска компонента (значение по умолчанию: info)		
-t	--timeout	<число секунд>
Описание: Максимальное время ожидания получения конфигурационных данных от **Dr.Web Agent**		
	--log-name	<имя компонента>
Описание: Имя компонента, под которым он будет выводить сообщения в журнал (лог)		
	--component	<имя компонента>
Описание: Имя компонента, под которым модуль будет обращаться к **Dr.Web Agent** для получения конфигурации. Обратите внимание, что этот параметр командной строки отсутствует у модуля **drweb-zmailer!**		
	--check-only	
Описание: Запуск модуля в режиме проверки конфигурации. Для корректной функциональности опции предварительно должен быть запущен **Dr.Web Agent**. При успешной проверке конфигурации на консоль выводится сообщение "Options OK", а при неудаче выводится описание проблемы и сообщение "Options ERROR". Обратите внимание, что этот параметр командной строки отсутствует у модуля **drweb-zmailer!**		

Пример:

Введенная в командной строке консоли операционной системы команда

```
$ drweb-maild -t 30 local:%var_dir/ipc/.agent
```

Запускает модуль компонента **MailD core** со временем ожидания конфигурационных данных в 30 секунд и указанием на сокет **Dr.Web Agent** `local:%var_dir/ipc/.agent`.

Параметры, специфические для модулей

У различных модулей, входящих в состав **Dr.Web MailD**, кроме общих, имеются также дополнительные параметры командной строки, зависящие от специфики модуля. Исключение составляют модули **drweb-notifier** и **drweb-proxy-client**, которые не имеют параметров кроме тех, которые перечислены выше.

1. **drweb-maild**

Специфические параметры командной строки этого модуля используются для проверки корректности заданных Правил обработки почты:

Краткий вариант	Расширенный вариант	Аргументы
-s	--sender	<почтовый адрес>
Описание: Адрес отправителя письма (из конверта)		
-r	--recipient	<почтовый адрес>
Описание: Адрес получателя письма (из конверта). Для задания нескольких получателей нужно несколько раз указать данный параметр		
-b	--block	<имя объекта>

Краткий вариант	Расширенный вариант	Аргументы
Описание: Имя блокирующего объекта, найденного в письме (например, название вируса). Для задания нескольких блокирующих объектов нужно несколько раз указать данный параметр		
	`--client-ip`	`<IP-адрес>`
Описание: IP-адрес клиента, от которого получено письмо		
	`--server-ip`	`<IP-адрес>`
Описание: IP-адрес интерфейса сервера, на который получено письмо		
	`--client-port`	`<номер порта>`
Описание: Порт клиента, с которого было получено письмо		
	`--server-port`	`<номер порта>`
Описание: Порт сервера, на который было получено письмо		
	`--server-us`	`<UNIX-сокет>`
Описание: Название UNIX-сокета сервера, на который получено письмо		
	`--id`	`<идентификатор>`
Описание: Уникальный идентификатор компонента **Receiver**, от которого получено письмо		
	`--auth`	
Описание: Отметка, что письмо получено от авторизованного пользователя		
	`--size`	`<размер>`
Описание: Размер проверяемого письма (значение имеет тип size)		
	`--score`	`<счет>`
Описание: Счет, присвоенный письму (число)		
	`--md-client`	`<имя Клиента MailDesk>`
Описание: Уникальный идентификатор **Клиента MailDesk**		

2. У остальных модулей (кроме **`drweb-zmailer`**) имеются два специфических параметра, используемых для организации работы с несколькими компонентами **Sender** и **Receiver** одновременно:

Краткий вариант	Расширенный вариант	Аргументы
	`--unique-id`	`<идентификатор>`
Описание: Уникальный идентификатор компонента. Данная настройка позволяет компоненту **MailD core** осуществлять работу с несколькими экземплярами компонентов **Receiver** и **Sender**. Для этого каждый новый **Receiver** и **Sender** должен запускаться со своим уникальным идентификатором. Для отправки письма будет выбираться **Sender** с тем же идентификатором, что и у **Receiver**, или **Sender** по умолчанию, если для **Receiver** не был найден соответствующий **Sender**. Список доступных компонентов **Sender** переинициализируется через сигнал `SIGHUP` Ниже указан смысл идентификатора компонента для каждого из модулей с точки зрения **MailD core**: • `drweb-receiver` – Идентификатор компонента **Receiver**;		

Краткий вариант	Расширенный вариант	Аргументы
• `drweb-sender` – Идентификатор компонента **Sender**; • `drweb-proxy-server` – Общий идентификатор пары компонентов **Receiver** и **Sender**, взаимодействующих с **MailD core** через этот компонент (см. Проксирование) • `drweb-imap` – Идентификатор компонента **IMAP filter** (используется аналогично идентификатору компонента **Receiver** для поиска соответствующего **Sender**); • `drweb-pop3` – Идентификатор компонента **POP3 filter** (используется аналогично идентификатору компонента **Receiver** для поиска соответствующего **Sender**); • `drweb-milter` – Идентификатор компонента **Receiver**; • `drweb-cgp-receiver` – Идентификатор компонента **Receiver**; • `drweb-cgp-sender` – Идентификатор компонента **Sender**; • `drweb-courier` – Идентификатор компонента **Receiver**; • `drweb-qmail` – Идентификатор компонента **Receiver**.		
	`--section`	`<имя секции>`
Описание: Имя секции в конфигурационном файле, из которой данный модуль будет извлекать свои настройки компонента. Если данный параметр не указан, будет использована секция компонента по умолчанию. Ниже указаны секции по умолчанию для каждого модуля: • `drweb-receiver – [Receiver]` • `drweb-sender – [Sender]` • `drweb-proxy-server – [ProxyServer]` • `drweb-imap – [IMAP]` • `drweb-pop3 – [POP3]` • `drweb-milter – [Milter]` • `drweb-cgp-receiver – [CgpReceiver]` • `drweb-cgp-sender – [CgpSender]` • `drweb-courier – [Courier]` • `drweb-qmail – [Qmail]`		

3. **`drweb-zmailer`**

Специфические параметры командной строки этого модуля:

Краткий вариант	Расширенный вариант	Аргументы
`-u`	`--user`	`<имя пользователя>`
Описание: Имя учетной записи пользователя, с правами которого запущен модуль **`drweb-maild`**. Обратите внимание, что если этот параметр не указан, то **`drweb-zmailer`** запустится с правами суперпользователя `root`, что может привести к проблемам взаимодействия с модулем **`drweb-maild`**, если он работает не с правами суперпользователя `root`		
`-i`	`--ipclevel`	`<уровень подробности>`
Описание: Уровень подробности ведения журнала событий библиотеки IPC, используемый модулем **`drweb-zmailer`**. Допустимые значения: `quiet, error, alert, info, debug`		
`-f`	`--facility`	`<метка syslog>`
Описание: Используемая метка syslog (если журнал ведется при помощи системной службы **syslog**). Допустимые значения: `daemon, mail, local0, ..., local7`		

Краткий вариант	Расширенный вариант	Аргументы
-b	--basedir	<путь к каталогу>
Описание: Путь к основному каталогу, в котором расположены модули **Dr.Web MailD**		
	--id	<идентификатор>
Описание: Аналогичен параметру --unique-id у других компонентов (см. выше). С точки зрения **MailD core** трактуется как иентификатор компонента **Receiver**		
	--log-filename	<имя файла журнала>
Описание: Имя используемого файла журнала или syslog, если журнал ведется при помощи системной службы **syslog**		
	--file	<путь к файлу>
Описание: Путь к файлу, который должен быть обработан при запуске модуля		
	--hash	<значение>
Описание: Значение параметра **SecureHash** из секции [Sender] основного конфигурационного файла **Dr.Web MailD**		
	--interface	<0|1>
Описание: Обозначение версии используемого **smtpserver:** 0 – для версии 2.99.55 или более ранней, 1 – для версии 2.99.56 или более поздней		
-e	--error-action	<действие>
Описание: Действие, применяемое в случае если произойдет внутренняя ошибка в модуле при обработке письма. Допустимые значения: pass, reject, discard, tempfail		
-Z		<путь к файлу>
Описание: Путь к конфигурационному файлу **ZMailer**, который будет игнорироваться		

Обратите внимание, что параметры командной строки утилит в данном разделе не рассматриваются. Параметры командной строки, используемые утилитами, указаны в описаниях этих утилит.

Обрабатываемые сигналы

Все постоянно находящиеся в памяти модули компонентов программного комплекса **Dr.Web для почтовых серверов UNIX** поддерживают обработку следующих сигналов:

- SIGHUP – при получении этого сигнала компоненты перечитывают свои конфигурационные файлы. Если этот сигнал получает компонент **Dr.Web Monitor**, то конфигурационные файлы перечитываются модулями всех запущенных компонентов;
- SIGINT и SIGTERM – при получении любого из этих сигналов модули завершают свою работу.

Некоторые модули **Dr.Web MailD** поддерживают обработку дополнительных сигналов:

- модули, играющие роль компонента **Sender**, получив сигнал SIGUSR2, производят попытку отправить все сообщения, находящиеся во внутренней очереди и ожидающие отправки;

- модуль **drweb-receiver**, получив сигнал SIGUSR1, сохраняет в файл restrictions.txt внутреннюю статистику срабатывания **SMTP-ограничений**;
- все компоненты при получении сигнала SIGUSR1 сохраняют статистику по работе пулов потоков и соединений.

Файлы статистики сохраняются в каталог, указанный в значении параметра **BaseDir** секции [General] конфигурационного файла **Dr.Web MailD**. Статистика работы пулов потоков сохраняется также в виде записей в файлы журналов на уровне Debug. Следовательно, если установлен менее подробный уровень подробности журнала, то статистические записи не будут сохраняться в журнал.

О формате внутренней статистики **Dr.Web MailD** см. в следующем разделе.

Обратите внимание, что изменение значений некоторых параметров из конфигурационного файла **Dr.Web MailD**, невозможно применить посредством перечитывания конфигурационных файлов после посылки сигнала SIGHUP (в описании конфигурационного файла такие параметры отмечены особо). В случае необходимости изменения этих параметров необходимо перезапустить **Dr.Web MailD**.

Также обратите внимание, что компонент **Dr.Web Monitor** и компонент **Dr.Web Agent**, управляющие работой **Dr.Web MailD**, не обрабатывают сигналы SIGUSR1 и SIGUSR2. Если послать эти сигналы модулям этих компонентов, то не произойдет попытки отправки сообщений или сброса статистики, а произойдет остановка их работы.

Журнал работы

Информация в журнал может выводиться как через демон системной службы **syslog**, так и в обычный файл. Соответствующие настройки управления выводом задаются в конфигурационном файле **Dr.Web MailD**, в секции [Logging].

Формат каждой выводимой строки имеет следующий вид (в случае вывода через **syslog**):

```
'['tid']' name[.sub] level [sid(/mta-id)] text
```

где:

- tid – идентификатор потока, выводящего строку;
- name – название компонента, производящего вывод – например, название подключаемого модуля или модуля компонента;
- sub – название службы компонента, который производит вывод.

  К самым важным службам относятся следующие:

  - ipc – служба межпроцессного взаимодействия;
  - thrN – служба поддержки пула потоков с номером N;
  - report – служба поддержки отчетов;
  - ldap, odbc, oracle, sqlite, mysql, postgres, cdb, berkeley, firebird – служба поддержки соответствующих Lookup;
  - control – служба поддержки интерактивного управления;
  - parser – служба разбора шаблонов отчетов;
  - MRS – служба приема сообщения по протоколу SMTP/LMTP;
  - smtp – служба отправления письма по протоколу SMTP;
  - lmtp – служба отправления письма по протоколу LMTP;
  - pipe – служба отправления письма через pipe;
  - queue – служба обработки внутренней очереди;

- `level` – уровень подробности журнала работы. Возможны следующие значения: `FATAL`, `ERROR`, `WARN`, `INFO`, `DEBUG`.
- `sid` – идентификатор сессии для сообщения, к которому относится данная строка журнала. Номер выводится в шестнадцатеричном виде;
- `mta-id` – идентификатор сообщения внутри MTA, от которого получено письмо. Выводится, если **Dr.Web MailD** работает не в режиме **SMTP/LMTP-прокси** и из MTA удалось получить данную информацию;
- `text` – собственно текст выводимого сообщения.

При запуске любого модуля уровень подробности его журнала по умолчанию устанавливается в `INFO` до момента получения конфигурации от **Dr.Web Agent** и установки уровня, обозначенного в конфигурационном файле. Если необходимо при загрузке модуля сразу установить его журнал в уровень `DEBUG` (например, для получения информации о загруженных от **Dr.Web Agent** параметрах), то для этого служит параметр командной строки `--level`, установив который в `debug`, можно получить требуемую информацию.

Внутренняя статистика работы

Dr.Web MailD позволяет вести следующие виды внутренней статистики работы:

1. Статистика срабатывания **SMTP-ограничений**.
2. Статистика пулов потоков и соединений.

Рекомендуется периодически проводить сбор внутренней статистикй работы **Dr.Web MailD** для оценки нагрузки на него и его пропускной способности. Полученную информацию рекомендуется использовать для оптимизации работы и использования системных ресурсов.

Статистика срабатывания SMTP-ограничений

Статистика срабатывания **SMTP-ограничений** (ограничений для SMTP-сессии, указанных в настройках компонента **Receiver** в секции `[Receiver]` основного конфигурационного файла **Dr.Web MailD**), сохраняется в файл `restrictions.txt`. Статистика сохраняется только в случае если явно разрешен ее сбор. Включением сбора статистики управляет параметр **`RestrictionStat`**, находящийся в этой же секции конфигурационного файла. Статистические записи всегда добавляются в конец файла.

Каждая запись начинается со следующих строк:

```
==================================
start:  Tue Oct 9 14:44:15 2008
curr:   Tue Oct 9 14:44:29 2008
period: 0d 0h 0m 14s
```

в которых указывается время предыдущего сбора статистики, время сброса в файл текущей статистической записи, а также временной период, прошедший между этими моментами.

Далее, после пустой строки, перечисляются все ограничения, настроенные для SMTP-сессии, по одному на строку. Примерный вид строки со статистикой ограничения показан ниже:

```
reject_unknown_domain: total: 19    trusted: 0    reject: 0    tempfail: 0
```

Для каждого ограничения указывается:

- название ограничения;
- общее количенство писем, для которых выполнялась проверка этого ограничения `(total:)`;
- количество писем, которым был присвоен статус `trusted` в результате срабатывания этого ограничения `(trusted:)`;

- количество писем, которые были отклонены в результате срабатывания этого ограничения (`reject:`);
- количество писем, по которым клиенту вернулся SMTP-ответ **`Tempfail`** (`tempfail:`).

Обратите внимание, что статистика не суммируется. Каждая статистическая запись фиксирует количество сработавших ограничений в период между текущим и предыдущим моментами сохранения статистики. Статистика срабатывания ограничений фиксируется только в случае если **Dr.Web MailD** работает в режиме **SMTP/LMTP-прокси**.

Статистика пулов потоков и соединений

Статистика по работе пула потоков (соединений) собирается, если ее явно включить в настройках этого пула (например, параметры **`InPoolOptions`** и **`OutPoolOptions`** в секции `[Maild]` конфигурационного файла **Dr.Web MailD**), установив дополнительную настройку **`stat`** = `yes`.

Пример:

```
InPoolOptions = auto, stat = yes
```

Обратите внимание, что для компонентов **Sender** и **Receiver** статистика собирается всегда, поскольку для настроек их пулов дополнительные параметры (такие как `timeout`, `stat` и т.п.) не задаются, а если заданы, то не имеют никакого эффекта.

Накопленная статистика сбрасывается в файл журнала как сообщение уровня `Debug` при получении сигнала `SIGUSR1`, а также при завершении работы **Dr.Web MailD**. Обратите внимание, что если у вас уровень подробности ведения журнала установлен в менее подробный уровень, чем `Debug`, статистика в него попадать не будет. Статистическая запись имеет следующий вид (пример):

```
size = 50, active = 0, pending = 0, min = 50, max = 500, threshold = 50
```

Здесь указывается:

- `size` – текущий размер пула потоков (количество потоков в пуле);
- `active` – количество потоков, активных в пуле на момент сохранения записи;
- `pending` – количество задач, ожидающих освобождения потока в пуле на момент сохранения записи;
- `min` – минимальное возможное число потоков в пуле;
- `max` – максимальное возможное число потоков в пуле;
- `threshold` – порог, превышение которого приводит к увеличению количества потоков в пуле.

Статистика не имеет накапливающего характера, при получении сигнала в журнале всегда фиксируется текущее состояние пула.

Кроме того, для некоторых пулов также формируются отдельные файлы со статистикой, статистические записи в которые также сбрасываются по получению сигнала `SIGUSR1` и при завершении работы **Dr.Web MailD**.

Имена файлов статистики соответствуют следующему шаблону:

- `name_[callback_](cli|srv)[.unique-id].txt` – для статистики по соединениям;
- `name_[callback_](thr[N])[.unique-id].txt` – для статистики по потокам.

где:

- `name` – имя компонента (является именем соответствующего модуля без части "`drweb-`").
- `callback` – указывается для `callback`-интерфейса компонента **Receiver**.

- `cli` – для соединений клиентской части.
- `srv` – для соединений серверной части.
- `unique-id` – указывается для модулей, запущенных с уникальным идентификатором.
- `thr` – указывается для пула потоков.

Если такой файл уже существует, то статистика будет добавлена в конец файла.

Каждая запись начинается со следующих строк:

```
==================================
start:  Tue Oct 9 14:44:15 2008
curr:   Tue Oct 9 14:44:29 2008
period: 0d 0h 0m 14s
```

в которых указывается время предыдущего сбора статистики, время сброса в файл текущей статистической записи, а также временной период, прошедший между этими моментами.

Для `srv` затем указывается число закрытых и созданных соединений, а также максимальное число элементов в различных очередях:

```
closed: 0 (0 num/sec)
total created = 0 (0 num/sec)
max rea = 0 est = 0 don = 0 act = 0
```

Для `cli` также указывается число созданных соединений по запросу, число закрытых соединений по истечении времени ожидания, среднее и текущее число соединений:

```
created on request = 0 (0 num/sec)
closed by timeout = 0 (0 num/sec)
avg number = 0
current = 2
```

Для `thr` вывод имеет вид:

```
min = 2 max = 2147483647 type = 0 freetime = 120
busy max = 0 avg = 0
requests for new threads = 0 (0 num/sec)
creating fails = 0
max processing time = 0 ms; avg = 0 ms
curr = 2 busy = 0
```

Здесь указывается:

- в первой строке – минимальное/максимальное число потоков в пуле, тип пула, время в секундах, в течение которого дополнительный поток будет в бездействии перед тем, как завершится;
- во второй строке – максимальное и среднее число занятых одновременно потоков;
- в третьей строке – число запросов на создание дополнительных потоков и частота таких запросов;
- в четвертой строке – число неудавшихся попыток создания потоков (скорее всего из-за нехватки ресурсов);
- в пятой строке – максимальное и среднее время обработки одного запроса в миллисекундах;
- в шестой и последней строке – текущее число потоков в пуле и какое их количество сейчас занято обработкой.

Стандарты реализации

Стандарты RFC, которым соответствует реализация **Dr.Web MailD**:

№	Наименование
1123	Requirements for Internet Hosts - Application and Support
1652	SMTP Service Extension for 8bit-MIME transport
1830	SMTP Service Extensions for Transmission of Large and Binary MIME Messages
1870	SMTP Service Extension for Message Size Declaration
1894	An Extensible Message Format for Delivery Status Notifications
2033	Local Mail Transfer Protocol
2034	SMTP Service Extension for Returning Enhanced Error Codes
2045 – 2049	Multipurpose Internet Mail Extensions
2222	Simple Authentication and Security Layer (SASL)
2231	MIME Value and Encoded Word Extensions
2245	Anonymous SASL Mechanism
2289	A One-Time Password System
2444	The One-Time-Password SASL Mechanism
2505	Anti-Spam Recommendations for SMTP MTAs
2646	The Text/Plain Format Parameter
2821	Simple Mail Transfer Protocol
2822	Internet Message Format
2831	Using Digest Authentication as a SASL Mechanism
2945	The SRP Authentication and Key Exchange System
3174	US Secure Hash Algorithm 1 (SHA1)

Настройка и запуск

Dr.Web для почтовых серверов UNIX может запускаться с настройками по умолчанию, но для оптимальной работы программного комплекса рекомендуется настроить его для соответствия конкретным требованиям и условиями эксплуатации. Все настройки **Dr.Web для почтовых серверов UNIX** содержатся в трех конфигурационных файлах, расположенных в каталоге `%etc_dir`:

- `maild_<MTA>.conf` – общие настройки **Dr.Web MailD**;
- `agent.conf` – настройки **Dr.Web Agent**;
- `monitor.conf` – настройки **Dr.Web Monitor**.

Базовую настройку **Dr.Web для почтовых серверов UNIX** (при условии расположения всех файлов программного комплекса в каталогах по умолчанию) можно осуществить с помощью скрипта `configure.pl`, по умолчанию располагающегося в каталоге `%bin_dir/maild/scripts/`.

После запуска скрипт запросит значения основных параметров и запишет их в конфигурационный файл `maild_<MTA>.conf`.

Остальные параметры, необходимые для взаимодействия с почтовой системой, нужно будет настроить отдельно, вручную отредактировав конфигурационный файл **Dr.Web MailD**

maild_<MTA>.conf.

Для настройки почтовой системы следует запустить скрипт configure_mta.sh. Этот скрипт отвечает за настройку взаимодействия между программным комплексом **Dr.Web для почтовых серверов UNIX** и используемой почтовой системой. При запуске он проверит, установлена ли нужная почтовая система. В случае ее отсутствия скрипт завершит свою работу, а в случае обнаружения предложит ответить в интерактивном режиме на ряд вопросов про отдельные настройки конфигурации используемого MTA. В соответствии с полученными ответами скрипт вносит требуемые изменения в параметры соответствующих конфигурационных файлов.

Обратите внимание, что часть имени файла <MTA> зависит от названия MTA, с которым сопряжен **Dr.Web для почтовых серверов UNIX**.

Конфигурационные файлы Dr.Web MailD

Все настройки комплексного компонента **Dr.Web MailD**, включая его взаимодействие с почтовыми системами и использование подключаемых модулей для проверки почтовых сообщений, задаются в основном конфигурационном файле %etc_dir/maild_<MTA>.conf.

Основной конфигурационный файл

- Устройство (структура) конфигурационного файла и краткое описание принципов задания параметров даны разделе Конфигурационные файлы.
- Специальные типы параметров, используемые в настройках **Dr.Web MailD**, рассмотрены в разделе Специальные типы параметров.
- Правила описания Lookup, LookupLite и обращений к хранилищам данных (Storage) описаны в разделе Lookup.
- Файл состоит из перечня секций.

Обратите внимание, что часть имени файла <MTA> зависит от названия MTA, с которым сопряжен **Dr.Web для почтовых серверов UNIX**.

Конфигурационные файлы подключаемых модулей

Каждый из используемых **Dr.Web MailD** подключаемых модулей имеет собственный конфигурационный файл. Так же как и основной конфигурационный файл, конфигурационные файлы подключаемых модулей по умолчанию располагаются в каталоге %etc_dir. Конфигурационный файл конкретного подключаемого модуля по умолчанию имеет имя plugin_<name>.conf, где <name> – имя подключаемого модуля. Например, конфигурационный файл модуля **Drweb** имеет имя plugin_drweb.conf.

При необходимости в секции [Filters] основного конфигурационного файла **Dr.Web MailD** можно настроить каждый подключаемый модуль на использование конфигурационных файлов, название которых не соответствует схеме, принятой по умолчанию.

Специальные типы параметров

В этом разделе подробно рассмотрены следующие специальные типы параметров:

- **настройки TLS/SSL (TLSSettings)** — настройки для работы шифрованного соединения с использованием криптографических протоколов TLS и SSL.
- **настройки пула (pool options)** — настройки пула потоков.

Типы параметров **Lookup**, **LookupLite**, **Storage** рассмотрены в разделе Lookup.

Настройки TLS/SSL

Настройки для работы шифрованного соединения с использованием криптографических протоколов TLS и SSL.

Настройки задаются в формате: `НАЗВАНИЕ ЗНАЧЕНИЕ` и разделяются запятыми.

Если в качестве `ЗНАЧЕНИЯ` указан путь к файлу, то он будет зависеть от регистра (по соглашению, принятому для UNIX-побных ОС). В данной версии поддерживаются следующие настройки:

- `use_sslv2 {yes | no}` — использовать или не использовать протокол SSLv2. По умолчанию данный протокол отключен, т.к. не является безопасным.
- `use_sslv3 {yes | no}` — использовать или не использовать протокол SSLv3. По умолчанию протокол SSLv3 включен.
- `use_tlsv1 {yes | no}` — использовать или не использовать протокол TLSv1. По умолчанию протокол TLSv1 включен.
- `private_key_file {путь к файлу}` — абсолютный путь к файлу с закрытым ключом. Ключ должен быть в формате PEM. Поддерживается шифрование ключа. Данный параметр является обязательным для заполнения при настройке серверной части. По умолчанию значение этого параметра не задано.
- `private_key_password {строка}` — пароль для ключа, указанного в параметре **`private_key_file`**. По умолчанию значение данного параметра не задано.
- `certificate {путь к файлу}` — путь к файлу сертификата с подписанным открытым ключом. Значение данного параметра должно задаваться в паре со значением параметра **`private_key_file`**. Данный параметр является обязательным для заполнения для серверной части. По умолчанию значение параметра не задано.
- `verify_mode {none | peer | client_once | fail_if_no_peer_cert}` - задает режим проверки сертификата собеседника. Можно использовать следующие настройки:
  - `none` — не проверять сертификат собеседника. Это значение установлено по умолчанию для серверных соединений;
  - `peer` — проверять сертификат собеседника. В клиентском режиме эта настройка игнорируется, если при использовании анонимного шифрования серверная сторона не выслала сертификат. Это значение установлено по умолчанию для клиентских соединений;
  - `client_once` — для серверной стороны запрашивать сертификат только при установлении соединения (не запрашивать сертификат при повторении процедуры TLS handshake в рамках уже установленного соединения). Данное значение можно использовать только вместе с настройкой `peer`.
  - `fail_if_no_peer_cert` — для серверной стороны воспринимать отсутствие сертификата у клиента как ошибку. Данное значение можно использовать только вместе с настройкой `peer`.

  Примеры:

  ```
  verify_mode peer,
  verify_mode client_once
  verify_mode none
  ```

  Если `peer` и `none` встречаются в одном наборе настроек, то используется последнее указанное значение.
- **`verify_ca`** `{путь к файлу}` — абсолютный путь к файлу, где находятся CA-сертификаты в PEM-формате. Данные сертификаты используются при проверке валидности сертификата собеседника.
- **`cipher_list`** `{строка}` — список разрешенных алгоритмов шифрования. Формат списка алгоритмов шифрования можно узнать по команде `man ciphers` (для этого должен быть установлен **OpenSSL**).

Настройки пула (pool options)

Имеют комбинированный вид. Параметры настройки перечисляются через запятую

Первым определяется количество потоков в пуле:

- `auto` — количество потоков определяется автоматически в зависимости от загрузки системы;
- `N` — целое положительное число. В пуле всегда будет присутствовать ровно `N` активных потоков;
- `N-M` — целые положительное значения, и `M` ≥ `N`. В пуле, в зависимости от нагрузки, будет активно не менее `N` и не более `M` потоков (если требуется обеспечить неизменное число потоков, установите `M` = `N`).

Далее определяются дополнительные параметры:

- `timeout = {время}` — если поток не становится активным в течение заданного периода времени, поток закрывается. Этот параметр не влияет на первые `N` потоков (ожидающих запросов бесконечно). Значение по умолчанию: `2m`
- `stat = {yes|no}` — сбор статистики по потокам в пуле. Статистика сохраняется при получении системного сигнала `SIGUSR1` и при завершении работы **Dr.Web MailD** в специальном файле в каталоге, определенном значением параметра **`BaseDir`** секции `[General]`. Значение по умолчанию: `no`
- `log_level` — уровень подробности журнала для потоков в пуле. Если значение не задано, используется значение параметра **`LogLevel`** секции `[Logging]`.
- `stop_timeout = {время}` — максимальное время ожидания остановки работающего потока (например при завершении работы программы или когда требуется уменьшить число потоков в пуле).

Пример:

```
InPoolOptions = auto
```

В данном примере число потоков определяется автоматически, внутренняя статистика по потокам в пуле не ведется (если это не **Receiver** или **Sender**, поскольку для них она ведется всегда).

Пример: (для **Notifier**)

```
PoolOptions = 25, stat=yes
```

В данном примере число потоков в пуле потоков всегда ровно 25, внутренняя статистика по потокам в пуле ведется.

Для компонентов **Sender** и **Receiver** значение `auto` эквивалентно значению `2-500`, а для прочих компонентов **Dr.Web MailD** оно эквивалентно значению `2-1000`. Следует с осторожностью изменять количества потоков в пулах. Подробнее об этом см. в разделе Оптимизация работы и использования системных ресурсов.

Для компонентов **Sender** и **Receiver** дополнительные параметры пула (такие как `timeout`, `stat` и т.п.) не задаются, а если заданы, то не имеют никакого эффекта, так как **Sender** и **Receiver** всегда осуществляют сбор статистики по своим пулам потоков.

Lookup

`Lookup` – это обобщенный интерфейс поиска объектов и получения связанных с ними значений. Значения разделяются запятыми. Перед значением может стоять префикс, обозначающий тип `Lookup`, который отделяется двоеточием:

```
[prefix1:]value1, [prefix2:]value2, ...
```

Если префикс не указан, то значение используется непосредственно.

Использование специальных символов

В составе `Lookup`-запросов возможно применение следующих *специальных символов*:

- `$s` – будет заменен перед отправлением на запрашиваемый элемент. Например, если запрашивается адрес, то будет подставлен весь адрес (без угловых скобок), а если домен – то весь домен.
- `$d` – если запрашиваемым элементом является адрес, то из него будет выделено доменное имя и передано в качестве запроса. В противном случае подставляется весь запрос.
- `$u` – если запрашивается адрес, то будет выделено имя пользователя и передано в качестве запроса. Если параметр запрашивает домен, то передается пустая строка.
- `$$` – заменяется на одинарный `$`.

Пожалуйста, обратите внимание, что имеются случаи, когда значения некоторых специальных символов не могут быть определены при выполнении `Lookup` для подстановки значений в параметр. В первую очередь это касается параметров, используемых для проверки **SMTP-ограничений** компонентом **Receiver** (SMTP/LMTP) на этапе INTRO SMTP-сессии (подключение клиента). На этом этапе:

- Для всех проверок, связанных с принадлежностью IP-адреса клиента к списку сетей (см. параметр **`ProtectedNetworks`** в секции `[Maild]`):
  - `$s` – IP-адрес подключившегося клиента.
  - `$d`, `$u` – пустые, т.к. соответствующее значение не определено.
- Для всех проверок, связанных с принадлежностью к домена клиента списку доменов (см. параметр **`ProtectedDomains`** в секции `[Maild]`):
  - `$s` – доменное имя хоста, с которого подключился клиент (если удалось разрешить FQDN), иначе – его IP-адрес.
  - `$d = $s`.
  - `$u` – пустой, т.к. соответствующее значение не определено.

На последующих стадиях SMTP-сессии (MAIL FROM, RCPT TO) значения спецсимволов определены:

- `$s` – адрес user@domain целиком.
- `$d` – домен (часть domain адреса).
- `$u` – имя пользователя (часть user адреса).

Используемые типы префиксов

Существуют следующие варианты префикса, задающие источник данных для поиска:

- `value` – за ним указывается непосредственно искомое значение. Этот префикс не обязателен и подразумевается по умолчанию, если не указано иного префикса. Этот префикс может быть указан явно, если, к примеру, в искомом значении встречается символ ":".
- `file` – значение является путем к файлу. Каждое значение в файле должно находиться в новой строке. При поиске это один из самых быстрых вариантов, так как позволяет использовать сортировку и бинарный поиск.

Пожалуйста, обратите внимание, что только первые 2 типа префиксов могут быть использованы для параметров, имеющих специальный ограниченный тип LookupLite, а остальные типы, указанные ниже, там использовать запрещено

- `regex` – значение является регулярным выражением (совместимым с синтаксисом регулярных выражений **Perl**) – при проверке ищется подстрока, а не полное совпадение;
- `rfile` – значение является путем к файлу. Файл содержит набор регулярных выражений (совместимых с синтаксисом регулярных выражений **Perl**), каждое из которых должно находиться в новой строке. При проверке ищется подстрока, а не полное совпадение;

Пожалуйста, обратите внимание, что содержимое файлов, которые используются в качестве источников данных для `Lookup` типа **`file`** и **`rfile`**, никак не проверяется, поэтому перед использованием их в качестве источников данных, убедитесь в следующем:

- Файлы являются текстовыми (содержат только текстовые строки);
- В файлах отсутствуют пустые строки, а также строки-"мусор" (типа комментариев или разделителей), которые не могут использоваться в качестве значений для `Lookup`;
- Регулярные выражения (для **`rfile`**) составлены корректно в соответствии с синтаксисом регулярных выражений **Perl**;
- Файл не содержит лишних (избыточных) данных;
- Следите за размером файла, поскольку при чтении файлов большого размера может произвойти ошибка выделения памяти, что приведет к аварийному завершению работы **Dr.Web MailD**.

- `ldap` – значение представляет собой путь к поиску на LDAP-сервере;

  Формат значения следующий:

  ```
  [param1=val1|param2=val2|...|] ldap_url
  ```

  где `ldap_url` – это интернет-адрес LDAP-запроса.

  Пары `param=value` указываются в том случае, если в данном `Lookup`-запросе требуется переопределить параметры LDAP, заданные в секции `[LDAP]` конфигурационного файла **Dr.Web MailD**. Можно указывать только те параметры из этой секции, для которых явно написано, что это возможно (см. описание параметров секции). Для параметров, значение которых не переопределено в данном запросе, будут использованы значения, заданные в `[LDAP]`.

  URL LDAP-запроса выглядит следующим образом:

  ```
  ldap://hostport/dn[?attrs[?scope[?filter[?exts]]]]
  ```

  где:

  - `hostport` – имя хоста (возможно, вместе с номером порта, указанным через двоеточие);
  - `dn` – имя базы данных, в которой осуществляется поиск;
  - `attrs` – список атрибутов запроса, разделенных запятой;
  - `scope` – может принимать три значения: `base`, `one`, `sub`;
  - `filter` – название поискового фильтра;
  - `exts` – набор расширений LDAP и/или API.

  Пример:

  ```
  ldap://ldap.example.net/dc=example,dc=net?cn,sn?sub?(cn=*)
  ```

- `odbc`, `postgres`, `oracle`, `mysql`, `firebird`, `sqlite` – значение представляет собой SQL-запрос к соответствующей базе данных:

  ```
  [param1=val1|param2=val2|...|] sql_request
  ```

  `sql_request` – строка SQL-запроса к базе данных (источнику данных DSN в случае **ODBC**). Параметры подключения берутся из соответствующей секции конфигурационного файла **Dr.Web MailD** (соответственно `[ODBC]`, `[PostgeSQL]`, `[Oracle]`, `[MySQL]`, `[Firebird]`, `[SQLite]`). В качестве SQL-запроса должен использоваться запрос `SELECT` или любой запрос, возвращающий значение, включая вызов хранимой процедуры (stored procedure).

  Пары **`param`**`=value` указываются в том случае, если в данном `Lookup`-запросе требуется локально переопределить параметры обращения к БД, заданные в соответствующей секции параметров БД конфигурационного файла **Dr.Web MailD**. Можно указывать только те параметры из этой секции, для которых явно написано, что это возможно (см. описание параметров секции). Для параметров, значение которых не переопределено в данном запросе, будут использованы значения, заданные в секции. В запросе можно указывать *специальные символы*.

  Замечания:

  1. Обратите внимание, что если запрос `SELECT` будет извлекать записи, содержащие более одного поля (вида `SELECT field1,field2,... FROM ...`), то значениями `Lookup` будут являться все строки записей с перечнем значений полей в том виде, в каком их возвращает СУБД. В силу того, что разные СУБД могут по-разному формировать вывод записей, содержащих более одного поля, не рекомендуется использовать в `Lookup` запросы, возвращающие записи, содержащие более одного поля.
  2. `Lookup` типа `sqlite` предназначены для работы с базами данных **SQLite** версии 3.x. Пожалуйста, обратите внимание на особенности работы СУБД **SQLite**.

- `cdb` – значение представляет собой текстовое имя ключа в базе данных **CDB**.

  Базы данных **CDB** не поддерживают язык запросов SQL. Поэтому драйвер **CDB** эмулирует единственную команду SQL для унификации работы с `Lookup`:

  ```
  select * from @tablename where key='@string'
  ```

  где `@tablename` следует заменить на имя одного из файлов, которые были заданы в секции `[CDB]` конфигурационного файла **Dr.Web MailD** как источники данных.

  В запросе можно указывать *специальные символы*.

  Пример:

  ```
  cdb:skipdomains=file:/home/skipdomains.list|select * from my_file where
  key='$s'
  ```

  Обратите внимание, что параметр **`SkipDomains`**, локально переопределенный в данном `Lookup`, имеет тип LookupLite (т.е. является `Lookup`, для которого разрешены только префиксы `file:` и `value:`).

- `berkeley` – обеспечивает взаимодействие с **Berkeley DB**. Формат запроса аналогичен `cdb`. Используются параметры заданные в секции `[Berkeley]` конфигурационного файла **Dr.Web MailD**. В запросе можно указывать *специальные символы*.

Использование локального переопределения параметров

После префикса в `Lookup` можно (но не обязательно) указывать список локальных значений параметров **`SkipDomains`** и **`OnError`** для этого `Lookup`. Локальные значения параметров для `Lookup` задаются в формате:

```
NAME1 = VALUE1 | NAME2 = VALUE2 | ... |
```

где `NAME` – имя параметра (не зависит от регистра), а `VALUE` – значение параметра.

Если параметр локально не переопределен, то его значение будет совпадать со значением по умолчанию, или со значением, заданным в секции конфигурационного файла **Dr.Web MailD**, соответствующей префиксу `Lookup` (тип источника данных для поиска).

Обратите внимание, что использование параметра `SkipDomains` не имеет смысла использовать в `Lookup`, которые выполняются для определения значений параметров на тех этапах SMTP-сессии, когда не может быть определен домен (т.е. пуст спецсимвол `$d`). Домен не может быть определен на этапе INTRO SMTP-сессии (см. выше).

Особенности обработки Lookup

Обратите внимание, что при обработке `Lookup` **Dr.Web MailD** будет ожидать подключения к источнику данных (СУБД или LDAP-серверу) в течение заданного в настройках (или локально переопределенного в префиксе `Lookup`) тайм-аута, что может привести к замедлению работы **Dr.Web MailD** при нестабильном сетевом соединении или при неправильно указанных параметрах подключения. В случае если в течении тайм-аута произвести подключение не удастся, будет зафиксирована ошибка, которая будет обрабатываться в соответствии со значением параметра **`OnError`** (заданного в настройках источника данных или локально переопределенного в префиксе `Lookup`).

Если `Lookup` используется как значение параметра **`Router`** в секции `[Sender]`, и при этом задан режим обработки ошибок **`OnError`**`=exception` (в настройках используемого источника данных или переопределен локально), то, в случае если будет невозможно получить нужный маршрут из источника данных, эта ситуация будет обработана как ошибка в компоненте **Sender**, запись о чем будет зафиксирована в журнале. При этом в синхронном режиме компонент **Receiver** всегда возвращает отправителю письма код SMTP **451** (`Requested action aborted: local error in processing`), а обработанное письмо, согласно коду `Tempfail` (временная ошибка), удаляется из всех очередей. В асинхронном режиме письмо будет помечено как "потерянное" и **Sender** будет пытаться его отправлять с периодичностью, указанной в параметре **`StalledProcessingInterval`** этой же секции.

Неправильно записанная строка `Lookup` приведет к аварийному завершению работы **Dr.Web MailD** при запуске (на этапе чтения файла конфигурации), если он не сможет разобрать ее структуру и определить тип.

Для тестирования правильности `Lookup` рекомендуется использовать утилиту проверки Lookup.

Обратите внимание, что полный вывод в журнал информации о запросах, передаваемых через `Lookup` в источники данных (к БД) доступен только если уровень подробности ведения журнала не менее подробный, чем `DEBUG`.

Примеры использования Lookup

Пример 1:

```
ProtectedDomains = "odbc:select domain from maild where domain='\$s'"
```

С помощью этого запроса все письма, домен которых был найден в столбце `domain` таблицы `maild` в хранилище ODBC, отмечаются как принадлежащие защищаемому домену.

Пример 2:

```
ProtectedEmails = file:%etc_dir/email.ini, localhost,
ldap:skipdomains=file:/home/trusted_domains|ldaps:///??sub?(mail=$s)
```

С помощью этого запроса отмечаются как защищаемые следующие почтовые адреса:

- все адреса, находящиеся в файле `%etc_dir/email.ini`;
- адрес `localhost`;

- все адреса, которые были найдены по LDAP-запросу `ldap:///??sub?(mail=$s)` за исключением адресов, перечисленных в файле на `/home/trusted_domains` (для них запроса не происходит).

Пример 3:

```
Router = mysql:select routerinfo from maild where email='\$s', foo
inet:234@foo.ru
```

С помощью этого запроса проверяется, присутствует ли адрес в базе данных **MySQL** в таблице `maild` в столбце `email`. Если присутствует, то письмо высылается на адрес, указанный в найденной строке в столбце `routerinfo`, в противном случае для всех получателей, в адресе которых присутствует foo, высылается письмо на адрес `inet:234@foo.ru`.

Также `Lookup` могут использоваться в Правилах.

Пример 4:

```
"rcpt:ldap:///?rules?sub?(mail=$s)" cont
```

Запрос позволяет для всех LDAP-полей `mail`, в которых содержится получатель письма, получить поле `rules`, содержащее настройки, которые будут применены к данному получателю.

Обратите внимание на использование кавычек: необходимо любое условие, указанное в секции `CONDITION` в Правилах, заключать в кавычки, так как в нем могут содержаться специальные символы (например, круглые скобки "`()`"). Таким образом, если будет написано:

```
rcpt:"ldap:///?rules?sub?(mail=$s)" cont
```

компилятор выдаст ошибку:

```
Mon Jun 29 18:53:01 2009 [3081262768] maild.rules ERROR '(' can not follow
'"ldap:///?rules?sub?'
Mon Jun 29 18:53:01 2009 [3081262768] maild.rules ERROR error in parse
condition:
'rcpt:"ldap:///?rules?sub?(mail=$s)" cont'
```

Пример 5:

```
"any:sqlite:select skipaddr from domain where skipaddr = '$s'" cont
scan=all:-drweb
```

По данному запросу проверяются адреса, и если адрес отправителя или получателя содержится в поле `skipaddr` таблицы `domain` базы данных **SQLite**, то для них не будет использоваться подключаемый модуль **Drweb**.

Ограничения использования Lookup и тип LookupLite

Существуют некоторые ограничения по использованию определенных типов `Lookup`. В некоторых случаях полный набор префиксов не может быть использован (не все префиксы разрешены). В этом случае используется специальный тип данных `LookupLite`.

`LookupLite` – это тип значений, аналогичный `Lookup`, в котором можно указывать только следующие типы (префиксы):

- непосредственное значение (`value:`)
- `file:`

Тип `LookupLite` используется:

- в настройках самих `Lookup` (например, в настройке **`SkipDomains`** для каждого `Lookup`);
- во всех настройках подключаемых модулей.

При попытке задать `Lookup` у параметров, требующих тип `LookupLite`, в журнал выводится ошибка вида:

```
Wed Jun 10 14:02:20 2009 [4160149200] Modifier ERROR Error in init lookup
[cdb:select * from /root/mail/base_file_for_CDB.txt where key='domain']:
can't use this lookup here.
```

Для тестирования правильности `Lookup` и `LookupLite` рекомендуется использовать утилиту проверки Lookup.

Тип данных Storage (хранилище)

Тип данных `хранилище (Storage)` описывает объекты для хранения данных. Синтаксис аналогичен Lookup за исключением следующих отличий:

- этот тип имеет другой список префиксов;
- здесь нельзя использовать специальный символ `$s`.

Существуют следующие варианты префиксов:

- `value` — за ним указывается непосредственно искомое значение. Этот префикс используется, если, к примеру, в значении встречается символ ":".
- `odbc`, `oracle` — синтаксис аналогичен тому же в `Lookup`. В SQL-выражении можно задавать сохраняемые значения в формате:

  ```
  :name<type>
  ```

  где `name` — имя сохраняемого объекта (для каждого параметра имеется свой собственный список возможных имен), а `type` – тип параметра, под которым надо сохранять параметр в хранилище.
- `postgres`, `mysql`, `sqlite`, `firebird` — синтаксис аналогичен предыдущему за исключением того, что SQL-тип полей `char(length)` не поддерживается, и для строковых данных следует использовать SQL-тип `varchar_long`.

Префиксы `odbc`, `oracle`, `postgres`, `mysql`, `sqlite`, `firebird` используются для обращения к соответствующим базам данных с использованием настроек из одноименных секций конфигурационного файла **Dr.Web MailD**.

Пример:

```
"odbc:insert into plugin_stat values \
(:plugin_name<varchar_long>, :size<int>, \
:num<int>)";
```

Обратите внимание на использование кавычек: они необходимы, так как в запросе содержатся запятые.

Секции основного конфигурационного файла

Конфигурационный файл комплексного компонента **Dr.Web MailD**, как и любой конфигурационный файл любого компонента программного комплекса **Dr.Web для почтовых серверов UNIX**, является текстовым и состоит из секций (см. общее описание конфигурационных файлов **Dr.Web для почтовых серверов UNIX**).

В конфигурационном файле комплексного компонента **Dr.Web MailD** могут присутствовать следующие секции:

Секции, задающие основные параметры **Dr.Web MailD**:

- [General] – Содержит общие настройки работы **Dr.Web MailD**, должна присутствовать обязательно;
- [Maild] – Содержит общие параметры работы компонента **MailD core**, должна присутствовать обязательно;
- [MailBase] – Содержит настройки работы с хранилищем почтовых сообщений **MailBase**, должна присутствовать обязательно;
- [Notifier] – Содержит настройки работы компонента **Notifier**, используемого для отправки уведомлений, отчетов и DSN. Должна присутствовать обязательно;
- [Quarantine] – Содержит настройки работы **Карантина**, должна присутствовать обязательно;
- [Filters] – Содержит настройки всех используемых подключаемых модулей, включая порядок их вызова при проверке писем. Должна присутствовать обязательно;
- [Rule] – Содержит группу настроек по умолчанию для параметров, используемых в Правил обработки писем (требуется для инициализации значений этих параметров в случае если они не определяются никаким из Правил). Должна присутствовать обязательно и находиться выше секции `[Rules]`.
- [Rules] – Содержит перечень Правил обработки писем. Может отсутствовать в случае если такая обработка не требуется или если все правила помещены во встроенную базу данных;
- [Stat] – Содержит настройки сбора и формирования статистики. Может отсутствовать;
- [Reports] – Содержит настройки формирования отчетов. Может отсутствовать;
- [Logging] – Содержит настройки ведения журналов. Может отсутствовать.

Секции, задающие параметры аутентификации SASL, если она используется. Если SASL не используется, секции могут отсутствовать. Если SASL используется, должны присутствовать обе секции:

- [SASL] – Содержит параметры аутентификации SASL;
- [Cyrus-SASL] – Содержит параметры, управляющие работой SASL-драйвера `cyrus-sasl`.

Секции, определяющие параметры взаимодействия с почтовыми системами (MTA):

- [Receiver] – Содержит параметры работы компонента **Receiver**, работающего по протоколу SMTP/LMTP, а также с почтовыми системами **Exim**, **Zmailer** и **Postfix** (если **Postfix** не использует протокол **`Milter`**). Используется, если **Dr.Web для почтовых серверов UNIX** работает в режиме **SMTP/LMTP-прокси** или при интеграции с почтовыми системами **Exim**, **Zmailer** и **Postfix** (если **Postfix** не использует протокол **`Milter`**). Может отсутствовать, если не используется режим **SMTP/LMTP-прокси** или не производится взаимодействие с почтовыми системами **Exim**, **Zmailer**, **Postfix** (или **Postfix** использует протокол **`Milter`**).
- [Sender] – Содержит параметры работы компонента **Sender**, работающего как по протоколу SMTP/LMTP, так и со всеми почтовыми системами, кроме **Communigate Pro**. Используется, если **Dr.Web для почтовых серверов UNIX** работает в режиме **SMTP/LMTP-прокси** или при интеграции с любой почтовой системой, кроме **Communigate Pro**. Может отсутствовать, если производится только взаимодействие с почтовой системой **Communigate Pro**.

  Обратите внимание, что в случае работы **Dr.Web для почтовых серверов UNIX** в режиме **SMTP/LMTP-прокси**, а также в случае работы с почтовыми системами **Exim** и **Zmailer** и **Postfix** (если **Postfix** не использует протокол **`Milter`**) в конфигурационном файле должны присутствовать обе секции (`[Receiver]` и `[Sender]`). Может присутствовать только секция `[Sender]`, если производится взаимодействие с почтовыми системами **Qmail**, **Courier** или любой почтовой системой, использующей протокол **`Milter`** (например, **Sendmail** или **Postfix**).

- [Courier] – Содержит параметры взаимодействия с почтовой системой **Courier**. Используется, если **Dr.Web для почтовых серверов UNIX** работает в режиме интеграции с почтовой системой **Courier**. Может отсутствовать, если интеграция с почтовой системой **Courier** не используется.
- [CgpReceiver] – Содержит параметры работы компонента **Receiver**, работающего с почтовой системой **Communigate Pro**. Используется, если **Dr.Web для почтовых серверов UNIX** работает в режиме интеграции с почтовой системой **Communigate Pro**. Может отсутствовать, если интеграция с почтовой системой **Communigate Pro** не используется.
- [CgpSender] – Содержит параметры работы компонента **Sender**, работающего с почтовой системой **Communigate Pro**. Используется, если **Dr.Web для почтовых серверов UNIX** работает в режиме интеграции с почтовой системой **Communigate Pro**. Может отсутствовать, если интеграция с почтовой системой **Communigate Pro** не используется.

  Обратите внимание, что в случае работы **Dr.Web для почтовых серверов UNIX** с почтовой системой **CommuniGate Pro** в конфигурационном файле должны присутствовать одновременно обе секции (`[CgpReceiver]` и `[CgpSender]`).

- [Milter] – Содержит параметры взаимодействия с почтовыми системами, использующими протокол интеграции **Milter**. Используется, если **Dr.Web для почтовых серверов UNIX** работает в режиме интеграции с почтовыми системами, использующими протокол интеграции **Milter** (например, **Sendmail** или **Postfix**). Может отсутствовать, если интеграция с почтовыми системами, использующими протокол интеграции **Milter**, не используется.
- [Qmail] – Содержит параметры взаимодействия с почтовой системой **Qmail**. Используется, если **Dr.Web для почтовых серверов UNIX** работает в режиме интеграции с почтовой системой **Qmail**. Может отсутствовать, если интеграция с почтовой системой **Qmail** не используется.
- [IMAP] – Содержит параметры работы **Dr.Web для почтовых серверов UNIX** в качестве посредника между MUA, использующими протокол **IMAP**, и почтовой системой MDA. Может отсутствовать, если работа в режиме посредника между MUA и почтовой системой MDA по протоколу **IMAP** не используется.
- [POP3] – Содержит параметры работы **Dr.Web для почтовых серверов UNIX** в качестве посредника между MUA, использующими протокол **POP3** и почтовой системой MDA. Может отсутствовать, если работа в режиме посредника между MUA и почтовой системой MDA по протоколу **POP3** не используется.

В файле обязательно должны присутствовать только те секции, которые определяют параметры взаимодействия с теми MTA, с которыми интегрирован **Dr.Web для почтовых серверов UNIX**. Секции задающие параметры взаимодействия с MTA, которые не используются, даже если они присутствуют в файле, не влияют на работу **Dr.Web для почтовых серверов UNIX**. Подробнее об использовании секций конфигурационного файла при интеграции с различными почтовыми системами см. в разделе Подключения почтовых систем.

Секции, определяющие параметры подключения к источникам данных (LDAP, реляционные БД, текстовые файлы):

- [LDAP] – Содержит параметры подключения к хранилищам LDAP;
- [Oracle] – Содержит параметры подключения к СУБД **Oracle**;
- [ODBC] – Содержит параметры подключения к любому источнику данных, для которого имеется настроенный драйвер ODBC, включая любую реляционную базу данных.
- [SQLite] – Содержит параметры подключения к СУБД **SQLite**;
- [Firebird] – Содержит параметры подключения к СУБД **Firebird**;
- [PostgreSQL] – Содержит параметры подключения к СУБД **PostgreSQL**;
- [MySQL] – Содержит параметры подключения к СУБД **MySQL**;

- [CDB] – Содержит параметры подключения к текстовой базе данных **CDB**;
- [Berkeley] – Содержит параметры подключения к текстовой базе данных **Berkeley**;

Обратите внимание, что данные секции содержат только общие параметры, используемые в Lookup и Storage по умолчанию. Некоторые параметры (отмеченные в описании особо) могут переопределяться в каждом конкретном `Lookup` и `Storage`.

Рекомендуется настраивать подключения к базам данных через «нативное» подключение, если это возможно, а не через **ODBC**, поскольку драйвера **ODBC** работают медленнее в силу своей универсальности.

В файле могут присутствовать любые секции подключения к источникам данных. Реальное использование источников данных определяется только ссылками на них в `Lookup` и `Storage`. Секции, задающие параметры подключения к источникам данных, которые не используются, даже если они присутствуют в файле, не влияют на работу **Dr.Web для почтовых серверов UNIX**.

Секции, определяющие параметры проксирования:

- [ProxyClient] – Содержит параметры прокси-клиента, через которого компоненты **Sender** и **Receiver** подключаются к основному компоненту **MailD core**.
- [ProxyServer] – Содержит параметры прокси-сервера, обслуживающего подключения прокси-клиентов к основному компоненту **MailD core**.

Обратите внимание, что:

- Отсутствие какой-либо секции в конфигурационном файле означает, что параметры, содержащиеся в этой секции, имеют при работе соответствующего компонента значения по умолчанию. Значения по умолчанию указаны в данном документе при описании каждого параметра.
- **Крайне не рекомендуется** добавлять в конфигурационный файл, который был автоматически сгенерирован при установке продукта, секции и параметры, которые описаны в документации, но отсутствуют в файле. Это связано с тем, что эти параметры специфичны для конкретной используемой почтовой системы, и изменение их значения в случае интеграции с другой почтовой системой может нарушить работу программного комплекса **Dr.Web для почтовых серверов UNIX.**

Основные параметры работы

В данном разделе описаны секции конфигурационного файла, содержашие основные параметры работы сервиса **Dr.Web для почтовых серверов UNIX** и его основного компонента – комплексного компонента **Dr.Web MailD**.

Как правило, все секции конфигурационного файла, перечисленные в данном разделе, всегда присутствуют в файле.

Секция [General]

В секции `[General]` собраны общие настройки работы **Dr.Web MailD**:

BaseDir = {путь к каталогу}	Основной рабочий каталог, в котором содержатся сокеты, база данных и другие файлы. В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала `HUP`. Необходимо перезапустить **Dr.Web MailD**.
	Значение по умолчанию: **BaseDir** = %var_dir

MaxTimeoutForThreadActivity = {время}	Максимальное время закрытия одного потока. Параметр используется для ожидания при перезапуске и завершении работы **Dr.Web MailD**. Общее максимальное время завершения работы программы рассчитывается следующим образом: оно будет не меньше, чем количество пулов потоков, умноженное на значение параметра **MaxTimeoutForThreadActivity**. Значение по умолчанию: **MaxTimeoutForThreadActivity** = 30s
IpcTimeout = {время}	Максимальное время установки соединения между компонентами или ожидания ответа на запрос. По достижению этого времени соединение разрывается и фиксируется ошибка взаимодействия компонентов, которая будет обрабатываться в зависимости от значения параметра **ProcessingErrors** в секции [Maild] В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала HUP. Необходимо перезапустить **Dr.Web MailD**. Величина тайм-аута должна быть соизмерима со средним временем обработки письма подключаемыми модулями, особенно, если используется синхронный режим **before-queue** (когда подключаемые модули вызываются из списка, заданного в параметре **BeforeQueueFilters** в секции [Filters]) Значение по умолчанию: **IpcTimeout** = 40s
Hostname = {текст}	Имя хоста, на котором работает **Dr.Web для почтовых серверов UNIX**. Если этот параметр не задан, используется значение, возвращенное функцией **gethostname(3)**. Значение по умолчанию: **Hostname** =

Секция [Maild]

В секции [Maild] собраны параметры работы центрального компонента **Dr.Web MailD** – **MailD core.**

ProtectedNetworks = {Lookup}	Список сетей, защищаемых **Dr.Web MailD**. Значения записываются с использованием бесклассовой адресации (CIDR). Этот параметр используется для определения доверенных сетей в соответствующих настройках подключаемого модуля **Vaderetro**, и если в параметре **SessionRestrictions** в секции [Receiver] указано значение trust_protected_networks. Обратите внимание, что значение параметра – Lookup. На использование Lookup в данном параметре наложен ряд ограничений, указанных ниже. **Пример**: **ProtectedNetworks** = 10.0.0.0/24, 127.0.0.0/8, "mysql:select net from networks where net='$s'"

	Значение по умолчанию: **ProtectedNetworks** = 127.0.0.0/8
ProtectedDomains = {Lookup}	Задает список доменов, защищаемых **Dr.Web MailD**. Этот параметр используется для определения доверенных доменов, если в параметре **SessionRestrictions** в секции [Receiver] указано значение trust_protected_domains. Обратите внимание, что значение параметра – Lookup. **Пример**: **ProtectedDomains** = example.ru, example.com Значение по умолчанию: **ProtectedDomains** =
IncludeSubdomains = {логический}	Включение поддоменов в список защищаемых доменов. Значение по умолчанию: **IncludeSubdomains** = yes
InPoolOptions = {настройки пула}	Настройки пула потоков, обрабатывающих письмо до помещения в очередь. Значение по умолчанию: **InPoolOptions** = auto
OutPoolOptions = {настройки пула}	Настройки пула потоков, обрабатывающих письмо после помещения в очередь. Значение по умолчанию: **OutPoolOptions** = auto
RedirectMail = {адрес электронной почты}	R Почтовый адрес, на который отсылаются сообщения при использовании действия Redirect, если в параметрах самого действия адрес перенаправления не указан. Значение по умолчанию: **RedirectMail** = root@localhost
OnlyTrustedControlMails = {логический}	Возможность отправлять управляющие письма только из защищаемой сети. Если компонент **Receiver** не передал информацию об IP-адресе клиента, то необходимо с помощью параметра **GetIpFromReceivedHeader** (см. ниже) заставить используемый MTA добавлять правильный заголовок Received ко всем письмам, передаваемым на обработку программному комплексу **Dr.Web для почтовых серверов UNIX**. Для успешной работы управляющих писем весь исходящий почтовый трафик клиентов должен проверяться с помощью **Dr.Web для почтовых серверов UNIX** Значение по умолчанию: **OnlyTrustedControlMails** = Yes
MaxScore = {числовое **значение**}	R Максимальный счет сообщения. Если счет сообщения превысит значение, указанное в данном параметре, то для него выполняются действия, указанные в параметре **MaxScoreAction**, и проверка сообщения прерывается.

	Данный параметр проверяется перед передачей сообщения на проверку подключаемым модулям, а также после проверки каждым из подключаемых модулей. Значение по умолчанию: **MaxScore** = 10000
MaxScoreAction = {список **действий**}	R Действия, выполняемые, если счет письма превысит значение параметра **MaxScore**. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: pass, discard, reject, tempfail. Дополнительно могут быть заданы следующие действия: quarantine, redirect, add-header, score. Если указано основное действие reject и значение параметра **UseCustomReply** установлено в yes, то SMTP-ответ берется из параметра **ReplyMaxScore** (см. ниже). После выполнения всех действий проверка сообщения завершается. Значение по умолчанию: **MaxScoreAction** = reject
MaxMimeParts = {числовое значение}	Максимальное число MIME-частей в письме. Если значение параметра равно 0, то проверка не производится. В случае, если число MIME-частей в письме превысит указанное в параметре значение, разбор и проверка сообщения прерываются и для него выполняются действия, указанные в параметре **ProcessingError** (см. ниже). Значение по умолчанию: **MaxMimeParts** = 1000
MaxNestedMimeParts = {числовое значение}	Максимальное число вложенных в письмо MIME-частей. Если значение параметра равно 0, то проверка не производится. В случае, если число вложенных MIME-частей в письме превышает указанное в параметре значение, то разбор и проверка сообщения прерываются, и для него выполняются действия, указанные в значении параметра **ProcessingError** (см. ниже). Значение по умолчанию: **MaxNestedMimeParts** = 100
LicenseLimit = {список действий}	R Действия над сообщениями, которые не были проверены из-за лицензионных ограничений. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: pass, discard, reject, tempfail. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **LicenseLimit** = pass

EmptyFrom = {список действий}	R Реакция на пустое поле From в заголовках письма. Подобная ситуация возможна при использовании DSN (которые в соответствии с требованиями протокола должны иметь пустое поле From); также это поле обычно не заполняют спамеры. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: continue, discard и reject. Дополнительно могут быть заданы следующие действия: quarantine, redirect, add-header, score. Значение по умолчанию: **EmptyFrom** = continue
ProcessingErrors = {список действий}	R Действие, применяемое к сообщениям, вызвавшим ошибки сканирования. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: pass, discard, reject, tempfail. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Обратите внимание на особенности обработки ошибок, приведенные ниже. Значение по умолчанию: **ProcessingErrors** = pass
PidFile = {путь к файлу}	Путь к PID-файлу процесса **drweb-maild** (исполняемого модуля центрального компонента **MailD core**) Значение по умолчанию: **PidFile** = %var_dir/run/drweb-maild.pid
RulesLogLevel = {уровень подробности}	Определяет уровень подробности журнала работы обработчика Правил. Допустимо использование следующих уровней: • Quiet • Error • Alert • Info • Debug Значение по умолчанию: **RulesLogLevel** = Alert

В тех случаях, когда сообщение блокируется программным комплексом (т.е. для сообщения срабатывает действие **reject**), его SMTP-ответ состоит из кода ошибки 550 5.7.0 и текстового сообщения, содержание которого может задаваться параметрами, описанными ниже.

UseCustomReply = {логический}	R Использование настраиваемых сообщений в SMTP-сессии. Данные сообщения будут отправляться в качестве SMTP-ответа в случае, если входящее сообщение отклонено. Значение по умолчанию: **UseCustomReply** = No

ReplyEmptyFrom = {текст}	R Ответ, отправляемый клиенту при срабатывании действия **EmptyFrom**, если: • EmptyFrom = reject; • **UseCustomReply** = Yes. Возможно задать только текстовую часть ответа: "550 5.7.0 <Текст>". В случае если текст содержит пробелы, он должен быть заключен в кавычки. Значение по умолчанию: **ReplyEmptyFrom** = "DrWEB maild: Messages from <> are blocked by administrator."
ReplyProcessingError = {текст}	R Ответ, отправляемый клиенту при срабатывании действия **ProcessingError**, если: • **ProcessingError** = reject; • **UseCustomReply** = Yes. Возможно задать только текстовую часть ответа: "550 5.7.0 <Текст>". В случае если текст содержит пробелы, он должен быть заключен в кавычки. Значение по умолчанию: **ReplyProcessingError** = "Dr.Web MailD: Message is rejected due to software error."
ReplyMaxScore = {текст}	R Ответ, отправляемый клиенту при срабатывании действия **MaxScoreAction**, если: • **MaxScoreAction** = reject; • **UseCustomReply** = yes. Возможно задать только текстовую часть ответа: "550 5.7.0 <Текст>". В случае если текст содержит пробелы, он должен быть заключен в кавычки. Значение по умолчанию: **ReplyMaxScore** = "Dr.Web MailD: Message is rejected due to score limit exceed."
GetIpFromReceivedHeader = {логический}	Предписывает **MailD core** использовать в качестве IP-адреса клиента значение, извлеченное компонентом **Receiver** из заголовка Received. Обратите внимание, что компонент **Receiver** не всегда может определить IP-адрес клиента на основе анализа заголовка Received. Значение по умолчанию: **GetIpFromReceivedHeader** = Yes
Control = {логический}	Включение интерактивного управления для модуля **drweb-maild** (компонент **MailD core**). Значение по умолчанию: **Control** = No
ControlAddress = {адрес}	Адреса, используемые подсистемой интерактивного управления модуля **drweb-maild** (компонент **MailD core**).

	Значение по умолчанию: **ControlAddress** = inet:3009@127.0.0.1
ControlPoolOption = {настройки пула}	Настройки пула потоков для сокета интерактивного управления **drweb-maild**. Значение по умолчанию: **ControlPoolOption** = auto
SkipDSNOnBlock = {логический}	Отказ от отправления DSN, если при выполнении действий Reject или Tempfail код возврата невозможно вернуть компоненту **Receiver**. Значение по умолчанию: **SkipDSNOnBlock** = No

Особенности использования Lookup в параметре ProtectedNetworks

Нельзя использовать в качестве значений этого параметра Lookup, использующие извлечение из источника данных IP-адреса сети по имени домена или имени пользователя (т.е. использующие макросы $d, $u), поскольку на том этапе сессии, когда производится обращение к этому параметру для проверки ограничения **SessionRestrictions** = trust_protected_network, имеется только информация об IP-адресе, с которого произведено подключение клиента и нет возможности разрешения его в FQDN.

Например, попытка использования Lookup с SQL-запросом

```
select net from networks where domain='$d'
```

приведет к тому, что адрес **net** не будет выбран из БД, и поэтому не будет помечен как доверенный.

Однако вы можете использовать Lookup, использующие извлечение из источника данных IP-адреса сети по полному адресу (т.е. использующие макрос $s), но помните, что на данном этапе он будет указывать на IP-адрес клиента, поэтому его можно использовать только в запросах к источникам данных, содержащим перечни IP-адресов, или имеющим возможность выполнить самостоятельное разрешение FQDN по известному IP-адресу. Например, такой запрос будет корректным:

```
select net from networks where net='$s'
```

И если в поле **net** присутствует IP-адрес клиента, подставленный в запрос через макрос $s, то он будет помечен как доверенный.

Также обратите внимание, что в Lookup для этого параметра не имеет смысла использование настройки **SkipDomains** (она никогда не сработает, поскольку имя домена на этапе сессии, когда используется этот параметр, не известно).

Подробнее об ограничениях на использование имени домена и пользователя см. в описании типа Lookup.

Особенности обработки ошибок

Пожалуйста, обратите внимание, что если в процессе обработки сообщений происходит событие, подпадающее под любое из указанных ограничений (**MaxScore**, **MaxMimeParts**, **LicenseLimit**) или произойдет ошибка обработки, то выполняется действие, указанное в соответствующем параметре:

- **EmptyFrom**
- **MaxScoreAction**
- **LicenseLimit**

- **ProcessingErrors**

Крайне внимательно относитесь к действию, заданному в этих параметрах. Помните, что:

1. Если задано действие `discard`, `reject` или `tempfail`, то обработка сообщения останавливается, оно удаляется и не доставляется получателю. В случае `discard` отправитель письма также не получает никакого уведомления о том, что письмо было отвергнуто. Действия `reject` и `tempfail` отличаются от `discard` тем, что в этом случае производится уведомление отправителя о том, что его сообщение было отклонено. В зависимости от режима работы, уведомление отправителя осуществляется либо SMTP-ответом (в синхронном режиме, через **Receiver**), либо отправкой ему соответствующего DSN (в асинхронном режиме, через **Sender**).
2. Если выбрано действие `pass`, то обработка письма прерывается и оно сразу же поступает на доставку, минуя всю незаконченную обработку (например, подключаемые модули, которые тоже должны были проверить это письмо, но еще не успели это сделать, вызваны не будут). При этом если событие возникло до сохранения письма на диск (т.е. в момент обработки письма подключаемыми модулями из очереди **BeforeQueue**), то письмо будет посылаться синхронно, а если после сохранения (при обработке подключаемыми модулями из очереди **AfterQueue**) – асинхронно.
3. В действии **EmptyFrom** действие `pass` не может быть указано. Вместо него можно использовать действие `continue`, которое приводит к тому, что письмо поступает на проверку подключаемыми модулями (т.к. событие **EmptyFrom** может наступить только до обработки письма подключаемыми модулями).

Пожалуйста, обратите внимание, что если в качестве основного действия в параметре **ProcessingErrors** задано действие `discard`, `reject` или `tempfail`, то не следует задавать значение параметра **IpcTimeout** (секция `[General]`) малым, поскольку проверка содержимого писем подключаемыми модулями может продолжаться долго. В этом случае истечение заданного тайм-аута до окончания проверки расценивается как ошибка, и поэтому, в соответствии с действием, заданным в **ProcessingErrors**, письмо будет удалено прямо в процессе проверки, что может привести к его потере и недоставке получателю без отправки соответствующего уведомления отправителю.

Секция [MailBase]

В секции `[MailBase]` собраны настройки встроенной базы данных **Dr.Web MailD**, используемой для хранения принятых писем до окончания момента их обработки подключаемыми модулями и отправки, если обработка происходит в асинхронном режиме **after-queue.** Секция содержит следующие параметры:

MaxStoredMessages = {числовое значение}	Максимальное количество сообщений в хранилище. При значении `0` ограничения отсутствуют. Если количество писем в хранилище превышает указанное значение, производится очистка хранилища от самых старых писем до достижения нужного количества писем. Уже отправленные письма сразу удаляются, еще не отправленные – отправляются и удаляются.
	Значение по умолчанию: **MaxStoredMessages** = 100000
MaxStorageSize = {числовое значение}	Максимальный размер хранилища сообщений в байтах. При значении `0` ограничения отсутствуют. Если размер хранилища превышает предельный, производится его очистка от самых старых писем до достижения нужного размера (см. описание параметра **MaxStoredMessages**)

	Значение по умолчанию: **MaxStorageSize** = 0
MaxPoolSize = {числовое значение}	Максимальное количество страниц памяти размером по 8 Кб, выделяемых для пула хранилища сообщений. При значении 0 количество устанавливается автоматически, исходя из доступного объема физической памяти. В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала HUP. Необходимо перезапустить **Dr.Web MailD**. Значение по умолчанию: **MaxPoolSize** = 0
SendTimeout = {время}	Максимальное время, отведенное на сканирование сообщения подключаемыми модулями перед отправкой его дальше. В случае если максимальное время сканирования превышено, считается, что при проверке сообщения произошла ошибка. Действия для такого случая определяются в параметре **ProcessingError** секции [Maild]. Значение по умолчанию: **SendTimeout** = 30s
FrozenTimeout = {время}	Дополнительное время на обработку письма. Если подключаемый модуль не может обработать письмо за время, указанное в значении параметра **SendTimeout**, он может продлить время обработки на величину, заданную в параметре **FrozenTimeout**. Обратите внимание, что это параметр устарел и более не используется (изменение его значения не влияет на работу **Dr.Web MailD**)! Значение по умолчанию: **FrozenTimeout** = 2h
DeleteTimeout = {время}	Максимальное время хранения письма в хранилище. Значение по умолчанию: **DeleteTimeout** = 48h
BackupPeriod = {время}	Промежуток времени, через который производится резервное копирование хранилища писем. При значении 0 резервное копирование не производится. Значение по умолчанию: **BackupPeriod** = 0
BackupName = {имя файла}	Имя файла резервной копии хранилища сообщений. Если указанное имя файла оканчивается знаком вопроса ("?"), то каждая резервная копия сохраняется в отдельный файл, а знак вопроса в имени файла заменяется значением времени, когда резервная копия была создана. Значение по умолчанию: **BackupName** = %var_dir/msgs/db/.maildb.backup
MaxBodySizeInDB = {размер}	Максимальный размер тела сообщения, сохраняемого в хранилище писем.

	При превышении указанного значения письмо сохраняется в отдельном внешнем файле. Значение по умолчанию: **MaxBodySizeInDB** = 1k
SyncMode = {логический}	Режим синхронизации, используемый для внутренней БД. Если для данного параметра указано значение Yes, то для каждой транзакции вызывается функция fsync. В результате БД на диске гарантированно находится в актуальном состоянии после каждой транзакции. Однако при этом производительность уменьшается (причем иногда – значительно). Если указано значение No, то при обновлении БД используется буферизация ОС. В результате в случае аварийного завершения работы модуля **drweb-maild** могут быть потеряны данные последних транзакций, но при этом БД не будет разрушена и производительность комплекса увеличится. Если нет повышенных требований к надежности системы, то рекомендуется оставить данный параметр в значении No. Значение по умолчанию: **SyncMode** = no

Секция [Notifier]

В секции [Notifier] содержатся настройки модуля **drweb-notifier** (компонента **Notifier**) отвечающего за создание и отправку пользователям (отправителям и получателям писем и администратору комплекса, в зависимости от настроек и Правил) уведомлений MailD о действиях компонентов программного комплекса **Dr.Web для почтовых серверов UNIX**, отчетов со статистикой, а также DSN.

PoolOptions = {настройки пула}	Настройки пула потоков обработки уведомлений. Значение по умолчанию: **PoolOptions** = auto
TemplatesBaseDir = {путь к каталогу}	Путь к каталогу, в котором хранятся шаблоны уведомлений и DSN. Значение по умолчанию: **TemplatesBaseDir** = %etc_dir/maild/templates
LngBaseDir = {путь к каталогу}	Путь к каталогу, в котором хранятся языковые файлы, используемые для формирования уведомлений. О структуре и назначении языковых файлов см. в разделе Языковые файлы. Значение по умолчанию: **LngBaseDir** = %etc_dir/maild/lng
AdminMail = {адрес электронной почты} R C	Адрес системного администратора. Можно указать несколько адресов, тогда все сформированные уведомления (за исключением DSN, которые всегда отправляются только отправителю письма, при доставке которого возникли проблемы) будут отправляться на все заданные адреса, и при этом в теле письма с уведомлением будут отображаться все указанные адреса. Рекомендуется определить этот параметр, иначе не будут

	отправляться уведомления MailD (включая отчеты). Обратите внимание, что при настройке данного параметра может быть одновременно задано несколько значений (список). Значение по умолчанию: **AdminMail** = root@localhost
FilterMail = {адрес электронной почты}	R Адрес, указываемый в заголовке From писем с уведомлениями MailD. Обратите внимание, что в соответствии со спецификацией почтовых протоколов, DSN отправляются с пустым адресом отправителя (для них этот параметр не используется). Значение по умолчанию: **FilterMail** = root@localhost
NotifyLangs = {список названий языков}	R C Список языков, используемых при формировании уведомлений. Перечисленные языки должны соответствовать языковым файлам, доступным в каталоге, указанном в **LngBaseDir** О структуре и назначении языковых файлов см.в разделе Языковые файлы. Подключаемые модули всегда используют первый язык из данного списка. Значение по умолчанию: **NotifyLangs** = en
TemplatesParserLogLevel = {уровень подробности}	Уровень подробности журнала работы подсистемы, формирующей уведомления на основе шаблонов. Допустимо использование следующих уровней: • quiet • error • alert • info • debug Значение по умолчанию: **TemplatesParserLogLevel** = info
RulesLogLevel = {уровень подробности}	Уровень подробности журнала работы обработчика Правил. Допустимо использование следующих уровней: • quiet • error • alert • info • debug Значение по умолчанию: **RulesLogLevel** = Alert
MsgIdMap = {строка}	Карта идентификаторов, использующаяся для отображения идентификатора сообщения, заданного в компоненте **Receiver**, в идентификатор компонента **Sender**, которому будут отправляться уведомления, сформированные для данного сообщения.

	Имеет вид: `<регулярное_выражение> <идентификатор_Sender>`, где `<регулярное_выражение>` – регулярное выражение, которому должен соответствовать идентификатор компонента **Receiver**, а `<идентификатор_Sender>` – идентификатор компонента **Sender**, которому будут отправляться уведомления. Если такое отображение не найдено, все уведомления будут отправляться компоненту **Sender** по умолчанию (с пустым идентификатором). **Пример**: **MsgIdMap** = id[12] sender_notifications В этом случае уведомления для сообщений, сформированных компонентами **Receiver** с идентификаторами id1 или id2, будут отправляться компоненту **Sender** с идентификатором sender_notifications. Данный параметр применяется в случае единовременной работы нескольких пар компонентов **Receiver** и **Sender**.
	Значение по умолчанию: **MsgIdMap** =
QuarantinePrefix = {строка}	Префикс, добавляемый при выводе пути к файлу в **Карантине**. Данный параметр позволяет получить доступ к файлам в **Карантине** через сторонний сервер. Например, установив HTTP-сервер на том же хосте, где работает **Dr.Web для почтовых серверов UNIX**, и настроив его, можно задать **QuarantinePrefix** = http://mailhost/quarantine/ – тогда в уведомлении для пути файла в **Карантине** будет выводиться, например, http://mailhost/quarantine/headersfilter/drweb.quarantine.2kqtvI
	Значение по умолчанию: **QuarantinePrefix** =

Об уведомлениях, шаблонах уведомлений и правилах их формирования см. раздел Шаблоны уведомлений.

Секция [Quarantine]

В секции [Quarantine] собраны настройки работы **Карантина**.

Path = {путь к каталогу}	Путь к каталогу **Карантина**.
	Значение по умолчанию: **Path** = %var_dir/infected/
FilesMode = {права}	Права на файлы, помещаемые в каталог **Карантина**.
	Значение по умолчанию: **FilesMode** = 0660
FilenamesMode = {std | tai | rand48}	Способ именования файлов, перемещаемых в **Карантин**: • std – переименование с использованием команды mkstemp. Используется шаблон имени %FilenamesPrefix.XXXXXX, где %FilenamesPrefix -

	префикс, задаваемый значением параметра **FilenamesPrefix**, а XXXXXX - комбинация случайных букв и цифр; • tai – переименование согласно TAI (международное атомное время). Используется шаблон имени %sec.%usec.%FilenamesPrefix.XXXXXX; • rand48 – переименование с использованием команды lrand48. Используется шаблон имени %FilenamesPrefix.XXXXXXXX. Значение по умолчанию: **FilenamesMode** = Std
FilenamesPrefix = {текст}	Префикс, применяемый при переименовании файлов, помещаемых в **Карантин**. Из значения параметра будут удалены символы "%", "/" и "_". Значение по умолчанию: **FilenamesPrefix** = maild
AccessByEmail = {логический}	Разрешение на обработку запросов на получение писем, сохраненных в **Карантине**, через отправление специальных управляющих писем. Управляющее письмо должно быть направлено отправителем на адрес, указанный в значении параметра **FilterMail** (или в Правилах) со специальным заголовком вида q:relative_path_to_file где relative_path_to_file – относительный путь к файлу в **Карантине** (например, /drweb/drweb.quarantine.puYtWx). Сохраненное письмо будет отправлено в ответ на такой запрос, только если один из его получателей или его отправитель совпадают с отправителем управляющего письма. Такое управляющее письмо может быть автоматически сгенерировано MUA получателя уведомления MailD при нажатии им ссылки, вставленной в уведомление MailD. Обратите внимание, что поскольку значение параметра **OnlyTrustedControlMails** секции [Maild] по умочанию равно **yes**, управляющие письма должны отправляться из защищенной сети (сети, указанной в параметре **ProtectedNetworks** секции [Maild]). В противном случае запрос будет проигнорирован. Значение по умолчанию: **AccessByEmail** = Yes
StoredTime = {время}	Время хранения письма в **Карантине**. При значении 0 ограничений на время хранения нет. Значение по умолчанию: **StoredTime** = 24h
MaxSize = {числовое значение}	Общий максимальный размер (в килобайтах) сообщений в **Карантине**. Если значение равно нулю, то размер не ограничен. Для каждого сообщения учитывается размер тела сообщения, а не фактический размер занимаемого на диске места. Данный параметр влияет только на размер внутренней БД, и никак не

	влияет на хранилище **DBI**, если оно подключено. Значение по умолчанию: **MaxSize** = 0
MaxNumber = {числовое значение}	Максимальное число сообщений в **Карантине**. Если значение равно 0, то число сообщений в **Карантине** не ограничено. Данный параметр влияет только на число сообщений во внутренней БД, и никак не влияет на хранилище **DBI**, если оно подключено. Значение по умолчанию: **MaxNumber** = 0
MoveToDBI = {логический}	Перемещение писем, сохраненных в **Карантине**, из файлового хранилища в хранилище **DBI**. Для перемещения сообщений в хранилище **DBI** должны быть установлены модули **Perl** File::Temp и DBI. Значение по умолчанию: **MoveToDBI** = No
DBISettings = {текст}	Настройки подключения к хранилищу **DBI**. **Пример**: "dbi:Pg:dbname=emails_db" База данных должна быть создана с использованием набора символов SQL-ASCII. Также ниже см. требования к формату таблицы, используемой для хранения сообщений. Значение по умолчанию: **DBISettings** =
DBIUsername = {текст}	Имя пользователя для подключения к хранилищу **DBI**. Значение по умолчанию: **DBIUsername** =
DBIPassword = {текст}	Пароль для подключения к хранилищу **DBI**. Значение по умолчанию: **DBIPassword** =
SQLInsertCommand = {текст}	Команда добавления письма в хранилище **DBI**. Список элементов должен соответствовать формату таблицы в хранилище (см. ниже) и содержать следующие поля: 1. Номер сообщения. 2. Относительный путь к файлу, из которого взят объект. Формат файла: client/plug-in/id.prefix, где client – строка 'def', plug-in – имя подключаемого модуля, id – номер сообщения в шестнадцатеричном виде (при выводе используется 8 знаков), prefix – префикс, зависящий от значений параметров **FilenamesMode** и **FilenamesPrefix**. 3. Время помещения сообщения в базу данных. 4. Поле From из заголовков письма (заключенное в угловые скобки).

	5. Список получателей из заголовков письма. Значения разделены запятыми и заключены в угловые скобки.
	6. Тело сообщения.
	Элементы в запросе должны быть заменены знаками вопроса ("?").
	Пример:
	`SQLInsertCommand = "INSERT INTO mail_export(id, filename, put_time, sender, rcpts,body) values (?,?,?,?,?,?)"`
	Значение по умолчанию:
	`SQLInsertCommand =`

SQLRemoveCommand = {текст}	Команда удаления письма из хранилища **DBI**.
	Используется, если задано ограничение на время хранения писем в **Карантине**. Единственным параметром запроса является время, все сообщения старше которого должны быть удалены. Элемент в запросе должен быть заменен знаком вопроса ("?").
	Пример:
	`SQLRemoveCommand = "DELETE FROM mail_export WHERE put_time<=?"`
	Значение по умолчанию:
	`SQLRemoveCommand =`

SQLSelectCommand = {текст}	Команда доступа к письму в хранилище **DBI**.
	Используется, например, при запросе письма из **Карантина** через управляющее письмо. Единственным параметром запроса является относительное имя файла из карантина. Элемент в запросе должен быть заменен знаком вопроса ("?"). Список возвращаемых элементов фиксирован (должен соответствовать формату таблицы в хранилище, см. ниже):
	1. `id` – Номер сообщения.
	2. `put_time` – Время помещения сообщения в базу данных.
	3. `body` – Тело сообщения.
	4. `sender` – Поле `From` из заголовков письма (заключенное в угловые скобки).
	5. `rcpts` – Список получателей из заголовков письма. Значения разделены запятыми и заключены в угловые скобки.
	6. `filename` – Относительный путь к файлу, из которого взят объект.
	Пример:
	`SQLSelectCommand = "SELECT id,put_time,body,sender,rcpts,filename FROM mail_export WHERE filename LIKE ?"`
	Значение по умолчанию:
	`SQLSelectCommand =`

PulseTime = {время}	Промежуток времени, через который сообщения перемещаются из файлового хранилища в хранилище **DBI**, и удаляются устаревшие сообщения. При значении `0` запуск утилиты из значения параметра **PathToDrwebQp** производиться не будет.

	Значение по умолчанию: **PulseTime** = 5m
PathToDrwebQp = {путь к файлу}	Путь к утилите **drweb-qp**. Значение по умолчанию: **PathToDrwebQp** = %bin_dir/drweb-qp
MoveAll = {логический}	Перемещение всей входящей почты сразу в каталог /Path_parameter_value/def/backup/ для архивирования. Данный параметр имеет смысл при **MoveToDBI** = Yes, иначе каталог может быстро заполниться файлами с входящими письмами. Значение по умолчанию: **MoveAll** = No

Формат таблицы в базе данных, используемой для хранения сообщений Карантина через DBI

Если используется хранилище **DBI**, то для хранения писем в БД следует использовать таблицу, содержащую следующие поля (порядок следования полей не важен, но названия и типы полей должны совпадать с указанными ниже):

- **id** (number) – Номер (идентификатор) сообщения;
- **filename** (string) – Относительный путь к файлу, из которого взят объект;
- **put_time** (timestamp) – Время помещения сообщения в базу данных;
- **sender** (string) – Значение поля From: из заголовка письма, заключенное в угловые скобки;
- **rcpts** (string) – Список получателей из заголовка письма (TO:, CC:, BCC:). Значения разделены запятыми и заключены в угловые скобки;
- **body** (string) – Тело сообщения.

Пожалуйста, обратите внимание, что "типы данных", которые представлены в списке, должны быть заменены в БД на аналогичные типы данных, реально имеющиеся в СУБД (integer, varchar и т.д.).

Секция [Filters]

В секции [Filters] сосредоточены общие настройки работы подключаемых модулей **Dr.Web MailD**:

LibDir = {путь к каталогу}	Каталог, в котором располагаются подключаемые модули **Dr.Web MailD**. Значение по умолчанию: **LibDir** = %bin_dir/maild/plugins/
Settings = {список настроек}	Параметры запуска подключаемых модулей. Настройки модулей перечисляются через запятую в следующем формате: <настройки_модуля>, <настройки_модуля>..., где • <настройки_модуля> – это строка <название_модуля>: <параметр1>|...|<параметрN>, • <параметрN> – это пара имя_параметра = значение_параметра.

	Пример: **Settings** = vaderetro: **max_size** = 400k| **log_level**=debug, drweb: **max_size** = 10m Эта строка устанавливает для подключаемого модуля **Vaderetro** максимальный размер сообщения в 400 Кб и уровень подробности журнала в Debug, а для подключаемого модуля **Drweb** – максимальный размер сообщения в 10 Мб. Полный перечень параметров, которые могут задаваться в параметре **Settings**, перечислен ниже. Значения параметров (за исключением путей к файлам и имен файлов в UNIX) регистронезависимы.
	Значение по умолчанию: **Settings** =
BeforeQueueFilters = {список строк}	Список имен подключаемых модулей, обрабатывающих письмо до его помещения в хранилище сообщений (в синхронном режиме **before-queue**).
	Значение по умолчанию: **BeforeQueueFilters** =
MaxSizeBeforeQueueFilters = {размер}	Максимальный размер письма для обработки подключаемыми модулями, указанными в параметре **BeforeQueueFilters**. Используется только для тех подключаемых модулей, для которых значение параметра **max_size** (см. ниже) не задано явным образом в параметре **Settings** или в Правилах. При значении 0 ограничения отсутствуют.
	Значение по умолчанию: **MaxSizeBeforeQueueFilters** =
AfterQueueFilters = {список строк}	Список имен подключаемых модулей, обрабатывающих письмо после его помещения в хранилище сообщений (в асинхронном режиме **after-queue**).
	Значение по умолчанию: **AfterQueueFilters** =
MaxSizeAfterQueueFilters = {размер}	Максимальный размер письма для обработки подключаемыми модулями, указанными в параметре **AfterQueueFilters**. Используется только для тех подключаемых модулей, для которых значение параметра **max_size** (см. ниже) не задано явным образом в параметре **Settings** или в Правилах. При значении 0 ограничения отсутствуют.
	Значение по умолчанию: **MaxSizeAfterQueueFilters** = 0
PluginsBaseDir = {путь к каталогу}	Путь к каталогу, в котором хранятся рабочие файлы, используемые подключаемыми модулями. Например, в этом каталоге подключаемый модуль **Vaderetro** ищет файл используемой им библиотеки **VadeRetro.**
	Значение по умолчанию: **PluginsBaseDir** = %var_dir/plugins/

В текущей версии **Dr.Web для почтовых серверов UNIX** для каждого подключаемого модуля

могут быть заданы индивидуальные значения параметров, перечисленные в таблице ниже. При этом в параметре **Settings** имя параметра для модуля указывается в формате <название_модуля>: <параметр>, а если параметр может быть использован в Правилах обработки почты, то там используется формат <название_модуля>/<параметр>.

section = {текст}	Название секции в конфигурационном файле подключаемого модуля, в которой задаются настройки его работы (как правило, конфигурационный файл такого модуля состоит из одной секции). Если имя секции не указано, то используется секция с именем модуля.

max_size = {размер}
R C
	Максимальный разрешенный для подключаемого модуля размер проверяемого сообщения.
	При значении 0 ограничения отсутствуют.
	Ограничение на размер по умолчанию зависит от того, в какой очереди (**BeforeQueueFilters** или **AfterQueueFilters**) запускается модуль, и определяется, соответственно, значением параметра **MaxSizeBeforeQueueFilters** или **MaxSizeAfterQueueFilters**.
	Использование параметра в Правилах (в файле конфигурации):
	`[Rules]`
	`#Правило, истинное для всех писем`
	`true cont plugin_name/max_size = {размер}`
	Пример:
	`[Rules]`
	`...`
	`#Для писем с адреса admin@domain.com установить max_size модуля Drweb в величину 100k`
	`from:admin@domain.com cont drweb/max_size = 100k`

log_level = {уровень подробности} R	Уровень подробности ведения журнала работы подключаемого модуля. Допустимые значения: • Quiet • Error • Alert • Info • Debug Значение параметра по умолчанию совпадает со значением параметра **Level** секции [Logging]
log_ipc_level = {уровень подробности} R	Уровень подробности ведения журнала работы библиотеки IPC. Допустимые значения: • Quiet • Error • Alert • Info • Debug Значение параметра по умолчанию совпадает со значением параметра **IpcLevel** секции [Logging]

syslog_facility = {метка syslog}	R Метка записи в журнал при использовании системного сервиса **syslog** Значение параметра по умолчанию совпадает со значением параметра **SyslogFacility** секции [Logging].
log_filename = {syslog | путь к файлу}	Путь к файлу журнала или syslog, если журнал ведется с помощью системной службы **syslog**
path_to_lib = {путь к файлу}	R Путь к динамически загружаемой библиотеке подключаемого модуля, если имя библиотеки модуля не соответствует правилам именования lib<plugin_name>.so или если она расположена не в каталоге, заданном в параметре **LibDir**. Путь может быть как абсолютным, так и относительным. Относительный путь задается от каталога, указанного в параметре **LibDir**. Значение по умолчанию: **path_to_lib** = **LibDir**/lib<plugin_name>.so

Обратите внимание, что если основное количество писем является "тяжелыми" письмами (с большими вложениями, или с большим количеством малых вложений), то их проверка подключаемыми модулями будет осуществляться долго. В этом случае перемещение подключаемых модулей в очередь **BeforeQueueFilters** не рекомендуется, поскольку это замедляет взаимодействие с MTA при передаче писем.

Кроме того в этом случае возможно возникновение проблем при проверке писем из-за некорректно заданной (малой) величины тайм-аута **IpcTimeout** (секция [General]), что может привести к их потере (недоставке получателю без соответствующего уведомления об этом отправителя).

Секция [Rule]

Наборы настроек значений параметров, часто используемые в Правилах обработки писем (в части SETTINGS), могут быть объединены в именованные группы. Каждая такая именованная группа настроек описывается в основном конфигурационном файле **Dr.Web MailD** в виде секции специального вида:

```
[Rule: <имя группы>]
```

где <имя группы> – уникальное имя группы настроек, которое может содержать латинские символы, числа и пробелы. Не чувствительно к регистру.

Значения параметров в секции, представляющей некоторую группу настроек, задаются в виде пар <Параметр> = <Значение>, по одному параметру на строку (поэтому запятые, встречающиеся в значении параметров, экранировать в данном случае не нужно). Окончанием секции именованной группы настроек считается начало любой следующей секции конфигурационного файла (в том числе – секции определения другой именованной группы), либо окончание конфигурационного файла.

Настройки, определенные в некоторой именованной группе, могут быть применены в любом Правиле обработки писем при помощи директивы **rule**=<имя группы>. В текущей версии **Dr.Web MailD** в каждом Правиле обработки писем может быть использовано не более одного параметра **rule**. Количество секций именованных групп настроек, определенных пользователем, не ограничено.

Все секции описания именованных групп настроек должны быть заданы в основном конфигурационном файле выше секции [Rules].

Пример:

Эти строки задают секцию, описывающую именованную группу настроек `MySettings`, которая задает значения двух параметров (блокировать уведомления MailD и отключать перемещение в **Карантин**, их описание см. ниже):

```
[Rule:MySettings]
quarantine = no
notify = block
```

Следующие два Правила, заданные в конфигурационном файле, используют эту именованную группу настроек, чтобы установить соответствующие значения параметров `quarantine` and `notify`:

```
[Rules]
Rcpt:regex:example\.com cont     rule=MySettings
Sender:lol@foo.com && block:vir1 cont notify.Skip=allow, notify.Virus=allow,
rule=MySettings
```

Поле определения этих Правил будут блокироваться уведомления MailD и перемещение файлов в **Карантин** для писем, адрес получателя которых принадлежит домену **`example.com`**. Если же письмо отправлено с адреса `lol@foo.com` и в нем найден блокирующий объект `vir1`, то будут разрешены уведомления MailD о найденных вирусах, и о пропуске писем (причем для всех типов получателей уведомлений), а другие типы уведомлений и перемещение файлов в **Карантин** будет запрещено, эти настройки импортируются из используемой именованной группы настроек `MySettings` (описана в секции `[Rule:MySettings]`).

Секция параметров по умолчанию

В основной конфигурационный файл всегда включена секция *особой группы настроек* – секция настроек значений по умолчанию для тех параметров, которые не задаются в секциях конфигурационного файла, а могут задаваться только в Правилах. Эта группа настроек имеет строго определенное имя **`default`**. Соответственно, секция, описывающая настройки параметров в этой группе, имеет заголовок `[Rule: default]`, причем имя `default` в заголовке этой секции, как правило, опущено, и она называется просто `[Rule]`. Чтобы явно применить в любом своем Правиле настройки по умолчанию, вы можете использовать директиву **`rule`**=`default`.

Не следует путать секцию группы настроек настроек по умолчанию `[Rule]` с секцией `[Rules]`, в которой определяются Правила обработки писем.

В секции `[Rule]` устанавливаются значения по умолчанию для всех параметров, которые встречаются в Правилах.

См. описание этих параметров в разделе Параметры, используемые в настройках (SETTINGS) Правил.

Пример:

```
[Rule]
notify         = block
notify.Virus   = allow(any)
notify.Cured   = allow(admin:sender)
notify.Skip    = block
notify.Archive = allow(admin)
notify.Error   = allow(admin)
notify.Rule    = allow(admin)
notify.License = allow(admin)
notify.Malware = allow(any)
html = yes
```

Определены значения по умолчанию для параметров `notify` и `html`, которые будут использованы для всех писем, для которых они не будут явно переопределены в Правилах.

Секция [Rules]

В секции `[Rules]` основного конфигурационного файла хранятся общие Правила обработки всех писем.

Обратите внимание, что структура секции `[Rules]` принципиально отличается от структуры других секций: в ней не перечисляются пары `<Параметр> = <Значение>`, а задаются Правила обработки, по одному на строку (в случае если при записи Правил не используются принудительные переносы).

Подробно Правила обработки писем рассмотрены в разделе Правила обработки писем.

При определении параметров обработки применяемых для каждого письма все Правила, заданные в секции `[Rules]`, проверяются сверху вниз в порядке их задания, поэтому порядок задания Правил имеет значение: более специфические Правила следует размещать ранее более общих (см. подробности также в разделе Обработка сообщений).

В случае если задаваемые правила обработки связаны с пользователями (т.е. имеют в качестве условия почтовые адреса отправителя или получателя), и число пользователей с разными Правилами обработки велико, то настройка специфических параметров обработки при помощи задания Правил в секции конфигурационного файлов становится неэффективной, так как сложность поиска настроек в нем пропорциональна количеству Правил. В связи с этим рекомендуется использовать возможность сохранения Правил для каждого пользователя в локальную базу данных. В этом случае поиск сработавших Правил работает более эффективно, и, кроме того, оптимизируется использование памяти.

Наборы настроек значений параметров, часто используемые в Правила обработки писем (в части `SETTINGS`), могут быть объединены в именованные группы. Каждая такая именованная группа настроек с именем **<NAME>** описывается в основном конфигурационном файле **Dr.Web MailD** в виде секции специального вида `[Rule: <NAME>]`.

Помните, что все секции, описывающие именованные группы настроек, должны располагаться в конфигурационном файле выше секции описания Правил `[Rules]`.

Секция [Stat]

В секции `[Stat]` собраны параметры сбора статистики работы **Dr.Web MailD**:

Detail = {off | low | medium | high}	Уровень подробности статистики. Доступные уровни: • `off` – статистика отключена. Это увеличивает производительность программного комплекса, но в результате этого функции отправки отчетов или экспорта статистики теряют смысл. • `low` – ведется статистика только по всей компании. В результате, становится возможным пользоваться отчетами и экспортом статистики. • `medium` – к статистике, доступной на уровне `low`, добавляется статистика по группам. Для каждой группы можно отдельно настраивать необходимость ведения статистики. • `high` – к статистике, доступной на уровне `medium`, добавляется статистика по каждому зарегистрированному во встроенной базе данных пользователю. Для каждого пользователя можно отдельно настраивать необходимость ведения статистики. Доступ к статистике можно получить как через интерфейс интерактивного управления, так и через web-интерфейс.

	Статистика, собранная на уровне `low`, также передается в отчетах, если они включены. Значение по умолчанию: **`Detail`** = `low`
`Send` = {логический}	Отсылка отчета серверу статистики (или серверу **Центра управления Dr.Web**, если **Dr.Web для почтовых серверов UNIX** работает в составе антивирусной сети в режиме централизованной защиты). Значение по умолчанию: **`Send`** = `Yes`
`SendPeriod` = {время}	Промежуток времени, через который статистика отсылается на сервер. Значение по умолчанию: **`SendPeriod`** = `10m`
`Timeout` = {время}	Максимальное время ожидания ответа от сервера статистики. Значение по умолчанию: **`Timeout`** = `30s`

Существует возможность экспорта статистики средствами компонента **MailD core** с помощью типа Storage.

Для включения экспорта статистики через тип `Storage` необходимо установить `Yes` значением параметра **`ExportStat`** и заполнить, как минимум, один из следующих параметров командами экспорта статистики:

`ExportStat` = {логический}	Возможность осуществлять экспорт статистики в хранилища, перечисленные в соответствующих параметрах (см. ниже). Значение по умолчанию: **`ExportStat`** = `No`
`ExportBlockObjectsStorage` = {текст запроса}	Список параметров для экспорта статистики по заблокированным сообщениям. Сохранение данных запроса будет выполняться сразу после блокировки письма, но только если письмо было просканировано антивирусным модулем (экспорт статистики для писем, заблокированных из-за ошибок обработки производится не будет). Имена таблицы и полей в базе данных могут быть произвольными, но их тип должен совпадать с типом соответствующих экспортируемых значений. Поля в запросе должны идти в том же порядке, что и в БД. В запросе необязательно использовать все доступные значения. Поля текстового типа (`<varchar_long>`) должны быть заключены в одинарные кавычки ('). Список значений, которые можно сохранять в запросе: • `:number<int>` - уникальный номер сообщения; • `:q_name<varchar_long>` - путь к файлу **Карантина**, куда было сохранено письмо (если было сохранено); • `:virus_name<varchar_long>` - имя заблокированного объекта, найденного в письме;

	• `:virus_code<int>` – код заблокированного объекта, найденного в письме. Список кодов: ○ `1` – зараженный; ○ `2` – модификация вируса; ○ `3` – подозрительный; ○ `4` – излечен; ○ `5` – удален; ○ `6` – отклонен; ○ `7` – пропущен; ○ `8` – ограничения на проверку архивов; ○ `9` – ошибки; ○ `10` – ошибки чтения; ○ `11` – ошибки записи; ○ `12` – рекламная программа; ○ `13` – программа дозвона; ○ `14` – программа-шутка; ○ `15` – потенциально опасная программа; ○ `16` – программа взлома. • `:plugin_name<varchar_long>` – имя подключаемого модуля, заблокировавшего письмо; • `:sender<varchar_long>` – адрес отправителя, заключенный в угловые скобки; • `:client_ip<varchar_long>` – IP-адрес клиента, загрузившего письмо в систему (если доступен); • `:date<timestamp>` – дата помещения данной записи в базу писем; • `:client_id<varchar_long>` – уникальный идентификатор **Клиента**, для которого производится сохранение в базу писем (всегда равно '`def`'). **Пример:** `ExportBlockObjectsStorage = "odbc:insert into viruses values (:number<int>, ':q_name<varchar_long>', ':virus_name<varchar_long>', :virus_code<int>, ':plugin_name<varchar_long>', ':sender<varchar_long>', ':client_ip<varchar_long>', :date<timestamp>, ':client_id<varchar_long>)'"` Значение по умолчанию: **`ExportBlockObjectsStorage`** =
`ExportStatStorage` = {текст запроса}	Возможность осуществлять экспорт статистики в хранилища, перечисленные в соответствующих параметрах (см. ниже). Экспорт статистики по общему числу обработанных сообщений. Сохранение данных запроса будет выполняться при: • завершении приложения; • через интервал времени, указанный в значении параметра **`SendPeriod`**. Если статистика пуста (не было обработано ни одного сообщения), сохранение не производится. Имена таблицы и полей в базе данных могут быть

	произвольными, но их тип должен совпадать с типом соответствующих экспортируемых значений. Поля в запросе должны идти в том же порядке, что и в БД. В запросе необязательно использовать все доступные значения. Список значений, которые можно сохранять в запросе: • `:size<int>` – общий размер проверенных сообщений в байтах; • `:num<int>` – общее число проверенных сообщений; • `:q_num<int>` – число сообщений, сохраненных в **Карантине**; • `:r_num<int>` – число перенаправленных сообщений; • `:n_num<int>` – число сообщений, для которых были отправлены уведомления; • `:pass_num<int>` – число пропущенных сообщений; • `:reject_num<int>` – число отвергнутых сообщений; • `:discard_num<int>` – число отклоненных сообщений; • `:tempfail_num<int>` – число временно отклоненных сообщений; • `:date<timestamp>` – дата помещения записи в базу писем; • `:q_size<int>` – размер сообщений, сохраненных в **Карантине**; • `:r_size<int>` – размер перенаправленных сообщений; • `:n_size<int>` – размер сообщений, для которых были отправлены уведомления; • `:pass_size<int>` – размер пропущенных сообщений; • `:reject_size<int>` – размер отвергнутых сообщений; • `:discard_size<int>` – размер отклоненных сообщений; • `:tempfail_size<int>` – размер временно отклоненных сообщений; • `:work_time<int>` – время работы подключаемого модуля в миллисекундах. **Пример**: `ExportStatStorage = "odbc:insert into g_stat values(:size<int>, :num<int>, :q_num<int>, :r_num<int>, :n_num<int>, :pass_num<int>, :reject_num<int>, :discard_num<int>, :tempfail_num<int>, :date<timestamp>)"`
	Значение по умолчанию: `ExportStatStorage =`
`ExportPluginStatStorage = {текст запроса}`	Экспорт статистики по числу обработанных сообщений для каждого подключаемого модуля. Статистика сохраняется только для модулей, указанных в значении параметра **Names** секции настроек `[Reports]` (или для всех работающих, если значение данного параметра не задано). Сохранение будет выполняться при: • завершении приложения; • получении сигнала `SIGHUP`; • отправлении отчета администратору; • через определенный интервал времени, если отчеты высылаются не слишком часто. Если статистика пуста (не было обработано ни одного

	сообщения), сохранение не производится. Имена таблицы и полей в базе данных могут быть произвольными, но их тип должен совпадать с типом соответствующих экспортируемых значений. Поля в запросе должны идти в том же порядке, что и в БД. Список значений, которые можно сохранять в запросе: • те же, что и для параметра **ExportStatStorage**; • `:plugin_name<varchar_long>` - имя подключаемого модуля, для которого сохраняется статистика. **Пример:** **ExportPluginStatStorage =** "odbc:insert into plugin_stat values(':plugin_name<varchar_long>', :size<int>, :num<int>, :q_num<int>, :r_num<int>, :n_num<int>, :pass_num<int>, :reject_num<int>, :discard_num<int>, :tempfail_num<int>, :date<timestamp>)"
	Значение по умолчанию: **ExportPluginStatStorage** =

Более подробно с имеющимися возможностями экспорта статистики можно ознакомиться в разделе Экспорт статистики.

Секция [Reports]

В секции `[Reports]` собраны параметры создания и отправки отчетов о работе подключаемых модулей программного комплекса.

Send = {логический}	Отсылка отчетов. Значение по умолчанию: **Send** = Yes
SendTimes = {расписание}	График отправки отчетов. Синтаксис: • `hour:minute:second[-period]` - отправлять отчет в заданное время каждый день; • `Nw/hour:minute:second[-period]` - отправлять отчет в заданное время в N-й день недели (`0` - воскресенье, `1` - понедельник, `2` — вторник, и т.д.); • `Nm/hour:minute:second[-period]` - отправлять отчет в заданное время в N-й день месяца. Если указан конкретный промежуток времени (`period`), отчет будет составляться именно за заданный промежуток времени, если нет, то промежуток времени считается равным 24 часам. **Пример**: **SendTimes** = 00:00:00-24h, 1w/00:00:00-7d, 2M/21:23:32-31d В данном случае будут отправляться три отчета: ежедневный в полночь, еженедельный в полночь понедельника и ежемесячный во второй день месяца в 21:23:32. Значение по умолчанию: **SendTimes** = 24h
Mail = {адрес электронной почты}	Адрес, на который высылаются отчеты. Если данный параметр не задан, отчеты высылаются на адреса, указанные в значении параметра **AdminMail** в секции настроек `[Notifier]`. Возможно задание нескольких

	адресов через запятую. Пожалуйста, обратите внимание, что если для **Mail** задано значение, то на **AdminMail** отчеты приходить не будут. Значение по умолчанию: **Mail** =
Names = {список строк}	Список имен подключаемых модулей через запятую, для которых создается отчет. Если этот параметр не задан, отчет создается для модулей, перечисленных в значениях параметров **BeforeQueueFilter** и **AfterQueueFilter** секции настроек [Filters]. Значение по умолчанию: **Names** =
TopListSize = {числовое значение}	Показ в отчете списков часто блокируемых объектов и адресов, с которых присылается наибольшее количество блокируемых объектов. Значение параметра определяет количество записей в каждом списке. При значении 0 списки не создаются. При значении -1 размер списков не ограничен. Значение по умолчанию: **TopListSize** = 20
MaxStoreInDbPeriod = {время}	Максимальное время хранения статистики в базе отчетов. При значении 0 старые записи удаляться не будут. Обратите внимание, что это параметр устарел и более не используется (изменение его значения не влияет на работу **Dr.Web MailD**)! Значение по умолчанию: **MaxStoreInDbPeriod** = 31d
CheckForRemovePeriod = {время}	Промежуток времени, через который старые записи будут удаляться из базы отчетов. Обратите внимание, что это параметр устарел и более не используется (изменение его значения не влияет на работу **Dr.Web MailD**)! Значение по умолчанию: **CheckForRemovePeriod** = 5m

Секция [Logging]

В секции [Logging] собраны параметры, управляющие ведением журналов работы основных модулей, образующих комплексный компонент **Dr.Web MailD**:

Level = {уровень подробности}	Уровень подробности сохранения в журнал работы компонента общих событий. Допускается использование следующих уровней: • Quiet • Error • Alert • Info • Debug

	Значение по умолчанию: **Level** = Info
IPCLevel = {уровень подробности}	Уровень подробности сохранения в журнал работы компонента событий подсистемы IPC. Допускается использование следующих уровней: • Quiet • Error • Alert • Info • Debug Значение по умолчанию: **IPCLevel** = Alert
SyslogFacility = {метка syslog}	Метка записи при использовании системного сервиса **syslog** Значение по умолчанию: **SyslogFacility** = Mail
FileName = {syslog | путь к файлу}	Имя файла журнала или syslog, если нужно использовать системный сервис **syslog** Значение по умолчанию: **FileName** = syslog

Использование SASL

В данном разделе описаны секции конфигурационного файла, содержащие параметры работы сервиса **Dr.Web для почтовых серверов UNIX** в режиме аутентификации соединений с использованием SASL.

Поскольку в **Dr.Web для почтовых серверов UNIX** аутентификация SASL реализована только через драйвер **Cyrus SASL,** при использовании аутентификации SASL должны быть заданы параметры в обеих секциях.

См. также пример настройки аутентификации **Cyrus SASL.**

Секция [SASL]

В секции [SASL] собраны параметры аутентификации SASL в версии **Dr.Web для почтовых серверов UNIX**, предназначенной для работы в режиме **SMTP/LMTP-прокси**:

Use = {логический}	Подключение возможности использования аутентификации SASL. Значение по умолчанию: **Use** = No
Driver = {текст}	Имя драйвера аутентификации SASL. В текущей версии продукта доступен только драйвер cyrus. Для его использования необходимо поставить и настроить библиотеку cyrus-sasl2. Значение по умолчанию: **Driver** = cyrus
BrokenAuthClients = {логический}	Возможность поддержки устаревших SMTP-клиентов,

	использующих нестандартный синтаксис протокола AUTH. Значение по умолчанию: **BrokenAuthClients** = Yes
AuthenticatedHeader = {логический}	Возможность добавления имен зарегистрированных пользователей к заголовку Received. При значении Yes данного параметра имена зарегистрированных пользователей видны всем. Значение по умолчанию: **AuthenticatedHeader** = No

Секция [Cyrus-SASL]

В секции [Cyrus-SASL] собраны параметры, управляющие работой SASL-драйвера cyrus-sasl (Cyrus SASL).

Lib = {путь к файлу}	Абсолютный путь к библиотеке cyrus-sasl2. Значение по умолчанию: **Lib** = /usr/lib/libsasl2.so.2
Path = {текст}	Имя конфигурационного файла (к значению параметра добавляется расширение .conf), из которого библиотека cyrus-sasl2 получает свои настройки. Обратите внимание, что **Dr.Web MailD** не проверяет корректность и существование файла, указанного в этом параметре. В случае если файл отсутствует или некорректный, библиотека cyrus-sasl2 будет автоматически использовать свои настройки по умолчанию, и об этом не будет сформировано никакого уведомления. Значение по умолчанию: **Path** = maild
ServerHostname = {текст}	Имя хоста. Определяет FQDN домена, которое при авторизации будет автоматически добавляться как @domain к части user, переданной клиентом (если часть user используется как UID и передана только она). Полученная строка user@domain будет использоваться в качестве логина, который **saslauthd** будет искать в базах для авторизации. Если значение параметра не задано, в качестве имени хоста используется значение параметра **Hostname** из секции настроек [General]. Если значение этого параметра также не задано, то в качестве имени хоста используется значение, возвращаемое функцией gethostname. Обратите внимание, что если задано значение параметра **ServerRealm** (см. ниже), или значение Realm/Domain при аутентификации передается клиентом, то к user добавляются именно они, а параметр **ServerHostname** игнорируется. Значение по умолчанию: **ServerHostname** =
ServerRealm = {текст}	Область SASL (SASL realm), в которой находится сервер. Определяет FQDN домена, которое при авторизации будет автоматически добавляться как @domain к части user,

	переданной клиентом. Полученная строка `user@domain` будет использоваться в качестве логина, который **saslauthd** будет искать в базах для авторизации. Если не задано, и клиент не передал значение `Realm/Domain` при аутентификации, в качестве подставляемого FQDN будет использован параметр **ServerHostname** Обратите внимание, что для корректной обработки логина в виде строки `user@domain` для демона **saslauthd** должна быть указана опция `-r`. Значение по умолчанию: **ServerRealm** =
SecurityOptions = {текст}	Список настроек безопасности, перечисленных через запятую. Доступны следующие настройки: • `noplaintext` – запрещение механизмов аутентификации, восприимчивых к простым пассивным атакам (например, `PLAIN`, `LOGIN`); • `noactive` – защита от активных (не словарных) атак во время обменной аутентификации; • `nodictionary` – запрещение механизмов аутентификации, восприимчивых к пассивным словарным атакам; • `noanonymous` – запрещение механизмов аутентификации, позволяющих анонимный вход; • `mutual_auth` – требование обоюдной аутентификации. Значение по умолчанию: **SecurityOptions** = noanonymous

Подключения почтовых систем

В данном разделе описаны секции конфигурационного файла, содержашие параметры взаимодействия **Dr.Web для почтовых серверов UNIX** с почтовыми системами.

В конфигурационном файле всегда должны присутствовать секции, задающие параметры взаимодействия с выбранной почтовой системой (а в режиме фильтрации писем, отправляемых на почтовый клиент – еще и секция `[POP3]` или `[IMAP]`, в зависимости от протокола, используемого клиентом).

В зависимости от того, с какой почтовой системой интегрируется **Dr.Web для почтовых серверов UNIX**, используются различные модули, выполняющие функции компонентов **Receiver** и **Sender**. В таблице ниже перечислены различные почтовые системы, с которыми может быть интегрирован **Dr.Web для почтовых серверов UNIX**. Для каждого варианта указано, какие модули из состава **Dr.Web MailD** используются в качестве компонентов **Receiver** и **Sender,** а также какие секции конфигурационного файла они используют для настройки своей работы.

Почтовая система	Режим подключения	Используемые модули Receiver и Sender и секции их настройки	
Sendmail (или любая, работающая через Milter)	—	**drweb-milter** **drweb-sender**	`[Milter]` `[Sender]`
Postfix	**before-queue**	**drweb-receiver** **drweb-sender**	`[Receiver]` `[Sender]`

Почтовая система	Режим подключения	Используемые модули Receiver и Sender и секции их настройки	
	after-queue	**drweb-receiver** **drweb-sender**	[Receiver] [Sender]
	Milter	**drweb-milter** **drweb-sender**	[Milter] [Sender]
Exim	—	**drweb-receiver** **drweb-sender**	[Receiver] [Sender]
CommuniGate Pro	—	**drweb-cgp-receiver** **drweb-cgp-sender**	[CgpReceiver] [CgpSender]
Courier	—	**drweb-courier** **drweb-sender**	[Courier] [Sender]
Zmailer	—	**drweb-zmailer** **drweb-sender**	[Receiver] [Sender]
Qmail	—	**drweb-qmail** **drweb-sender**	[Qmail] [Sender]
SMTP/LMTP proxy (по умолчанию)	—	**drweb-receiver** **drweb-sender**	[Receiver] [Sender]

Более подробно с процедурой настройки взаимодействия **Dr.Web для почтовых серверов UNIX** с той или иной почтовой системой вы можете ознакомиться в разделе Интеграция с почтовыми системами.

Секция [Receiver]

В секции `[Receiver]` собраны настройки компонента **Receiver** в тех версиях **Dr.Web для почтовых серверов UNIX**, которые предназначены для работы с почтовыми системами **Exim**, **Zmailer** и **Postfix** (в случае если **Postfix** не использует протокол **`Milter`**) и в версии, предназначенной для работы в режиме **SMTP/LMTP-прокси**.

1. Общие параметры работы компонента Receiver

`Address` = `{адрес}`	Адрес, используемый компонентом **Receiver** для получения сообщений. Задается сокет, через который получаются сообщения (либо TCP-сокет, либо UNIX-сокет). Значение по умолчанию: **`Address`** = `inet:25@0.0.0.0`
`PoolOptions` = `{настройки пула}`	Настройки пула потоков компонента **Receiver**. Значение по умолчанию: **`PoolOptions`** = `auto`
`RealClients` = `{логический}`	Возможность приема соединений напрямую от клиентов. Значение по умолчанию: **`RealClients`** = `Yes`
`ProcessingErrors` = `{действие}`	Действие, совершаемое над письмом в случае возникновения каких-либо ошибок.

	Значением параметра может быть только одно из основных действий: `tempfail`, `discard`, `reject` Значение по умолчанию: **`ProcessingErrors`** = `reject`
`StalledProcessingInterval` = {время}	Промежуток времени для обработки "потерянных" писем. "Потерянные" письма – сообщения, полученные компонентом **Receiver**, но по каким-либо причинам не обработанные подключаемыми модулями вовремя, из-за чего они не были переданы компоненту **MailD core**. Такая ситуация может случиться при возникновении проблем с сетью или питанием. При обнаружении таких сообщений компонент **Receiver** ставит их в очередь на обработку. Значение по умолчанию: **`StalledProcessingInterval`** = `10m`
`OneCommandTimeout` = {время}	Максимальный промежуток времени на исполнение одной команды. Значение по умолчанию: **`OneCommandTimeout`** = `5m`
`OneMessageTimeout` = {время}	Максимальный промежуток времени на получение одного сообщения. Значение по умолчанию: **`OneMessageTimeout`** = `10m`
`AddReceivedHeader` = {логический}	Добавление заголовка `Received` ко всем получаемым сообщениям. Значение по умолчанию: **`AddReceivedHeader`** = `Yes`
`ReturnReject` = {логический}	Параметр определяет поведение компонента **Receiver** в случае применения действия `reject` (отклонение письма с уведомлением) к письму, обрабатываемому в синхронном режиме. При значении `Yes` возвращается ошибка `SMTP 55*`, а при значении `No` возвращается положительный ответ `SMTP 250`, но отправителю сообщения высылается DSN (если не запрещена параметром **`SkipDSNOnBlock`**). Ответ, возвращаемый клиенту, дополняется строкой `Reply<Reason>`, заданной в настройках подключаемого модуля, выполнившего действие `reject`, но только в том случае, если это разрешено его настройкой **`UseCustomReply`**=`Yes` (где `<Reason>` – причина срабатывания действия `reject`). В противном случае будет выведено стандартное сообщение `"The message has been rejected by the Dr.Web MailD"`. Обратите внимание, что если **Dr.Web MailD** работает не в режиме **SMTP/LMTP-прокси**, а сопряжен с MTA, рекомендуется установить значение `No`, чтобы гарантировать корректное уведомление отправителя о том, что его сообщение было отвергнуто (поскольку возможно возникновение ситуации, когда MTA перед проверкой письма уже успела подтвердить отправителю его успешный прием).

	Если **ReturnReject**=No, то рекомендуется в действиях подключаемого модуля указать в дополнение к обязательному действию reject дополнительное действие notify (так как в рамках SMTP-сессии **Receiver**, отклонив сообщение, выдаст код об успешной обработке письма 250), или разрешить DSN. Но разрешать DSN (управляется параметром SkipDSNOnBlock в секции [Maild]) стоит только в том случае, если есть уверенность, что поток отклоняемых писем невелик, иначе может возникнуть лавинная нагрузка на MTA по отправке DSN. Значение по умолчанию: **ReturnReject** = Yes
GreetingString = {текст}	Строка, выводимая в качестве приветствия при подключении нового SMTP-клиента. Макрос %host% заменяется на значение параметра **Hostname** из секции [General], макрос %ver% заменяется на текущую версию модуля **drweb-receiver**. Значение по умолчанию: **GreetingString** = "%host% Dr.Web SMTP receiver v%ver% ready"
RelayDomains = {Lookup}	Список доменов, которым разрешена пересылка почты. При указании обычного списка доменов, для которых **Dr.Web MailD** будет являться почтовым релеем, их поддомены не учитываются, т.е. почта, приходящая от их поддоменов, пересылаться не будет. Для задания поддоменов возможно использование регулярного выражения или задание файла со списком регулярных выражений rfile (используется синтаксис регулярных выражений **Perl**). Обратите внимание, что значение параметра – Lookup. **Пример**: **RelayDomains** = regex:.*.domain.com Будет разрешена пересылка для всех поддоменов domain.com. **Пример**: **RelayDomains** = rfile:/path rfile содержит список регулярных выражений, каждое из которых должно располагаться на новой строке: .*.domain.com .*.domain1.com .*.domain2.com В текущей версии **Dr.Web MailD** параметр **RelayDomains** не поддерживает формат записей wildcard DNS. Таким образом, подобная запись некорректна: **RelayDomains** = *.domain Значение по умолчанию: **RelayDomains** =
RestrictionStat = {логический}	Сбор статистики по срабатыванию ограничений SMTP-сессии (про ограничения см. ниже). Получить статистику можно, послав сигнал SIGUSER1 процессу **drweb-receiver**. Статистика хранится в файле restrictions.txt в каталоге, указанном в значении

	параметра **BaseDir** секции настроек [General]. Значение по умолчанию: **RestrictionStat** = No
DelayRejectToRcpt = {логический}	Приостановка блокирования письма до стадии RCPT, даже если какое-либо ограничение сработало раньше. Установка данного параметра позволяет работать с устаревшими версиями почтовых клиентов и выводить список заблокированных адресов получателей в лог. Значение по умолчанию: **DelayRejectToRcpt** = Yes

2. Числовые ограничения SMTP-сессии

Перечисленные здесь параметры позволяют задавать ряд численных ограничений, при нарушении которых протокольный диалог SMTP с клиентом прерывается.

MaxRecipients = {числовое значение}	Максимальное количество получателей одного письма (число SMTP-команд RCPT TO). При значении 0 ограничений нет. Если IP-адрес, с которого было установлено данное соединение, отмечен как trusted, то данное ограничение не проверяется Значение по умолчанию: **MaxRecipients** = 100
MaxConcurrentConnection = {числовое значение}	Максимальное количество одновременных SMTP-подключений с одного IP-адреса. При значении 0 ограничений нет. Значение по умолчанию: **MaxConcurrentConnection** = 5
MaxMailsPerSession = {числовое значение}	Максимальное количество писем (SMTP-команд MAIL FROM), которые может отправить клиент за одну сессию. При значении 0 ограничений нет. Значение по умолчанию: **MaxMailsPerSession** = 20
MaxReceivedHeaders = {числовое значение}	Максимальное количество заголовков Received. При значении 0 ограничений нет. Проверка **MaxReceivedHeaders** будет осуществляться компонентом **Receiver** всегда, независимо от того, отмечен ли IP-адрес, с которого было установлено данное соединение, как trusted, или нет. Значение по умолчанию: **MaxReceivedHeaders** = 100
MaxErrorsPerSession = {числовое значение}	Максимальное количество ошибок за одну сессию. При значении 0 ограничений нет. Значение по умолчанию: **MaxErrorsPerSession** = 10

MaxMsgSize = {размер}	Максимальный размер сообщения (переданного в SMTP-команде DATA). Проверка **MaxMsgSize** будет осуществляться компонентом **Receiver** всегда, независимо от того, отмечен ли IP-адрес, с которого было установлено данное соединение, как trusted, или нет. Значение по умолчанию: **MaxMsgSize** = 10m
MaxJunkCommands = {числовое значение}	Максимальное число SMTP-команд RSET, NOOP и VRFY на сессию. Если число команд превысит указанное значение, то начнет увеличиваться счетчик ошибок. Значение счетчика сбрасывается при каждой успешной обработке письма модулем **drweb-maild**. Если значение равно 0, то данное ограничение не используется. Значение по умолчанию: **MaxJunkCommands** = 100
MaxHELOCommands = {числовое значение}	Максимальное число SMTP-команд HELO, EHLO и LHLO на сессию. Если число команд превысит указанное значение, то начнет увеличиваться счетчик ошибок. Значение счетчика сбрасывается при каждой успешной обработке письма модулем **drweb-maild**. Если значение равно 0, то данное ограничение не учитывается. Значение по умолчанию: **MaxHELOCommands** = 20

Обратите внимание, что некоторые ограничения на граничные значения параметров проверяются даже для доверенных (trusted) соединений. В таблице ниже перечислено поведение проверок для доверенных соединений, а также указано, какие сообщения отправляются отправителю сообщений при их срабатывании.

Ограничение	Сообщение отправителю при срабатывании	Выполняется ли для trusted-соединений
MaxRecipients	452 4.5.3 Too many rcpts	Нет
MaxConcurrentConnection	421 4.7.0 Too many concurrent SMTP connections from this IP address; please try again later	Нет
MaxMailsPerSession	421 4.2.1 too many messages in this connection	Нет
MaxReceivedHeaders	554 5.7.0 MailD error: Too many received headers: N	Да
MaxErrorsPerSession	421 4.7.0 Error: too many errors	Нет
MaxMsgSize	552 5.3.4 Message size exceeds file system imposed limit	Да
MaxJunkCommands	421 4.7.0 Error: too many errors	Нет
MaxHELOCommands	421 4.7.0 Error: too many errors	Нет

3. Ограничения и проверки различных этапов SMTP-сессии

Следующие несколько параметров **(*Restrictions)** определяют **SMTP-ограничения**, т.е. проверки, которым подвергаются IP-адреса соединений, не отмеченные как надежные (надежными являются соединения, для которых установлен внутренний флаг `trusted`), на различных этапах SMTP-сессии. По умолчанию надежными считаются соединения с `localhost` и от UNIX-сокетов. Ограничения позволяют фильтровать нежелательную корреспонденцию в модуле **drweb-receiver** на этапе SMTP-сессии еще до передачи писем на проверку модулю **drweb-maild**, экономя таким образом ресурсы и добавляя дополнительный уровень фильтрации спама (тем самым повышая вероятность его обнаружения).

SMTP-ограничения работают на следующих этапах SMTP-сессии:

- на этапе подключения нового клиента (INTRO) – используются ограничения, указанные в параметре **SessionRestrictions**;
- при получении команды `HELO/EHLO` – используются ограничения, указанные в параметре **HeloRestrictions**;
- при получении команды `FROM`, когда клиент устанавливает отправителя для нового письма – используются ограничения, указанные в параметре **SenderRestrictions**;
- при получении команды `RCPT`, когда клиент добавляет нового получателя к текущему письму – используются ограничения, указанные в параметре **RecipientRestrictions**;
- при получении команды `DATA`, когда клиент закончил передавать всех получателей письма и готов отправлять тело письма – используются ограничения, указанные в параметре **DataRestrictions**.

Ограничения задаются через запятую в каждой из настроек ***Restrictions** и проверяются в порядке их задания – слева направо. Проверка ограничений происходит только после проверки остальных условий, таких как порядок следования команд, корректность их параметров и т.п. Все ограничения проверяются последовательно до тех пор, пока для сообщения не будет установлен флаг `trusted`. Как только он будет установлен для соединения, то проверка всех последующих ограничений не производится.

SessionRestrictions = {список ограничений}	Параметр задает проверки, осуществляемые непосредственно в начале соединения (INTRO). Специфические для этого этапа действия: • `trust_protected_network` • `trust_protected_domains` • `trust_white_networks` • `trust_white_domains` • `reject_dnsbl` • `reject_black_networks` • `reject_black_domains`
	Значение по умолчанию: **SessionRestrictions** = trust_protected_network
HeloRestrictions = {список ограничений}	Проверки, выполняемые на стадии `HELO/EHLO`. Специфические для этого этапа действия: • `reject_unknown_hostname` • `reject_diff_ip`
	Значение по умолчанию: **HeloRestrictions** =

SenderRestrictions = {список ограничений}	Проверки, выполняемые на стадии FROM. Специфические для этого этапа действия: • reject_unknown_sndrs • reject_unknown_domain • trust_sasl_authenticated • pass_sasl_authenticated. Значение по умолчанию: **SenderRestrictions** = trust_sasl_authenticated
RecipientRestrictions = {список ограничений}	Проверки, выполняемые на стадии RCPT. Все заявленные получатели проверяются по очереди. Специфические для этого этапа действия: • reject_unknown_domain • reject_unauth_destination • reject_unknown_rcpts • pass_sasl_authenticated Значение по умолчанию: **RecipientRestrictions** = reject_unauth_destination
DataRestrictions = {список ограничений}	Проверки, выполняемые на стадии DATA. Специфические для этого этапа действия: • reject_spam_trap • reject_multi_recipient_bounce • pass_sasl_authenticated Значение по умолчанию: **DataRestrictions** =

Эффект от блокировки по ограничению различен в зависимости от этапа SMTP-сессии, на которой происходит проверка. В **SessionRestrictions** (на этапе INTRO) блокировка происходит на всю сессию – т.е. в ответ на все дальнейшие команды от пользователя возвращаются ошибки. На всех остальных этапах блокировка происходит только для конкретной указанной SMTP-команды.

Каждое из ограничений может принимать в качестве необязательного параметра дополнительное значение счета [SCORE] (кроме **set_score** и **add_score**, где значение счета является единственным обязательным параметром). В зависимости от типа ограничения данный счет обрабатывается по разному:

- ограничение может сработать, если текущий счет письма меньше указанного в параметре;
- ограничение может сработать, если текущий счет письма больше указанного в параметре;
- если ограничение срабатывает, вместо его основного действия выполняется добавление к текущему счету письма заданного в параметре значения.

В зависимости от этапа, на котором работает ограничение, оно работает с соответствующим счетом: если ограничение находится на этапе **SessionRestrictions** или **HeloRestrictions**, то работа происходит со счетом, который будет добавляться к каждому сообщению, передаваемому в этой сессии; для ограничений на остальных этапах работа происходит со счетом для каждого конкретного обрабатываемого сообщения.

Для каждого этапа проверки ограничений существуют свои собственные (специфичные) ограничения, а также имеются ограничения, которые можно использовать на всех этапах SMTP-сессии. К последним относятся:

Действие	Описание
sleep {time} [SCORE]	Приостановить SMTP-соединение на заданное время (в секундах). Если указан SCORE, то действие выполняется только в том случае, если текущий счет больше значения, указанного в параметре.
reject [SCORE]	Вернуть постоянную ошибку SMTP (код `5*`). Если указан SCORE, то ошибка возвращается только в том случае, если текущий счет больше значения, указанного в параметре.
tempfail [SCORE]	Вернуть временную ошибку (код `4*`). Если указан SCORE, то временная ошибка возвращается только в том случае, если текущий счет больше значения, указанного в параметре.
mark_trust [SCORE]	Установить для соединения флаг `trusted`. Если есть еще ограничения после даного, то они будут пропущены. Если указан SCORE, то флаг `trusted` устанавливается только если текущий счет меньше указанного в параметре.
set_score SCORE	Заменяет текущий счет на значение SCORE. Если используется на этапах **SessionRestrictions** или **HeloRestrictions**, то влияет на счет каждого проходящего в сессии сообщения, а на всех остальных этапах влияет на счет конкретного обрабатываемого сообщения.
add_score SCORE	Добавляет к значению счета заданное значение. Если используется на этапах **SessionRestrictions** или **HeloRestrictions**, то влияет на счет каждого проходящего в сессии сообщения, а на всех остальных этапах влияет на счет конкретного обрабатываемого сообщения.

Ограничения, специфичные для различных этапов проверки:

Действие	Описание
Действия, специфичные для ограничения `SessionRestrictions`	
trust_protected_network [SCORE]	Если IP-адрес соединения находится в списке, определенном значением параметра **ProtectedNetworks** из секции `[Maild]`, то либо адрес помечается как доверенный IP-адрес, либо, если указан SCORE, значение SCORE прибавляется к счету каждого письма, передаваемого в этой сессии, а также – к счету IP-адреса отправителя.
trust_protected_domains [SCORE]	Проверка, находится ли IP-адрес соединения в списке, определенном значением параметра **ProtectedDomains** из секции `[Maild]`. Проверка осуществляется посредством двойного DNS-запроса. Сначала производится PTR-запрос и проверяется, находится ли полученное имя хоста в списке **ProtectedDomains**. Если этот домен есть в списке, производится A-запрос и проверяется, находится ли IP-адрес соединения в полученном списке адресов. При совпадении адрес либо помечается как доверенный IP-адрес, либо, если указан SCORE, значение SCORE прибавляется к счету каждого письма, передаваемого в этой сессии, а также - к счету IP-адреса отправителя.
trust_white_networks [SCORE]	Если IP-адрес соединения находится в белом списке, определенном значением параметра **WhiteNetworks** (см. ниже), то адрес либо помечается как доверенный IP-адрес, либо, если указан SCORE, значение SCORE прибавляется к счету каждого письма, передаваемого в этой

Действие	Описание
	сессии, а также – к счету IP-адреса отправителя.
trust_white_domains [SCORE]	Проверка, находится ли домен, которому принадлежит IP-адрес, с которого произведено соединение, в белом списке, определенном значением параметра **WhiteDomains** (см. ниже). Для этого производится PTR-запрос. При совпадении адрес либо помечается как доверенный IP-адрес, либо, если указан SCORE, значение SCORE прибавляется к счету каждого письма, передаваемого в этой сессии, а также - к счету IP-адреса отправителя.
reject_dnsbl [SCORE]	Проверка, находится ли IP-адрес соединения в черных списках RBL/DNSBL, определенных значением параметра **DNSBLList** (см. ниже). Для этого к сначала проверяется доступность сервера, и, в случае его доступности, производится A-запрос DNSBL. Доступность DNSBL-сервера проверяется отправкой ему тестового запроса, определенного спецификацией (127.0.0.2), на который сервер обязан отвечать положительно. Если сервер не ответил на данный запрос утвердительно, он считается недоступным, запись о чем фиксируется в журнале. В случае если какой-либо DNSBL-сервер положительно ответил на A-запрос DNSBL относительно IP-адреса отправителя, сессия закрывается, либо (если указан SCORE) в журнал выводится ошибка, а значение SCORE прибавляется к счету каждого письма, передаваемого в этой сессии, а также к счету IP-адреса отправителя. Обратите внимание, что если при проверке не удалось опросить ни один DNSBL-сервер, отправитель будет считаться надежным, и **reject_dnsbl** для него не сработает (но в журнале будет зафиксировано, что опросить DNSBL-сервера не удалось).
reject_black_networks [SCORE]	Если IP-адрес соединения находится в черном списке, определенном значением параметра **BlackNetworks** (см. ниже), сессия закрывается, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету каждого письма, передаваемого в этой сессии, а также – к счету IP-адреса отправителя.
reject_black_domains [SCORE]	Проверка, находится ли домен, с которого произведено подключение, в черном списке, определенном значением параметра **BlackDomains** (см. ниже). Для этого производится PTR-запрос. При совпадении установленного имени домена хотя бы с одним доменом в списке **BlackDomains** сессия закрывается, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету каждого письма, передаваемого в этой сессии, а также – к счету IP-адреса отправителя.
Действия, специфичные для ограничения HeloRestrictions	
reject_unknown_hostname [SCORE]	Если имя хоста не имеет ни DNS A-, ни DNS MX-записи, то почта с такого адреса блокируется, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету каждого письма, передаваемого в этой сессии, а также данный счет прибавляется к счету IP-адреса отправителя. В процессе проверки производятся A-запросы и иногда MX-запросы.
reject_diff_ip [SCORE]	Если IP-адрес клиента не совпадает ни с одним из IP-адресов, определенных для указанного в EHLO/HELO доменного имени, то почта с такого адреса блокируется. В случае если указан аргумент SCORE, письмо пропускается, но в лог выводится ошибка, а значение SCORE прибавляется к счету каждого письма, передаваемого в этой сессии, а также к счету IP-адреса отправителя.
Действия, специфичные для ограничения SenderRestrictions	

Действие	Описание
reject_unknown_domain [SCORE]	Если имя хоста отправителя не имеет ни DNS A, ни DNS MX записи, почта с такого адреса блокируется, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету передаваемого письма. В процессе проверки производятся A-запросы и иногда MX-запросы. Рекомендуется использовать **reject_unknown_domain** вместе с другими ограничениями для данного этапа сессии (т.е. первым **reject_unknown_domain**, а далее остальные проверки для данной стадии сессии). Это связано с тем, что если в поле FROM конверта письма отсутствует доменное имя, то ограничение не сработает, т.к. нет имени домена, которое можно проверить в DNS. Запрет же на прием писем с пустым TO или FROM невозможен, т.к. согласно RFC 5321 **Dr.Web MailD** должен всегда иметь возможность принимать уведомления DSN и MSN, поле FROM которых содержит пустой адрес <>.
trust_sasl_authenticated [SCORE]	Если SASL-аутентификация IP-адреса была успешной, то он помечается как доверенный, либо, если указан SCORE, то только если и его текущий счет также меньше указанного в параметре.
pass_sasl_authenticated [SCORE]	Пропустить все остальные проверки на данном этапе SMTP-сессии, если клиент успешно прошел аутентификацию SASL. Если указан SCORE, то пропуск проверок для прошедшего аутентификацию SASL клиента выполняется только если и текущий счет также меньше указанного в параметре.
reject_unknown_sndrs [SCORE]	Проверяет отправителя на присутствие в списке **ProtectedSenderEmails** (см. ниже). Если адреса в этом списке нет, почта от такого отправителя блокируется, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету передаваемого письма. Рекомендуется использовать вместе с Reputation IP Filter **anti_dha**.
Действия, специфичные для ограничения RecipientRestrictions	
reject_unknown_domain [SCORE]	Если имя хоста получателя не имеет ни DNS A, ни DNS MX записи, почта на такой адрес блокируется, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету передаваемого письма. В процессе проверки производится A-запросы и иногда MX-запросы. Рекомендации по использованию ограничения **reject_unknown_domain** вместе с другими ограничениями для данного этапа сессии аналогичны приведенным выше (для этапа **SenderRestrictions**), только они распространяются на анализ содержимого поля TO.
reject_unauth_destination [SCORE]	Если домена получателя нет ни в списке **RelayDomains**, ни в списке **ProtectedDomains** из секции [Maild], почта на такой адрес блокируется, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету передаваемого письма. В случае если осуществляется также прием почты и для поддоменов (параметр **IncludeSubdomains** установлен в Yes), то на стадии RCPT помимо проверки reject_unauth_destination должна обязательно присутствовать проверка reject_unknown_domain, в противном случае почта будет приниматься даже для несуществующих поддоменов защищаемых доменов.
reject_unknown_rcpts [SCORE]	Проверяет получателя на присутствие в списке **ProtectedEmails** (см. ниже). Если адреса в этом списке нет, почта на такой адрес блокируется, либо,

Действие	Описание
	если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету передаваемого письма. Рекомендуется использовать вместе с Reputation IP Filter **anti_dha**.
pass_sasl_authenticated [SCORE]	Пропустить все остальные проверки на данном этапе SMTP-сессии, если клиент успешно прошел аутентификацию SASL. Если указан SCORE, то пропуск проверок для прошедшего аутентификацию SASL клиента выполняется только если и текущий счет также меньше указанного в параметре.
Действия, специфичные для ограничения DataRestrictions	
reject_spam_trap [SCORE]	Проверка на спам-ловушку. Адрес получателя должен иметь формат <USER@HOST>. Если имя хоста находится в списке, определенном значением параметра **ProtectedDomains** (если этот список не пуст, см. ниже), и имя пользователя находится в списке, определенном значением параметра **SpamTrap** (см. ниже), сообщение блокируется, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету передаваемого письма. В списке **SpamTrap** может быть также определен полный электронный адрес.
reject_multi_recipient_bounce [SCORE]	Блокирование сообщений с пустым полем FROM и несколькими получателями, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету передаваемого письма.
pass_sasl_authenticated [SCORE]	Пропустить все остальные проверки на данном этапе SMTP-сессии, если клиент успешно прошел аутентификацию SASL. Если указан SCORE, то пропуск проверок для прошедшего аутентификацию SASL клиента выполняется только если и текущий счет также меньше указанного в параметре.

Примеры:

```
SenderRestrictions = trust_protected_networks, reject
```

Позволяет принимать почту только с IP-адресов, указанных в **ProtectedNetworks**, а остальные IP-адреса блокирует;

```
SenderRestrictions = trust_protected_networks, trust_protected_domains, sleep 5,
add_score 10
```

Позволяет принять почту с IP-адресов, указанных в **ProtectedNetworks**, а также с доменов, указанных в **ProtectedDomains**. Для остальных почтовых сообщений перед продолжением он делает паузу в 5 секунд и увеличивает счет письма на 10 баллов.

Существует возможность собирать статистику по каждому из ограничений, чтобы определить число заблокированных им сообщений и, соответственно, определить его эффективность. Получить накопленную информацию можно, послав специальный сигнал процессу **drweb-receiver**, как описано в главе Обрабатываемые сигналы. Необходимость ведения статистики контролируется параметром **RestrictionStat** (указан выше, в пункте 1).

4. Параметры настройки проверки различных этапов SMTP-сессии

BlackNetworks = {Lookup} **WhiteNetworks** = {Lookup}	Черные и белые списки сетей, используемые в действиях trust_white_networks и reject_black_networks. Синтаксис данного параметра аналогичен синтаксису параметра **ProtectedNetworks** из секции [Maild]. Обратите внимание, что значение параметров – Lookup.

	Значение по умолчанию: **BlackNetworks** = **WhiteNetworks** =
DNSBLList = {LookupLite}	Список серверов DNSBL (предназначены для проверки IP-адреса соединения на наличие в списке спамеров). Данный список используется в действии reject_dnsbl. Сервера опрашиваются по очереди в том порядке, в котором они перечислены в значении параметра, до момента, пока какой-либо из них не ответит утвердительно, что IP-адрес – спамер, либо пока список серверов не закончится. Соединение с каждым сервером из списка предварительно тестируется отправкой ему тестового запроса, определенного спецификацией (127.0.0.2), на который сервер обязан отвечать положительно. Если сервер не ответил на данный запрос утвердительно, он считается недоступным, запись о чем фиксируется в журнале. Если ни один сервер из списка не удалось опросить, то IP-адрес соединения считается отсутствующим в списке серверов DNSBL, т.е. "чистым". Обратите внимание, что значение параметра – LookupLite. Значение по умолчанию: **DNSBLList** =
PositiveDNSBLCacheTimeout = {время}	Максимальный промежуток времени для кеширования положительных ответов от DNSBL-серверов. Значение по умолчанию: **PositiveDNSBLCacheTimeout** = 24h
NegativeDNSBLCacheTimeout = {время}	Максимальный промежуток времени для кеширования отрицательных ответов от DNSBL-серверов. Значение по умолчанию: **NegativeDNSBLCacheTimeout** = 10m
NegativeDNSCacheTimeout = {время}	Максимальный промежуток времени для кеширования отрицательных ответов от DNS-серверов. Значение параметра имеет смысл для всех ответов от DNS-серверов, кроме ответов от DNSBL-серверов. Значение по умолчанию: **NegativeDNSCacheTimeout** = 10m
BlackDomains = {Lookup} **WhiteDomains** = {Lookup}	Черные и белые списки доменов, используемые в действиях trust_white_domains и reject_black_domains. Синтаксис аналогичен синтаксису параметра **ProtectedDomains** из секции [Maild]. Обратите внимание, что значение параметров – Lookup. Значение по умолчанию: **BlackDomains** = **WhiteDomains** =
SpamTrap = {LookupLite}	Список адресов спам-ловушек. Данный список используется в действии reject_spam_trap. Обратите внимание, что значение параметра – LookupLite.

	Значение по умолчанию: **SpamTrap** =
ProtectedEmails = {Lookup}	Список защищаемых адресов. Используется в ограничении **reject_unknown_rcpts**. Позволяет отбрасывать письма для неверных получателей (тех, которые не принадлежат списку) и, при использовании фильтра anti_dha в Репутационном IP-фильтре, эффективно бороться с DHA-атаками. Рекомендуется задавать этот параметр вместе с **reject_unknown_rcpts** и использовать сместно с фильтром anti_dha. Обратите внимание, что значение параметра – Lookup. Значение по умолчанию: **ProtectedEmails** =
ProtectedSenderEmails = {Lookup}	Список доверенных адресов отправителей. Используется в ограничении **reject_unknown_sndrs**. Позволяет отбрасывать письма от неизвестных (неверных) отправителей и, при использовании фильтра anti_dha в Репутационном IP-фильтре, эффективно бороться с DHA-атаками. Рекомендуется задавать этот параметр вместе с **reject_unknown_sndrs** и использовать сместно с фильтром anti_dha. Обратите внимание, что значение параметра – Lookup. Значение по умолчанию: **ProtectedSenderEmails** =
ReputationIPFilter = {список фильтров}	Настройки использования Репутационного IP-фильтра. Позволяет выставлять счет IP-адресу на основе набираемой по данному адресу статистики и временно блокировать IP-адрес в случае, если его итоговый счет превышает некоторое пороговое значение. Доступны следующие фильтры: anti_dha, errors_filter, score_filter. Фильтры перечисляются через запятую и проверяются в порядке задания. Для каждого фильтра в начале указывается его название, затем перечисляются необязательные параметры, разделяемые пробелами. Значение по умолчанию: **ReputationIPFilter** =
MaxSessionScore = {числовое значение}	Пороговое значение для максимального общего счета для каждой сессии. Если общий счет сессии превысит указанное значение, то соединение закрывается с возвращением временной ошибки. Если значение установлено в 0, то данный параметр игнорируется. Значение по умолчанию: **MaxSessionScore** = 10000

Секция [Sender]

В секции `[Sender]` собраны настройки компонента **Sender**, отвечающего за отправку сообщений. Этой секции конфигурационного файла нет в дистрибутиве программного комплекса, предназначенного для работы с почтовой системой **CommuniGate Pro**.

UseSecureHash = {логический}	Предписание добавлять к письмам, отправляемым назад в почтовую систему, заголовка-«пометки» `X-DrWeb-Hash`. Используется, когда **Dr.Web MailD** вынужден при модификации писем выполнять процедуру отвергания старого письма с добавлением во входную очередь почтовой системы модифицированной версии письма. В этом случае письмо, поступившее на вход в **Dr.Web MailD** из почтовой системы и уже имеющее этот заголовок, сразу же, без проверки, отправляется на доставку, а в противном случае оно поступает на проверку. Прохождение повторных проверок может породить зацикливание письма и отказ от его доставки.
	Параметр должен использоваться только при работе с почтовыми системами **Postfix, Qmail, Sendmail** и **Zmailer:** • При работе **Dr.Web MailD** с почтовой системой **Postfix** значение `Yes` должно быть указано, только если взаимодействие с почтовой системой производится по протоколу `Milter` (используется модуль **drweb-milter**). В этом случае все сообщения, сформированные модулем **drweb-sender**, обрабатываются модулем **drweb-milter**. Значение по умолчанию: `No` • При работе **Dr.Web MailD** с почтовой системой **Qmail** или **Sendmail** значение `Yes` должно быть указано, если для получения и отправки сообщений используется одна и та же почтовая система. Значение по умолчанию: `Yes` • При работе **Dr.Web MailD** с почтовой системой **Zmailer** значение `Yes` должно быть указано, только если **drweb-zmailer** используется на стадии маршрутизации (например, запускается из `process.cf`). В этом случае все сообщения, сформированные модулем **drweb-sender**, обрабатываются модулем **drweb-zmailer**. Значение по умолчанию: `No`
SecureHash = {текст}	Параметр задает содержимое заголовка `X-DrWeb-Hash`. Значением параметра может быть произвольная строка, рекомендуемая длина строки – не менее 10 символов. Для повышения безопасности настоятельно рекомендуется изменить значение по умолчанию, указанное для данного параметра. При работе **Dr.Web MailD** с почтовой системой **Zmailer** значение параметра должно совпадать со значением параметра `--hash`, задаваемого при запуске модуля **drweb-zmailer** в случае, если эта почтовая система используется на этапе маршрутизации.
	Значение по умолчанию: **SecureHash** = `!!!---------___EDIT_THIS___!!!`
StalledProcessingInterval = {время}	Определяет периодичность, с которой компонент **Sender** проверяет базу данных сообщений на предмет "потерянных"

	писем с целью их доставки получателю. "Потерянные" письма – это сообщения, которые были получены и обработаны, но по каким-либо причинам не переданы компоненту **Sender** на отправку. Такая ситуация может случиться при возникновении проблем с сетью или питанием. При нахождении таких писем компонент **Sender** ставит их в очередь на отправку. Также этот тайм-аут используется для повторных запросов к источникам данных из Lookup, используемых в параметре **Router** (см. ниже), если источники данных оказались недоступны и письмо вследствие этого не может быть доставленным (не может быть получен целевой адрес доставки). Значение по умолчанию: **StalledProcessingInterval** = 10m
SendingIntervals = {время}	Промежутки времени между попытками отправить письма и уведомления, которые не удалось отправить с первой попытки (например, проблемы с сетью, ошибки на принимающей стороне и т.п.) При работе **Dr.Web для почтовых серверов UNIX** в синхронном режиме, **Sender** производит попытку отправки сразу после получения обработанного письма вне зависимости от установленого значения первого интервала. В случае неудачи **Sender** перейдет к отложенной отправке спустя **SendingIntervals**. Если в качестве первого значения параметра используется 0, он будет проигнорирован, поскольку попытка отправки письма уже была. Если **Dr.Web для почтовых серверов UNIX** работает в асинхронном режиме, попытка отправки письма всегда будет осуществляться согласно интервалам, заданым в значении данного параметра. Обратите внимание, что все генерируемые уведомления и DSN отправляются только в асинхронном режиме, вне зависимости от того, в каком режиме работает **Dr.Web MailD**. Поэтому отложенная отправка уведомлений и DSN будет всегда стартовать по времени с первого интервала, указанного в списке **SendingIntervals**. Поэтому желательно первым интервалом в списке всегда иметь 0s. Если первое значение в **SendingIntervals** не равно 0, то в асинхронном режиме будет иметься гарантированная задержка отправки обработанных сообщений и уведомлений, равная первому элементу списка. Если параметр имеет только значение 0s, попытки отложенной отправки осуществляться не будут, а письмо будет сразу перемещено в каталог /out/failed. Также см. замечание в конце раздела. Значение по умолчанию: **SendingIntervals** = 0s, 30s, 60s, 10m, 30m, 2h, 8h, 1d, 1d
Method = {SMTP | LMTP | pipe}	Метод, используемый компонентом **Sender** для доставки сообщения. • SMTP – сообщения отправляются по протоколу SMTP; • LMTP – сообщения отправляются по протоколу LMTP; • pipe – сообщения отправляются по программному каналу (pipe) внешней почтовой программе.

	Значение по умолчанию: зависит от дистрибутива.
MailerName = {SMTP | Sendmail | Postfix | CommuniGate | Qmail | Exim | Zmailer | Courier}	Имя почтовой системы, работающей совместно с **Dr.Web для почтовых серверов UNIX**. Данный параметр используется, если **Method** = pipe. Значение по умолчанию: зависит от дистрибутива.
Address = {адрес}	Адрес MTA, на который компонентом **Sender** будут отправляться сообщения после проверки. Если **Method** = pipe, то в данном параметре следует указать полный путь к внешней почтовой системе, получающей сообщения. При других значениях параметра **Method** в параметре **Address** задается сокет, через который отправляются сообщения. При работе программного комплекса в режиме **SMTP/LMTP-прокси** кроме стандартных типов адресов, можно также использовать тип mx:HOSTNAME, где HOSTNAME - имя хоста. В случае использования такого типа программный комплекс получает для HOSTNAME все MX-записи и отправляет сообщение в соответствии с ними. Если указать только префикс mx: без указания имени хоста, то программный комплекс получит и использует для отправки письма MX-записи домена получателя письма (из поля TO конверта письма). Можно указать несколько адресов для отправки сообщений. Значения разделяются запятой (","). Значение параметра не может быть пустым даже при использовании маршрутизации (см. параметр **Router**) . **Пример**: **Address** = inet:25@10.4.0.90, inet:25@10.4.0.91, inet:25@10.4.0.92 В данном примере, в случае если MTA, находящийся по адресу 10.4.0.90, перестанет отвечать, **Sender** предпримет попытку отправить письмо на адрес 10.4.0.91. В случае неудачной передачи, письмо будет передано на адрес 10.4.0.92. При большом количестве адресов рекомендуется увеличить значения параметров **MaxTimeoutForThreadActivity** и **IpcTimeout** в секции [General] до величины **не менее 5 минут**, чтобы **Sender** успел переключиться к последнему адресу в случае отсутствия ответа от предыдущих адресов. Значение по умолчанию: зависит от дистрибутива.
PipeTimeout = {время}	Максимальный промежуток времени на получение ответа при использовании pipe. Значение по умолчанию: **PipeTimeout** = 2m
Options = {текст}	Дополнительные параметры для метода pipe. Они передаются почтовой системе, которая получает сообщения.

	Значение по умолчанию: **Options** =
InPoolOptions = {настройки пула}	Настройки пула потоков для обработки перед очередью. Обратите внимание, что это параметр устарел и более не используется (изменение его значения не влияет на работу **Dr.Web MailD**)! Значение по умолчанию: **InPoolOptions** = auto
OutPoolOptions = {настройки пула}	Настройки пула потоков для обработки после очереди. Значение по умолчанию: **OutPoolOptions** = auto

Следующие параметры данной секции конфигурационного файла задаются только при работе программного комплекса с почтовыми системами **Exim** и **Postfix**, а также при работе в режиме **SMTP/LMTP-прокси**:

HeloCmdTimeout = {время}	Максимальный промежуток времени на выполнение команд HELO/EHLO Значение по умолчанию: **HeloCmdTimeout** = 5m
MailFromCmdTimeout = {время}	Максимальный промежуток времени на выполнение команды MAIL. Значение по умолчанию: **MailFromCmdTimeout** = 5m
RcptToCmdTimeout = {время}	Максимальный промежуток времени на выполнение команды RCPT. Значение по умолчанию: **RcptToCmdTimeout** = 5m
DataCmdTimeout = {время}	Максимальный промежуток времени на выполнение команд DATA/BDAT. Значение по умолчанию: **DataCmdTimeout** = 2m
DataBlockTimeout = {время}	Максимальный промежуток времени на отправку тела сообщения. Значение по умолчанию: **DataBlockTimeout** = 3m
EndOfDataTimeout = {время}	Максимальный промежуток времени на получение подтверждения о доставке сообщения. Значение по умолчанию: **EndOfDataTimeout** = 10m
OtherCmdsTimeout = {время}	Максимальный промежуток времени на выполнение остальных команд по протоколу SMTP/LMTP. Значение по умолчанию: **OtherCmdsTimeout** = 2m

SendDSN = {логический}	Разрешает или запрещает отправку DSN в случае возникновения проблем с доставкой письма. Пожалуйста, обратите внимание, что если **Dr.Web MailD** работает через **Sender** не в режиме **SMTP/LMTP-прокси**, а в режиме интеграции с MTA (**Exim**, **Postfix**, **Zmailer**), нужно быть осторожным при использовании DNS, поскольку в этом случае при задержке отправки письма DSN может быть сгенерирована как **Dr.Web MailD** (после всех неудачных попыток отправки), так и самим MTA (после истечения тайм-аута ожидания обработанного письма от **Dr.Web MailD**). В этом случае отправитель письма может получить при возникновении проблем сразу два DSN. DSN отправляется на доставку в компонент **Sender** в случае, когда невозможно передать его непосредственно в MTA (к примеру, если в адресе отправителя указано неполное доменное имя) и код возврата невозможно вернуть компоненту **Receiver**. Значение по умолчанию: **SendDSN** = No
Router = {Lookup}	Правила маршрутизации сообщений в зависимости от доменов, которым принадлежат получатели, используемые при работе **Dr.Web MailD** в режиме **SMTP/LMTP-прокси**. Сообщения, адресованные получателям, находящимся в разных доменах, могут быть отправлены на доставку по разным маршрутам. В данном параметре указывается, на какие адреса следует отправлять письма в различные домены. Значения параметра задаются в формате DOMAIN ADDRESS, где: • DOMAIN – подстрока, которая должна целиком входить в конверты получателей (конверт имеет вид <user@host>). Осуществляется поиск частичного соответствия, не чувствительный к регистру. Например, подстрока @localhost будет входить в конверты <test@LocalHost> и <yy@localhost.localdomain>, а подстрока @localhost> – только в конверт <test@LocalHost>. • ADDRESS – адреса, на которые будут отправляться сообщения, если подстрока DOMAIN будет целиком найдена в конверте. Формат ADDRESS аналогичен формату параметра **Address** в данного конфигурационного файла. Возможно указание нескольких адресов с разделением их символом "|", тогда письмо будет доставлено по первому адресу, с которым удалось установить соединение. Обратите внимание, что порядок указания адресов в списке существенен, поскольку ищется первое подходящее совпадение. Если для письма, подлежащего отправке, не нашлось соответствия DOMAIN в приведенном списке, для его отправки будет использоваться адрес, указанный в параметре **Address**. Поэтому значение параметра **Address** не должно быть пустым (см. замечание ниже, в подразделе **Использование Router**). Обратите внимание, что если источники данных, используемые в Lookup, оказались недоступны и письмо вследствие этого не может быть доставленным (не может быть получен целевой адрес отправки), такое письмо окажется

	"потерянным", и компонент **Sender** будет совершать повторные попытки его отправки через период **StalledProcessingInterval**.
	
	Пример:
	`Router = @main.server.com> mx:main.server.com| inet:25@backup.server.com`
	
	В этом случае письма, получатели которых принадлежат домену `main.server.com`, будут отправлены на адреса, указанные в MX-записи для `main.server.com`. Если доставить письма не удастся, то система попытается отправить письмо по адресу `backup.server.com` на порт `25`.
	
	Обратите внимание, что значение параметра – Lookup.
	
	Значение по умолчанию:
	`Router =`

Использование Router

Параметр **Router** позволяет использовать Lookup (за исключением типов `regex`, `wildcard` и `rfile`).

Пример:

```
Router = "mysql:select address from senders where user='$u'"
```

С помощью этого запроса проверяется, присутствует ли локальная часть адреса получателя в базе данных **MySQL** в столбце `user` таблицы `senders`. Если присутствует, то письмо высылается на адрес, указанный в найденной строке в столбце `address`.

Пример:

```
Router = "ldap:///description?sub?(cn='$d')", domain1.com inet:25@example.com | inet:1025@example.com | inet:2025@example.com, mail.com mx: | inet:25@mail.backup, domain2.com mx:mail.ru | inet:25@mail.backup, "file:/path/to/routers.list"
```

В данном случае письма будут отправляться на доставку следующим образом:

1) Сначала будет производиться поиск доменного имени отправителя через LDAP (используется атрибут `cn`, в случае обнаружения параметры перенаправления берутся из поля `description`).
2) Письма, получатели которых имеют домен `domain1.com`, будут отправлены по адресу `example.com` на порт `25`. В случае неудачи, будет предпринята попытка передать письма на тот же адрес на порт `1025`, а затем на порт `2025`.
3) Письма, отправленные адресованные домену `mail.com`, будут пересылаться на адреса, соответствующие MX-записи `mail.com`.
4) Письма, адресованные домену `domain2.com`, будут перенаправляться на адреса, указанные для MX-записи `mail.ru`, либо на порт `25` сервера `mail.backup`.
5) Если адрес отправителя не совпал ни с одним из описанных ранее в правилах, соответствие будет проверено в файле `/path/to/routers.list`.

Обратите внимание, что каждому конкретному домену в соответствие ставится один адрес, поэтому конструкции подобного вида недопустимы:

```
Router = domain, domain2 25@host
```

Перед использованием `Lookup` в параметре **Router** рекомендуется проверить его корректность (используйте утилиту **drweb-lookup**), а также убедиться в доступности используемого источника данных. Если в качестве значения параметра **Router** используется `Lookup` с режимом обработки ошибок **OnError**=`exception` (определенном либо в настройках используемого источника данных или переопределенном локально, в значении `Lookup`), то, в случае если будет

невозможно получить нужный маршрут из источника данных, эта ситуация будет обработана как ошибка в компоненте **Sender**, запись о чем будет зафиксирована в журнале. При этом в синхронном режиме компонент **Receiver** всегда возвращает отправителю письма код SMTP **451** (`Requested action aborted: local error in processing`), а обработанное письмо будет удалено из всех очередей. В асинхронном режиме письмо будет помечено как "потерянное" и **Sender** будет пытаться его отправлять с периодичностью, указанной в параметре **StalledProcessingInterval**.

В случае если письмо не удалось отослать ни на один из найденных адресов, то в зависимости от кода ответа SMTP, который вернет последний MTA:

- **Sender** перейдет к отложенной отправке спустя интервал, указанный в значении параметра **SendingIntervals**. При отложенной отправке также действуют правила определения целевого MTA, указанные в параметре **Router**. Если отложенная отправка закончится неудачей, то (если это разрешено в параметре **SendDSN**) компонент **Notifier** сгенерирует DSN. Если для домена отправителя, указанного в конверте недоставленного письма, не заданы специальные маршруты (в параметре **Router** или в Правилах обработки почты), то данный DSN будет отправлен на адрес, указанный в значении параметра **Address**.
- если последний MTA ответит кодом ошибки SMTP **5****, то DSN будет сгенерирована сразу, а письмо будет удалено из out-очередей. Схема отправки DSN в этом случае аналогична вышеописанному. Если DSN не может быть доставлен, то он будет удален спустя интервал, указанный в значении параметра **SendingIntervals**.

При определении разных маршрутов для одного и того же домена в Правилах обработки почты и в параметре **Router**, будут действовать только маршруты, определенные в Правилах обработки почты.

Даже при настройке маршрутизации всей почты с помощью параметра **Router** значение параметра **Address** не должно быть пустым, поскольку в противном случае при старте компонет **Sender** выведет сообщение об ошибку и завершит свою работу. Это связано с тем, что адрес, указанный в значении параметра **Address**, используется, если для получателя не будет найдено соответствия в таблице маршрутизации или в Правилах обработки почты.

После того, как компонент **Sender** исчерпает все попытки отложенной отправки письма, "потерянное" письмо остается в подкаталоге `/out/failed` каталога хранилища писем навечно, и в дальнейшем его требуется либо явно отправить при помощи утилиты **drweb-inject**, либо удалить средствами ОС.

Секция [Courier]

В секции `[Courier]` сосредоточены настройки для взаимодействия с почтовой системой **Courier**. Эта секция присутствует в конфигурационном файле только той версии программного комплекса, которая рассчитана на работу с вышеуказанной почтовой системой.

ProcessingTimeout = {время}	Максимальное время ожидания модулем **drweb-courier** окончания сканирования сообщения. Рекомендуется, чтобы значение этого параметра было больше, чем значение параметра **SendTimeout** в секции настроек `[MailBase]`. Значение по умолчанию: **ProcessingTimeout** = 40s
ProcessingErrors = {действие}	Действие, применяемое к сообщениям, вызвавшим ошибки сканирования.

	Может быть задано только одно из основных действий: `tempfail, discard, pass, reject` Значение по умолчанию: **`ProcessingErrors`** = `reject`
`MainPoolOptions` = `{настройки пула}`	Настройки пула потоков, обрабатывающих запросы. Значение по умолчанию: **`MainPoolOptions`** = `auto`
`ReplyPoolOptions` = `{настройки пула}`	Настройки пула потоков, обрабатывающих ответы от модуля **`drweb-maild`**. Значение по умолчанию: **`ReplyPoolOptions`** = `auto`
`BaseDir` = `{путь к каталогу}`	Каталог почтовой системы **Courier**. Значение по умолчанию: **`BaseDir`** = `/usr/lib/courier`
`SocketDirs` = `{список строк}`	Список путей, используемых для создания UNIX-сокетов при взаимодействии с почтовой системой **Courier**. Сокет создается в первом каталоге списка, а остальные каталоги проверяются на наличие UNIX-сокетов, чьи имена совпадают с названием модуля **`drweb-courier`**. При нахождении такие UNIX-сокеты удаляются. В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала `HUP`. Необходимо перезапустить **Dr.Web MailD**. Значение по умолчанию: **`SocketDirs`** = `/var/lib/courier/allfilters, /var/lib/courier/filters`
`SocketAccess` = `{права}`	Права на файлы UNIX-сокетов для взаимодействия программного комплекса и почтовой системы **Courier**. В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала `HUP`. Необходимо перезапустить **Dr.Web MailD**. Значение по умолчанию: **`SocketAccess`** = `0660`

Секция [CgpReceiver]

В секции `[CgpReceiver]` сосредоточены настройки компонента **Receiver** для взаимодействия с почтовой системой **CommuniGate Pro**. Эта секция присутствует в конфигурационном файле только той версии программного комплекса, которая рассчитана на работу с вышеуказанной почтовой системой.

`ProcessingTimeout` = `{время}`	Максимальное время ожидания компонентом **Receiver** окончания сканирования сообщения. Рекомендуется, чтобы значение этого параметра было больше, чем значение параметра **`SendTimeout`** секции настроек `[MailBase]`. Значение по умолчанию: **`ProcessingTimeout`** = `40s`

PoolOptions = {настройки пула}	Настройки пула потоков компонента **Receiver**. Значение по умолчанию: **PoolOptions** = auto
ProcessingErrors = {действие}	Действие, применяемое к сообщениям, вызвавшим ошибки сканирования. Может быть задано только одно из основных действий: tempfail, discard, pass, reject Значение по умолчанию: **ProcessingErrors** = reject
ChownToUser = {строка}	Установка владельца на файл с сообщением, полученным от почтовой системы **CommuniGate Pro**. Поскольку модуль **drweb-cgp-receiver** работает с правами администратора (root), необходимо либо оставить данный параметр пустым и запускать весь комплекс **Dr.Web для почтовых серверов UNIX** с правами администратора, либо установить значением данного параметра имя определенного пользователя, с правами которого работает программный комплекс (drweb по умолчанию). Значение по умолчанию: **ChownToUser** = drweb

Секция [CgpSender]

В секции [CgpSender] сосредоточены настройки компонента **Sender** для взаимодействия с почтовой системой **CommuniGate Pro**. Эта секция присутствует в конфигурационном файле только той версии программного комплекса, которая рассчитана на работу с вышеуказанной почтовой системой.

UseSecureHash = {логический}	Предписание добавлять к письмам, отправляемым назад в почтовую систему, заголовка-«пометки» X-DrWeb-Hash. Используется, когда **Dr.Web MailD** вынужден при модификации писем выполнять процедуру отвергания старого письма с добавлением во входную очередь почтовой системы модифицированной версии письма. Если указано значение No, письмо, поступившее на вход в **Dr.Web MailD** из почтовой системы, сразу же, без проверки, отправляется на доставку, если оно было помещено во входную очередь почтовой системы локально (через PIPE). Если указано значение Yes, письмо, поступившее на вход в **Dr.Web MailD** из почтовой системы и уже имеющее этот заголовок, сразу же, без проверки, отправляется на доставку, если оно было помещено во входную очередь почтовой системы локально (через PIPE). Обратите внимание, что поскольку этот параметр используется не только **Sender**, но и **Receiver**, то после изменения значения этого параметра не достаточно отправить сигнал HUP компоненту **Dr.Web Monitor** (это заставит перечитать конфигурацию компонент **Sender**). Необходимо также перезапустить почтовую систему **CGP,** поскольку именно она запускает компонент **Receiver**, и только ее перезапуск заставит компонент **Receiver** перечитать измененное значение этого параметра конфигурации.

	Значение по умолчанию: **UseSecureHash** = No
SecureHash = {строка}	Параметр задает содержимое заголовка X-DrWeb-Hash. Значением параметра может быть произвольная строка, рекомендуемая длина строки – не менее 10 символов. Для повышения безопасности настоятельно рекомендуется изменить значение по умолчанию, указанное для данного параметра. Значение по умолчанию: **SecureHash** = !!!---------___EDIT_THIS___!!!
PoolOptions = {настройки пула}	Настройки пула потоков компонента **Sender** для работы с **CGP**. Значение по умолчанию: **PoolOptions** = auto
SubmitDir = {путь к каталогу}	Каталог, в который модуль **drweb-cgp-sender** сохраняет сообщения для их последующей отправки посредством почтовой системы **CommuniGate Pro**. Значение по умолчанию: **SubmitDir** = /var/CommuniGate/Submitted
SubmitFilesMode = {права}	Права на файлы, хранящие создаваемые уведомления или сохраненные сообщения. Значение по умолчанию: **SubmitFilesMode** = 0600
SubmitFilenamesPrefix = {строка}	Префикс для имен файлов сохраненных сообщений. Формат имени файла: %{SubmitDir}/%{SubmitFilenamesPrefix}XXXXXX Возможно использование макроса %s, который будет заменен на идентификатор сообщения, присвоенный письму почтовой системой **CommuniGate Pro** и полученный из имени файла. Использование данного макроса позволяет облегчить анализ файлов журналов. Значение по умолчанию: **SubmitFilenamesPrefix** = drweb_submit_%s_
SubmitFilenamesMode = {std | tai | rand48}	Способ именования файлов сохраненных сообщений: • std – переименование с использованием команды mkstemp. Используется шаблон имени drweb_submit_XXXXXX; • tai – переименование согласно TAI (международное атомное время). Используется шаблон имени %sec.%usec.drweb_submit_XXXXXX; • rand48 – переименование с использованием команды lrand48. Используется шаблон имени drweb_submit_XXXXXX. Значение по умолчанию: **SubmitFilenamesMode** = std

Секция [Milter]

Параметры секции `[Milter]` управляют работой модуля **drweb-milter**, отвечающего за взаимодействие программного комплекса с почтовой системой **Postfix** или почтовой системой **Sendmail** по протоколу `Milter`. Эта секция присутствует в конфигурационных файлах только тех версий программного комплекса, которые рассчитаны на работу с вышеуказанными почтовыми системами.

Address = {адрес}	Адрес соединения по протоколу `Milter`, соответствующий определению, заданному в настройках почтовой системы (в конфигурационном файле `sendmail.cf` для почтовой системы **Sendmail** и в конфигурационном файле `main.cf` – для **Postfix**).
	В качестве адреса нельзя использовать путь к PID-файлу.
	Пример:
	`Address = local:%var_dir/ipc/drweb-milter.skt`
	В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала `HUP`. Необходимо перезапустить **Dr.Web MailD**.
	Значение по умолчанию:
	`Address = inet:3001@127.0.0.1`

Timeout = {время}	Максимальное время ожидания соединения по протоколу `Milter`. Данное значение должно быть больше, чем значение любого параметра **Timeout** в конфигурационном файле используемой почтовой системы. Значение по умолчанию: `Timeout = 2h`
PendedConnections = {числовое значение}	Максимальная длина очереди на соединение с почтовой системой (**drweb-milter** ожидает окончания обработки сообщений от почтовой системы). Значение по умолчанию: `PendedConnections = 64`
CanChangeBody = {логический}	Возможность изменения тела сообщения, полученного на проверку от почтовой системы. Почтовая система **Postfix** данную функцию поддерживает начиная с версии `2.4`. В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала `HUP`. Необходимо перезапустить **Dr.Web MailD**. Если этот параметр имеет значение `Yes`, то возврат проверенного сообщения обратно в очередь доставки MTA осуществляется через модуль **drweb-milter** (**Receiver**) вне зависимости от того, в каких очередях (**after-queue** или **before-queue**) расположены подключаемые модули, проверяющие письмо. В противном случае (если этот параметр имеет значение `No`) возврат проверенного сообщения обратно в очередь доставки MTA осуществляется через модуль **drweb-sender** (**Sender**), если тело письма было модифицировано при проверке (например, из него мог быть удален вирус), т.к. в этом случае его уже нельзя вернуть назад в почередь почтовой системы,

	поэтому оно передается в MTA как новое поступившее. Если тело письма не было модифицировано, то оно будет возвращено в очередь доставки MTA через модуль **drweb-milter** (**Receiver**). Определение сопосба возврата в данном случае также не зависит от того, в каких очередях расположены подключаемые модули, проверяющие письмо. Обратите внимание, что добавление или изменение заголовков письма, но не его тела, модификацией не считается. Все служебные уведомления (включая DSN), отчеты, перенаправленные (по действию `redirect`) и клонированные письма отправляются только через модуль **drweb-sender** (**Sender**), вне зависимости от значения параметра **CanChangeBody** и расстановки подключаемых модулей по очередям. Дополнительную информацию см. в разделе Обработка сообщений. Значение по умолчанию: **CanChangeBody** = Yes
ProcessingTimeout = {время}	Максимальное время ожидания модулем **drweb-milter** окончания сканирования сообщения компонентами **Dr.Web MailD**. Рекомендуется, чтобы значение этого параметра было больше, чем значение параметра **SendTimeout** в секции [MailBase]. Обратите внимание, что в синхронном режиме учитывается также значение параметра **IPCTimeout** в секции [General]. При ожидании ответа от **Dr.Web MailD** выбирается большее из значений параметров **ProcessingTimeout** и **IPCTimeout**. Если в течение выбранного максимального таймаута **Dr.Web MailD** не успеет вернуть ответ модулю **drweb-milter**, то будет выполнено действие, указанное в параметре **ProcessingErrors** (см. ниже), а в журнале **Dr.Web MailD** будут зафикированы ошибки типа "broken pipe" Значение по умолчанию: **ProcessingTimeout** = 40s
ProcessingErrors = {действие}	Действие, применяемое к сообщениям, вызвавшим ошибки сканирования. Может быть задано только одно из основных действий: tempfail, discard, pass, reject Значение по умолчанию: **ProcessingErrors** = reject
MinPersistConnection = {числовое значение}	Минимальное количество соединений с модулем **drweb-maild**. В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала HUP. Необходимо перезапустить **Dr.Web MailD**. Значение по умолчанию: **MinPersistConnection** = 2
UseStat = {логический}	Статистика по соединениям с модулем **drweb-maild**. Статистика записывается в файл при получении процессом

	drweb-milter сигнала SIGUSR1. Значение по умолчанию: **UseStat** = No
MaxFreetime = {время}	Максимальное время бездействия, после которого соединения с модулем **drweb-maild** закрываются. Значение по умолчанию: **MaxFreetime** = 2m
ReplyPoolOptions = {настройки пула}	Настройки пула потоков, обрабатывающих ответы от модуля **drweb-maild**. Значение по умолчанию: **ReplyPoolOptions** = auto

Секция [Qmail]

В секции [Qmail] находятся настройки для взаимодействия с почтовой системой **Qmail**. Эта секция присутствует в конфигурационном файле только той версии программного комплекса, которая рассчитана на работу с вышеуказанной почтовой системой.

ProcessingTimeout = {время}	Максимальное время ожидания модулем **drweb-qmail** окончания сканирования сообщения. Рекомендуется, чтобы значение этого параметра было больше, чем значение параметра **SendTimeout** в секции настроек [MailBase]. Значение по умолчанию: **ProcessingTimeout** = 40s
ReadingTimeout = {время}	Максимальное время ожидания получения всех заголовков и тела сообщения от модуля **qmail-queue**. Значение по умолчанию: **ReadingTimeout** = 20m
ProcessingErrors = {действие}	Действие, применяемое к сообщениям, вызвавшим ошибки сканирования. Может быть задано только одно из основных действий: tempfail, discard, pass, reject Значение по умолчанию: **ProcessingErrors** = reject
MainPoolOptions = {настройки пула}	Настройки пула потоков, обрабатывающих запросы. Значение по умолчанию: **MainPoolOptions** = auto
ReplyPoolOptions = {настройки пула}	Настройки пула потоков, обрабатывающих ответы модуля **drweb-maild**. Значение по умолчанию: **ReplyPoolOptions** = auto
ListenUnixSockets = {адрес сокета}	Список UNIX-сокетов, которые использует модуль **drweb-qmail** для получения запросов на сканирование сообщений от модуля **qmail-queue**.

	Сокеты, присутствующие в этом списке, должны быть также указаны в списке файлов, за которыми следит модуль **qmail-queue**. Этот список можно посмотреть командой **qmail-queue** `--help`. Значение по умолчанию: **ListenUnixSockets** = `local:%var_dir/ipc/.qmail`
QmailQueue = `{путь к файлу}`	Путь к оригинальному исходному файлу **qmail-queue**. Значение по умолчанию: **QmailQueue** = `/var/qmail/bin/qmail-queue.original`

Секция [IMAP]

В этой секции находятся настройки модуля **drweb-imap** (компонент **IMAP filter**, используемый для перехвата писем по протоколу IMAP при работе с почтовыми клиентами):

ServerAddress = `{адрес}`	Адрес, по которому следует подключаться к серверу IMAP. Значение по умолчанию: **ServerAddress** = `inet:imap@127.0.0.1`
ListenAddress = `{список адресов}`	Список адресов сокетов, на которых следует ожидать подключений клиентов. Допустимы адреса вида `inet:` или `inet-ssl:` (если вы используете TLS/SSL шифрование). Последний требует от фильтра задействовать протокол IMAPS. Значение по умолчанию: **ListenAddress** = `inet:5200@0.0.0.0`
ServerTLSSettings = `{настройки TLS/SSL}`	Настройки TLS/SSL, используемые для подключений в качестве сервера. Подключения в качестве TLS/SSL сервера возможны, только если заданы сертификат (`certificate`) и закрытый ключ (`private_key_file`), а адрес для подключения указан с типом сокета `inet-ssl`. **Пример**: **ServerTLSSettings** = `use_sslv2 no, private_key_file /path/to/pkey, certificate / path/to/certificate` Обратите внимание, что пользователь, с правами которого работает IMAP-фильтр (обычно `drweb`), должен иметь права на чтение файла сертификата. Кэширование SSL-сессий в текущей версии не поддерживается. Значение по умолчанию: **ServerTLSSettings** =
ClientTLSSettings = `{настройки TLS/SSL}`	Настройки TLS/SSL, используемые для подключений в качестве клиента. **Пример**: **ClientTLSSettings** = `use_sslv2 no, private_key_file /path/to/pkey, certificate / path/to/certificate` Обратите внимание, что пользователь, с правами которого работает IMAP-фильтр (обычно `drweb`), должен иметь права

	на чтение файла сертификата. Кэширование SSL-сессий в текущей версии не поддерживается. Значение по умолчанию: **ClientTLSSettings** =
IoTimeout = {время}	Максимальное время ожидания для любых операций ввода и вывода с сокетом клиента для уже начавшейся операции. Значение по умолчанию: **IoTimeout** = 60s
ProcessingTimeout = {время}	Максимально допустимое время обработки письма модулем **drweb-maild**. Значение по умолчанию: **ProcessingTimeout** = 60s
MinFilterToMaildConnections = {числовое значение}	Минимальное число соединений между фильтром IMAP и **drweb-maild**. Значение по умолчанию: **MinFilterToMaildConnections** = 2
MaxFilterToMaildConnections = {числовое значение}	Максимальное число соединений между фильтром IMAP и модулем **drweb-maild**. При значении 0 количество соединений не ограничено. Значение по умолчанию: **MaxFilterToMaildConnections** = 0
FilterToMaildKeepAliveTime = {время}	Максимальное время удержания неактивных соединений между фильтром IMAP и **drweb-maild** сверх минимального количества соединений. Для обращения к **drweb-maild** фильтр поддерживает несколько соединений с ним, каждое из которых может обслуживать одну операцию. Если свободных соединений нет, создаются новые, пока их число не достигнет порогового значения, указанного в параметре **MaxFilterToMaildConnection**. При простое свободных соединений в течение времени, заданного в параметре **FilterToMaildKeepAliveTime**, они закрываются, но общее их число не снижается ниже значения **MinFilterToMaildConnections**. Значение по умолчанию: **FilterToMaildKeepAliveTime** = 60s
CallbackPoolOptions = {настройки пула}	Настройки дополнительного пула потоков, обрабатывающих сигналы от **drweb-maild** об окончании обработки письма. Значение по умолчанию: **CallbackPoolOptions** = auto
PoolOptions = {настройки пула}	Настройки основного пула потоков, обрабатывающих подключения клиентов. На каждое подключение требуется новый поток, иначе некоторые клиенты будут ожидать появления потока неподключенными.

	Значение по умолчанию: **PoolOptions** = auto
MaxConnections = {числовое значение}	Максимальное количество входящих соединений. Если указано значение 0, то количество входящих соединений не ограничено. Значение по умолчанию: **MaxConnections** = 0
MaxConnectionsPerIp = {числовое значение}	Максимальное количество одновременных соединений, разрешенных для одного IP-адреса (клиента). Если указано значение 0, то количество входящих соединений для одного IP-адреса не ограничено. Значение по умолчанию: **MaxConnectionsPerIp** = 0
DisablePlainText = {логический}	Запретить клиенту передачу имени и пароля в незашифрованном виде. Требует предварительной настройки **OpenSSL**. Значение по умолчанию: **DisablePlainText** = no
DoS_Blackhole = {логический}	Обрывать соединение, если с одного IP приходит слишком много запросов на подключение, не возвращая клиенту сообщение о причине ошибки. Значение по умолчанию: **DoS_Blackhole** = no
MaxCommandLength = {размер}	Максимальный размер команды для протокола IMAP. Команда - это строка, которую посылает клиент серверу. Максимальный размер команды, которую клиент может послать - около 1000 байт согласно действующему RFC. Обратите внимание, что если значение параметра установить равным нулю или очень маленьким (до 10 байт), то команды клиентов не будут восприниматься. Значение по умолчанию: **MaxCommandLength** = 1000b
MaxCachedHeadersPerMail = {размер}	Максимальное количество памяти, которое можно выделять для сохранения часто используемых заголовков. Фильтр IMAP кэширует основные заголовки сообщений в оперативной памяти для ускорения доступа к ним. Если указано значение 0, то размер выделяемой для сохранения заголовков памяти не регулируется (ограничен лишь размером доступной памяти). Значение по умолчанию: **MaxCachedHeadersPerMail** = 64k
MaxLettersPerUser = {числовое значение}	Максимальное количество писем, которые следует кэшировать в течение одной сессии. Фильтр IMAP содержит кэш проверенных писем, поскольку протокол IMAP позволяет клиенту совершать множество частичных запросов к одному сообщению.

	В большинстве случаев запросы идут последовательно, но если пользователь будет обращаться к нескольким записям, то потребуется кэшировать более одного сообщения. Если значение данного параметра установить в `0` (что настоятельно не рекомендуется), то это будет означать отсутсвие ограничений по количеству кэшируемых писем. Значение по умолчанию: **`MaxLettersPerUser`** = `6`
`MaxDiskPerUSer` = `{размер}`	Максимальный размер места на диске, отведенного под кэшированные письма. Значение по умолчанию: **`MaxDiskPerUSer`** = `10m`
`OnFilterErrors` = `{список действий}`	Действие, применяемое к письму при ошибке, возникшей до отправки письма модулю **`drweb-maild`**. Возможные значения: `reject` или `pass`. Значение по умолчанию: **`OnFilterErrors`** = `reject`

Секция [POP3]

В этой секции находятся настройки модуля **`drweb-pop3`** (компонент **POP3 filter**, используемый для перехвата писем по протоколу POP3 при работе с почтовыми клиентами):

`ServerAddress` = `{адрес}`	Адрес, по которому следует подключаться к серверу POP3. Значение по умолчанию: **`ServerAddress`** = `inet:pop3@localhost`
`ListenAddress` = `{список адресов}`	Список адресов сокетов, на которых следует ожидать подключений клиентов. Допустимы адреса вида `inet:` или `inet-ssl:` (если вы используете TLS/SSL шифрование). Последний требует от фильтра задействовать протокол POP3S. Значение по умолчанию: **`ListenAddress`** = `inet:5110@localhost`
`ServerTLSSettings` = `{настройки TLS/SSL}`	Настройки TLS/SSL, используемые для подключений в качестве сервера. Настройки задаются через запятую. Подключения в качестве TLS/SSL сервера возможны, только если заданы сертификат (`certificate`) и закрытый ключ (`private_key_file`), а адрес для подключения указан с типом сокета `inet-ssl`. **Пример**: **`ServerTLSSettings`** = `use_sslv2 no, private_key_file /path/to/pkey, certificate /path/to/certificate` Обратите внимание, что пользователь, с правами которого работает POP3-фильтр (обычно `drweb`), должен иметь права на чтение файла сертификата. Значение по умолчанию: **`ServerTLSSettings`** =

ClientTLSSettings = {настройки TLS/SSL}	Настройки TLS/SSL, используемые для подключений в качестве клиента. **Пример**: **ClientTLSSettings** = use_sslv2 no, private_key_file /path/to/pkey, certificate / path/to/certificate Обратите внимание, что пользователь, с правами которого работает POP3-фильтр (обычно drweb), должен иметь права на чтение файла сертификата. Значение по умолчанию: **ClientTLSSettings** =
IoTimeout = {время}	Максимальное время ожидания для любых операций ввода и вывода с сокетом клиента для уже начавшейся операции. Значение по умолчанию: **IoTimeout** = 60s
ProcessingTimeout = {время}	Максимально допустимое время обработки письма модулем **drweb-maild**. Значение по умолчанию: **ProcessingTimeout** = 60s
MinFilterToMaildConnections = {числовое значение}	Минимальное число соединений между фильтром POP3 и **drweb-maild**. Значение по умолчанию: **MinFilterToMaildConnections** = 2
MaxFilterToMaildConnections = {числовое значение}	Максимальное количество соединений между фильтром POP3 и модулем **drweb-maild**. При значении 0 количество соединений не ограничено. Значение по умолчанию: **MaxFilterToMaildConnections** = 0
FilterToMaildKeepAliveTime = {время}	Максимальное время удержания неактивных соединений между фильтром POP3 и **drweb-maild** сверх минимального количества соединений. Для обращения к **drweb-maild** фильтр поддерживает несколько соединений с ним, каждое из которых может обслуживать одну операцию. Если свободных соединений нет, создаются новые, пока их число не достигнет порогового значения, указанного в параметре **MaxFilterToMaildConnection**. При простое свободных соединений в течение времени, заданного в параметре **FilterToMaildKeepAliveTime**, они закрываются, но общее их число не снижается ниже значения **MinFilterToMaildConnections**. Значение по умолчанию: **FilterToMaildKeepAliveTime** = 30s
PoolOptions = {настройки пула}	Настройки основного пула потоков, обрабатывающих подключения клиентов. На каждое подключение требуется новый поток, иначе некоторые клиенты будут ожидать появления потока неподключенными.

	Значение по умолчанию: **PoolOptions** = auto
CallbackPoolOptions = {настройки пула}	Настройки дополнительного пула потоков, обрабатывающих сигналы от **drweb-maild** об окончании обработки письма. Значение по умолчанию: **CallbackPoolOptions** = auto
MaxConnections = {числовое значение}	Максимальное количество входящих соединений. Если указано значение 0, то количество входящих соединений не ограничено. Значение по умолчанию: **MaxConnections** = 0
DoS_Blackhole = {логический}	Обрывать соединение, если с одного IP-адреса приходит слишком много запросов на подключение, не возвращая клиенту сообщение о причине ошибки. Значение по умолчанию: **DoS_Blackhole** = no
DisablePlainText = {логический}	Запретить клиенту передачу имени и пароля в незашифрованном виде. Требует предварительной настройки **OpenSSL**. Значение по умолчанию: **DisablePlainText** = No
MaxConnectionsPerIp = {числовое значение}	Ограничение на общее количество одновременных подключений с одного адреса. Если указано значение 0, то ограничений нет. Значение по умолчанию: **MaxConnectionsPerIp** = 0
MaxCommandLength = {размер}	Максимальный размер команды для протокола POP3. Команда - это строка, которую посылает клиент серверу. Максимальный размер команды, которую клиент может послать - около 1000 байт согласно действующему RFC. Обратите внимание, что если значение параметра установить равным нулю или очень маленьким (до 10 байт), то команды клиентов не будут восприниматься. Значение по умолчанию: **MaxCommandLength** = 1000b
OnFilterErrors = {действие}	Действие, применяемое к письму при ошибке, возникшей до отправки письма модулю **drweb-maild**. Возможные значения: reject или pass. Значение по умолчанию: **OnFilterErrors** = reject

Источники данных

В данном разделе описаны секции конфигурационного файла, содержащие параметры подключения к источникам данных, используемых в Lookup и Storage для параметров **Dr.Web**

для почтовых серверов UNIX которые могут в качестве своих значений использовать данные, извлекаемые из хранилищ данных (базы данных, текстовые файлы регулярной структуры, LDAP).

Обратите внимание, что данные секции содержат только общие параметры, используемые в Lookup и Storage по умолчанию. Некоторые параметры (отмеченные особо) могут переопределяться в каждом конкретном `Lookup` и `Storage`.

В файле могут присутствовать любые секции подключения к источникам данных. Реальное использование источников данных определяется только ссылками на них в Lookup и Storage. Секции, задающие параметры подключения к источникам данных, которые не используются, даже если они присутствуют в файле, не влияют на работу **Dr.Web для почтовых серверов UNIX**.

Секция [LDAP]

В секции `[LDAP]` собраны настройки взаимодействия **Dr.Web MailD** с сервером LDAP:

Lib = {путь к файлу}	Путь к библиотеке `OpenLDAP` версии 2.0 или выше. Библиотека должна быть собрана с поддержкой потоков (т.е. иметь в имени файла суффикс `_r`). Поиск библиотеки осуществляется в соответствии с правилами системного вызова `dlopen` (см. документацию по `dlopen`). В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала `HUP`. Необходимо перезапустить **Dr.Web MailD**. Значение по умолчанию: **Lib** = /usr/lib/libldap_r.so

Обратите внимание, что в операционной системе **FreeBSD** версии 6.4 для amd64 при использовании библиотеки `libldap_r.so` возможно возникновение следующей ошибки: `Undefined symbol "gethostbyname_r"`

Hostname = {текст}	Имя хоста, на котором работает сервер LDAP. Если значение параметра не указано, используется `localhost`. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **Hostname** =
Port = {числовое значение}	Порт для подключения к серверу LDAP. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **Port** = 389
Timeout = {время}	Максимальное время выполнения LDAP-запросов. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **Timeout** = 10s
Version = {текст}	Версия протокола LDAP. Для обеспечения защищенной передачи данных с помощью

	TLS/SSL должен использоваться протокол LDAP не ниже версии 3. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **Version** = 3
Bind = {логический}	Необходимость привязки перед выполнением запросов. Для протокола LDAP версии 3 привязка является необязательной. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **Bind** = No
BindDn = {текст}	Уникальное имя при выполнении привязки. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **BindDn** =
BindPw = {текст}	Пароль, используемый при выполнении привязки. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **BindPw** =
SearchBase = {текст}	Базовый DN, с которого будет начинаться поиск (RFC2253). Значение по умолчанию: **SearchBase** =
SizeLimit = {числовое значение}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. При значении 0 ограничения отсутствуют. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **SizeLimit** = 0
Dereference = {3 | 2 | 1 | 0}	Разрешение LDAP-псевдонимов. • 0 – никогда; • 1 – при поиске; • 2 – при определении базового объекта для поиска; • 3 – всегда. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **Dereference** = 0
ChaseReferrals = {числовое значение}	Настройка LDAP_OPT_REFERRALS. Для установки данного параметра необходим протокол LDAP

	версии не ниже 3. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **ChaseReferrals** = 0
SkipDomains = {LookupLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность. Обратите внимание, что значение параметра – LookupLite. Значение параметра может быть указано в локальных настройках Lookup Значение по умолчанию: **SkipDomains** =
OnError = {ignore | exception}	Задает процедуру обработки ошибок, возникающих в обработке Lookup при обращении к указанному источнику данных. • ignore – следует игнорировать возникшую ошибку и продолжить обработку письма (происходит только регистрация ошибки в журнале); • exception – следует сгенерировать исключение, которое будет обрабатываться как ошибка обработки письма, в соответствии со значением параметра **ProcessingError**, заданного в настройках того компонента, при работе которого эта ошибка возникла. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **OnError** = ignore
CheckPeriod = {время}	Максимальный период бездействия LDAP-соединения, после которого оно будет закрыто. Проверка на неактивные соединения осуществляется с использованием этого же промежутка времени. Значение по умолчанию: **CheckPeriod** = 2m

Dr.Web MailD работает с LDAP через библиотеку **OpenLDAP** (должна быть не ниже версии 2.0).

В случае если указанный LDAP-сервер недоступен, попытка установления подключения будет производиться в течение таймаута, заданного параметром **Timeout**. По его истечении будет зафиксирована ошибка, которая будет обработана в соответствии со значением параметра **OnError**.

Секция [Oracle]

В секции [Oracle] собраны настройки взаимодействия **Dr.Web MailD** с СУБД **Oracle**:

Lib = {путь к файлу}	Путь к библиотеке, поддерживающей **Oracle OTL** версии 8 или выше. Библиотека должна быть собрана с поддержкой потоков. Поиск библиотеки осуществляется в соответствии с правилами системного вызова dlopen (см. документацию по dlopen).

	В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала `HUP`. Необходимо перезапустить **Dr.Web MailD**. Значение по умолчанию: **`Lib =`**
`ConnectData =` `{текст}`	Параметры **Oracle**-соединения. Поддерживаются два формата задания параметра: • `"USER/PASSWORD@CONNECTION"` – синтаксис **Oracle**; • `"DSN=value;UID=value;PWD=value"` – синтаксис **ODBC**. Для начала работы необходимо, как минимум, задать название DSN, который ссылается на нужную базу данных. Правила задания **`ConnectData`** в случае использования синтаксиса **Oracle** см. в примечании под таблицей. Дополнительно рекомендуется использовать параметр `connect_timeout`, задающий время ожидания подключения. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **`ConnectData =`**
`SizeLimit =` `{числовое значение}`	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. При значении `0` ограничения отсутствуют. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **`SizeLimit =`** `0`
`SkipDomains =` `{LookupLite}`	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка зачастую позволяет значительно снизить нагрузку на сервер и повысить производительность. Обратите внимание, что значение параметра – LookupLite. Значение параметра может быть указано в локальных настройках Lookup Значение по умолчанию: **`SkipDomains =`**
`OnError =` `{ignore | exception}`	Задает процедуру обработки ошибок, возникающих в обработке `Lookup` при обращении к указанному источнику данных. • `ignore` – следует игнорировать возникшую ошибку и продолжить обработку письма (происходит только регистрация ошибки в журнале); • `exception` – следует сгенерировать исключение, которое будет обрабатываться как ошибка обработки письма, в соответствии со значением параметра **`ProcessingError`**, заданного в настройках того компонента, при работе которого эта ошибка возникла. Значение параметра может быть указано в локальных настройках Lookup.

	Значение по умолчанию: **OnError** = ignore

Примечания:

1. С **Oracle** **Dr.Web MailD** работает через библиотеку `libclntsh`, которая поставляется совместно с клиентом **Oracle** и поддерживает версию `OTL v8` или выше.
2. Для подключения к **Oracle** необходимо указать в значении параметра **ConnectData** имя пользователя, пароль и название подключения: `user/password@CONNECTION`.

   Название подключения можно задать двумя способами:

   - если **Dr.Web MailD** установлен на том же компьютере, что и **Oracle**, то сперва необходимо задать для **Dr.Web MailD** переменную окружения `ORACLE_HOME` согласно документации на СУБД **Oracle**. Потом нужно указать в качестве названия подключения одно из имен `TNS` в файле `$ORACLE_HOME/network/admin/tnsnames.ora`;
   - также можно скопировать (без символов переноса строки) описание подключения непосредственно из `$ORACLE_HOME/network/admin/tnsnames.ora`, расположенного на сервере.

     Пример:

     Имеется файл `tnsnames.ora`:

     ```
     CONNECTIONNAME =
      (DESCRIPTION =
       (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
       (CONNECT_DATA =
         (SERVER = DEDICATED)
         (SERVICE_NAME = CONNECTIONNAME)
       )
      )
     ```

     Соответственно, можно указать в качестве строки подключения:

     ```
     user/password@CONNECTIONNAME
     ```

     либо:

     ```
     user/pasword@(DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST =
     localhost)(PORT = 1521))(CONNECT_DATA = SERVER = DEDICATED)
     (SERVICE_NAME = CONNECTIONNAME)))
     ```

3. В случае если указанный хост или сама база данных недоступна, попытка установления подключения к СУБД будет производиться в течение таймаута, заданного в строке подключения при помощи параметра **connect_timeout**. По его истечении будет зафиксирована ошибка, которая будет обработана в соответствии со значением параметра **OnError**.

Секция [ODBC]

В секции `[ODBC]` собраны настройки взаимодействия **Dr.Web MailD** с базами данных через **ODBC**:

Lib = {путь к файлу}	Путь к библиотеке, поддерживающей **ODBC** версии 3.0 или выше. Библиотека должна быть собрана с поддержкой потоков. Рекомендуется использовать **UnixODBC**. Поиск библиотеки осуществляется в соответствии с правилами системного вызова `dlopen` (см. документацию по `dlopen`). В текущей версии продукта изменение этого параметра не может быть применено с помощью сигнала `HUP`. Необходимо перезапустить **Dr.Web MailD**.

	Значение по умолчанию: **Lib** = /usr/lib/libodbc.so
ConnectData = {текст}	Параметры ODBC-соединения. Поддерживаются два формата задания параметра: • "USER/PASSWORD/@DSN" – синтаксис **Oracle**; • "DSN=value;UID=value;PWD=value" – синтаксис **ODBC**. Для начала работы необходимо, как минимум, указание DSN. Дополнительно рекомендуется использовать параметр connect_timeout, задающий время ожидания подключения. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **ConnectData** =
SizeLimit = {числовое значение}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. При значении 0 ограничения отсутствуют. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **SizeLimit** = 0
SkipDomains = {LookupLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка зачастую позволяет значительно снизить нагрузку на сервер и повысить производительность. Обратите внимание, что значение параметра – LookupLite. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **SkipDomains** =
OnError = {ignore | exception}	Задает процедуру обработки ошибок, возникающих в обработке Lookup при обращении к указанному источнику данных. • ignore – следует игнорировать возникшую ошибку и продолжить обработку письма (происходит только регистрация ошибки в журнале); • exception – следует сгенерировать исключение, которое будет обрабатываться как ошибка обработки письма, в соответствии со значением параметра **ProcessingError**, заданного в настройках того компонента, при работе которого эта ошибка возникла. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **OnError** = ignore

Dr.Web MailD работает с **ODBC** через любую библиотеку, которая поддерживает версию **ODBC** 3.0 или выше. Библиотека должна быть собрана с поддержкой потоков. Рекомендуется использовать **UnixODBC** 2.0 или выше.

В случае если указанный хост или сама база данных недоступна, попытка установления подключения к СУБД будет производиться в течение таймаута, заданного в строке подключения при помощи параметра **connect_timeout**. По его истечении будет зафиксирована ошибка, которая будет обработана в соответствии со значением параметра **OnError**.

Секция [SQLite]

В секции [SQLite] собраны настройки взаимодействия **Dr.Web MailD** с СУБД **SQLite**:

Database = {путь к файлу}	Путь к файлу базы данных **SQLite**. Значение по умолчанию: **Database** =
SizeLimit = {числовое значение}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. При значении 0 ограничения отсутствуют Значение по умолчанию: **SizeLimit** = 1
SkipDomains = {LookupLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка зачастую позволяет значительно снизить нагрузку на сервер и повысить производительность. Обратите внимание, что значение параметра – LookupLite. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **SkipDomains** =
OnError = {ignore | exception}	Задает процедуру обработки ошибок, возникающих в обработке Lookup при обращении к указанному источнику данных. • ignore – следует игнорировать возникшую ошибку и продолжить обработку письма (происходит только регистрация ошибки в журнале); • exception – следует сгенерировать исключение, которое будет обрабатываться как ошибка обработки письма, в соответствии со значением параметра **ProcessingError**, заданного в настройках того компонента, при работе которого эта ошибка возникла. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **OnError** = ignore
Lib = {путь к файлу}	Путь к библиотеке libsqlite3.so. Значение по умолчанию: **Lib** = /usr/lib/libsqlite3.so
BusyTimeout = {числовое значение}	Максимальное время (в миллисекундах), в течение которого **Dr.Web MailD** будет пытаться осуществить запись в базу данных. Значение по умолчанию: **BusyTimeout** = 2000

Пожалуйста, обратите внимание на ряд особенностей использования СУБД **SQLite**:

- В силу особенностей работы СУБД **SQLite**, файл базы данных блокируется каждый раз, когда в него осуществляется запись. Поэтому если с файлом базы данных **SQLite** работает несколько программ, то возможны ситуации, когда очередному записывающему процессу не удастся получить монопольный доступ к хранилищу в течение времени, указанного в значении параметра **BusyTimeout**. В результате процесс записи прервется с ошибкой "Database is locked".
- Следует избегать использования графических интерфейсов к базе данных **SQLite**. Они способны блокировать базу данных "про запас".
- Если сторонний процесс заблокировал файл базы данных надолго, то возможны ошибки при экспорте статистики. Конфликты возможны также, если **Dr.Web MailD** настроен на экспорт различных видов статистики в один и тот же файл базы данных **SQLite**, а заданное время ожидания слишком мало.
- Если база данных **SQLite**, из которой брались настройки для ряда параметров (с помощью Lookup типа sqlite), была недоступна в течение некоторого времени, а потом соединение с базой данных было восстановлено, то для восстановления соединения **Dr.Web MailD** с **SQLite** нужно послать сигнал SIGHUP модулю **drweb-maild**.

Секция [Firebird]

В секции [Firebird] собраны настройки взаимодействия **Dr.Web MailD** с СУБД **Firebird**:

Host = {текст}	Имя хоста, на котором работает база данных **Firebird**. Значение по умолчанию: **Host** = localhost
Database = {текст}	Имя базы данных **Firebird**. Значение по умолчанию: **Database** =
User = {текст}	Имя пользователя базы данных **Firebird**. Значение по умолчанию: **User** =
Password = {текст}	Пароль для доступа к базе данных **Firebird**. Значение по умолчанию: **Password** =
Charset = {текст}	Кодировка, используемая базой данных **Firebird**. Значение по умолчанию: **Charset** = us-ascii
SizeLimit = {числовое значение}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. При значении 0 ограничения отсутствуют Значение по умолчанию: **SizeLimit** = 10
SkipDomains = {LookupLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность.

	Обратите внимание, что значение параметра – LookupLite. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **SkipDomains** =
OnError = {ignore | exception}	Задает процедуру обработки ошибок, возникающих в обработке Lookup при обращении к указанному источнику данных. • ignore – следует игнорировать возникшую ошибку и продолжить обработку письма (происходит только регистрация ошибки в журнале); • exception – следует сгенерировать исключение, которое будет обрабатываться как ошибка обработки письма, в соответствии со значением параметра **ProcessingError**, заданного в настройках того компонента, при работе которого эта ошибка возникла. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **OnError** = ignore
Lib = {путь к файлу}	Путь к библиотеке libFBclient.so Значение по умолчанию: **Lib** = /usr/lib/libFBclient.so

Секция [PostgreSQL]

В секции [PostgreSQL] собраны настройки взаимодействия **Dr.Web MailD** с СУБД **PostgreSQL**:

ConnectionsString = {текст}	Строка с параметрами подключения к СУБД **PostgreSQL**. Параметры задаются в формате keyword = value и разделяются пробелами. Пробелы около знака = не обязательны. Если для какого-либо параметра нужно указать пустое значение, либо если значение параметра содержит пробелы, то оно заключается в одинарные кавычки. Если указана пустая строка, то используются параметры по умолчанию. Более подробную информацию о параметрах вы найдёте по ссылке: http://www.postgresql.org/docs/9.3/static/libpq-connect.html. Дополнительно рекомендуется использовать параметр connect_timeout, задающий время ожидания подключения. **Примеры**: **ConnectionString** = host=localhost port=5432 user=ai password=qwerty dbname=drweb **ConnectionString** = hostaddr=127.0.0.1 port=5432 dbname=mailddb user=mailddbuser password=Str0ngPaSSw0rd connect_timeout=5s Значение по умолчанию: **ConnectionsString** =
SizeLimit = {числовое значение}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных.

	При значении `0` ограничения отсутствуют. Значение по умолчанию: **`SizeLimit`** = `10`
`SkipDomains` = `{LookupLite}`	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность. Обратите внимание, что значение параметра – LookupLite. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **`SkipDomains`** =
`OnError` = `{ignore | exception}`	Задает процедуру обработки ошибок, возникающих в обработке `Lookup` при обращении к указанному источнику данных. • `ignore` – следует игнорировать возникшую ошибку и продолжить обработку письма (происходит только регистрация ошибки в журнале); • `exception` – следует сгенерировать исключение, которое будет обрабатываться как ошибка обработки письма, в соответствии со значением параметра **`ProcessingError`**, заданного в настройках того компонента, при работе которого эта ошибка возникла. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **`OnError`** = `ignore`
`Lib` = `{путь к файлу}`	Путь к библиотеке `libpq.so`. Значение по умолчанию: **`Lib`** = `/usr/lib/libpq.so`

В случае если указанный хост или сама база данных **PostgreSQL** недоступна, попытка установления подключения к СУБД будет производиться в течение таймаута, заданного в строке подключения при помощи параметра **`connect_timeout`** (в указанном примере – 5 сек). По его истечении будет зафиксирована ошибка, которая будет обработана в соответствии со значением параметра **`OnError`**.

Секция [MySQL]

В секции `[MySQL]` собраны настройки взаимодействия **Dr.Web MailD** с СУБД **MySQL**:

`User` = `{текст}`	Имя пользователя базы данных **MySQL**. Значение по умолчанию: **`User`** =
`Password` = `{текст}`	Пароль для доступа к базе данных **MySQL**. Значение по умолчанию: **`Password`** =
`DB` =	Имя базы данных **MySQL**.

	Значение по умолчанию: **DB** =
Host = {имя хоста}	Имя узла, на котором работает база данных **MySQL**. Значение по умолчанию: **Host** = localhost
Port = {адрес порта}	Порт для подключения к базе данных **MySQL**. Требуется также указывать префикс типа сокета (TCP или UNIX). **Пример**: При использовании TCP-сокета: **Port** = tcp://1234 При использовании UNIX-сокета: **Port** = unix:///path/to/socket Значение по умолчанию: **Port** =
Connections = {числовое значение}	Число одновременных подключений к базе данных **MySQL**. При значении 0 подключения будут создаваться по мере обращений к базе данных, что потребует дополнительного времени. Подключения, созданные заранее, могут обслуживать запросы к базе данных в порядке очереди, без затрат времени на создание новых подключений. Значение по умолчанию: **Connections** = 4
SizeLimit = {числовое значение}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. При значении 0 ограничения отсутствуют. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **SizeLimit** = 10
SkipDomains = {LookupLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность. Обратите внимание, что значение параметра – LookupLite. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **SkipDomains** =
OnError = {ignore | exception}	Задает процедуру обработки ошибок, возникающих в обработке Lookup при обращении к указанному источнику данных. • ignore – следует игнорировать возникшую ошибку и продолжить обработку письма (происходит только регистрация ошибки в журнале);

	• `exception` – следует сгенерировать исключение, которое будет обрабатываться как ошибка обработки письма, в соответствии со значением параметра **`ProcessingError`**, заданного в настройках того компонента, при работе которого эта ошибка возникла. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **`OnError`** = `ignore`
`Lib` = `{путь к файлу}`	Путь к библиотеке `libmysqlclient_r.so`. **Dr.Web MailD** работает только с библиотекой с поддержкой потоков. Значение по умолчанию: **`Lib`** = `/usr/lib/libmysqlclient_r.so`

Обратите внимание, что в операционной системе **FreeBSD** версии 6.4/amd64 при использовании библиотеки `libmysqlclient_r.so` возможно возникновение следующей ошибки: `Undefined symbol "gethostbyname_r"`

В случае если указанный хост или сама база данных **MySQL** недоступна, попытка установления подключения к СУБД будет производиться в течение таймаута, равного 2 сек. По его истечении будет зафиксирована ошибка, которая будет обработана в соответствии со значением параметра **`OnError`**.

Секция [CDB]

В секции `[CDB]` собраны настройки взаимодействия **Dr.Web MailD** с базой данных **CDB**:

`Sources` = `{путь к файлу}`	Путь к файлу базы данных **CDB**. Значение по умолчанию: **`Sources`** =
`SkipDomains` = `{LookupLite}`	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность. Обратите внимание, что значение параметра – LookupLite. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **`SkipDomains`** =
`OnError` = `{ignore | exception}`	Задает процедуру обработки ошибок, возникающих в обработке `Lookup` при обращении к указанному источнику данных. • `ignore` – следует игнорировать возникшую ошибку и продолжить обработку письма (происходит только регистрация ошибки в журнале); • `exception` – следует сгенерировать исключение, которое будет обрабатываться как ошибка обработки письма, в соответствии со значением параметра **`ProcessingError`**, заданного в настройках того компонента, при работе которого эта ошибка возникла.

	Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **OnError** = ignore

База данных **CDB** представляет собой доступное только для чтения хранилище пар `[текстовый ключ]:[текстовое значение]`. Для создания файла базы данных можно использовать пакет **tinycdb**. Каждый файл представлен в виде одной таблицы, названием которой является только имя файла без указания полного пути к нему.

База данных **CDB** не поддерживает язык запросов SQL. Поэтому драйвер эмулирует единственную SQL-команду для унификации работы с `Lookup`:

```
select * from @tablename where key='@string'
```

где `@tablename` – имя файла.

Секция [Berkeley]

В секции `[Berkeley]` собраны настройки взаимодействия **Dr.Web MailD** с базой данных **Berkeley**:

Databases = {путь к файлу}	Путь к файлу базы данных **Berkeley**. Значение по умолчанию: **Databases** =
Environment = {путь к каталогу}	Путь к каталогу, используемому для хранения файлов блокировки **Berkeley**. Значение по умолчанию: **Environment** =
SizeLimit = {числовое значение}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. Допустимые значения от 1024 до 65536, другие значения будут преобразованы к ближайшему допустимому значению. Значение по умолчанию: **SizeLimit** = 1
SkipDomains = {LookupLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка зачастую позволяет значительно снизить нагрузку на сервер и повысить производительность. Обратите внимание, что значение параметра – LookupLite. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **SkipDomains** =
OnError = {ignore | exception}	Задает процедуру обработки ошибок, возникающих в обработке `Lookup` при обращении к указанному источнику данных. • `ignore` – следует игнорировать возникшую ошибку и продолжить обработку письма (происходит только регистрация ошибки в журнале); • `exception` – следует сгенерировать исключение, которое

	будет обрабатываться как ошибка обработки письма, в соответствии со значением параметра **ProcessingError**, заданного в настройках того компонента, при работе которого эта ошибка возникла. Значение параметра может быть указано в локальных настройках Lookup. Значение по умолчанию: **OnError** = ignore
Lib = {путь к файлу}	Путь к библиотеке libdb.so. Обычно при установке создается символическая ссылка /usr/lib/libdb.so, указывающая на текущую библиотеку. Если такой ссылки не создано, следует указать правильную версию библиотеки (т.е. /usr/lib/libdb-4.5.so). Значение по умолчанию: **Lib** = /usr/lib/libdb.so

Dr.Web MailD предназначен для работы с библиотеками версий 4.3 – 4.6.

Проксирование

В данном разделе описаны секции конфигурационного файла, содержашие параметры работы компонентов проксирования сервиса **Dr.Web для почтовых серверов UNIX**.

Проксирование используется для обеспечения прозрачного распределения модулей, составляющих **Dr.Web MailD**, по нескольким хостам.

Обратите внимание, что при проксировании на одних хостах работают компоненты **Sender** и **Receiver**, а на других – центральный компонент обработки **MailD core**. На тех хостах, на которых работают компоненты **Sender** и **Receiver**, запускается компонент проксирования **Proxy client**, а на хостах, на которых работает **MailD core**, запускается компонент **Proxy server**.

На хосте, на котором функционирует компонент **Proxy client**, в файле конфигурации **Dr.Web MailD** должна присутствовать секция [ProxyClient], а на хостах, на которых функционирует компонент **Proxy server** – секция [ProxyServer].

Если проксирование не используется, то обе секции файле конфигурации могут отсутствовать.

Секция [ProxyClient]

В секции [ProxyClient] находятся настройки модуля **drweb-proxy-client** (компонент **Proxy client**), отвечающего за работу прокси-клиента модулей **Dr.Web MailD**.

ProxyServersAddresses = {список адресов}	Список адресов сокетов, на которых слушают компоненты **drweb-proxy-server**. Адреса задаются в виде: ADDRESS1 [WEIGHT1], ADDRESS2 [WEIGHT2] .., где ADDRESS указан в стандартном формате, а WEIGHT - представляет собой необязательный вес этого адреса. WEIGHT определяет относительную нагрузку на данный узел сети, и может принимать значения от 0 до 100 включительно. Почта, полученная от компонента **Receiver**, запущенного на хосте вместе с **drweb-proxy-client**, будет передаваться на проверку по указанным адресам. Среди указанных адресов должен присутствовать хотя бы один корректный адрес сервера. Выбор адресов осуществляется в

	соответствии с алгоритмом, описанным в главе Использование прокси. Значение по умолчанию: **ProxyServersAddresses** = inet:8088@SERVER-IP
Address = {список адресов}	Список адресов сокетов, на которых компонент **Sender** слушает запросы на отправку почты от компонентов **drweb-proxy-server**. Компоненты **drweb-proxy-server** будут отправлять почту на эти адреса в соответствии со значением параметра **ProxyClientAddresses** секции [ProxyServer]. Значение по умолчанию: **Address** = inet:8066@0.0.0.0
MailPoolOptions = {настройки пула}	Настройки пула потоков, обрабатывающих запросы от компонента **Receiver**. Пул потоков обабатывает запросы от компонента **Receiver** и отправляет письма удаленно в **drweb-proxy-server** на проверку. Затем по результатам проверки письмо либо передается назад компоненту **Receiver**, либо отправляется через компонент **Sender**. Значение по умолчанию: **MailPoolOptions** = auto
SenderPoolOptions = {настройки пула}	Настройки пула потоков, обрабатывающих запросы от **drweb-proxy-server** на отправку почты через компонент **Sender**. Перед отправлением письма в **Sender** создается временный каталог, в который сбрасывается сообщение, а затем оно отдается на передачу компоненту **Sender**. Результаты передачи сообщения возвращаются назад в **dwreb-proxy-server**. Значение по умолчанию: **SenderPoolOptions** = auto

Секция [ProxyServer]

В секции [ProxyServer] находятся настройки модуля **drweb-proxy-server** (компонент **Proxy server**), отвечающего за работу прокси-сервера модулей **Dr.Web MailD**.

Address = {список адресов}	Список адресов сокетов, на которых компонент **drweb-proxy-server** ожидает запросов от компонентов **drweb-proxy-client**. **drweb-proxy-client** передает письма на проверку компоненту **drweb-proxy-server** в соответствии со значением параметра **ProxyServersAddresses** в секции [ProxyClient]. Значение по умолчанию: **Address** = inet:8088@0.0.0.0
ProxyClientsAddresses = {список адресов}	Список адресов сокетов, на которых компоненты **drweb-proxy-client** принимают запросы на отправление писем. Адреса заданы в виде: ADDRESS1 [WEIGHT1], ADDRESS2 [WEIGHT2] .., где ADDRESS указан в стандартном формате, а WEIGHT - представляет собой необязательный вес этого

	адреса. `WEIGHT` определяет относительную нагрузку на данный узел сети, и может принимать значения от `0` до `100` включительно. Адреса сокетов, указанные в значении данного параметра, должны соответствовать прослушиваемым адресам, указанным в значении параметра **Address** секции `[ProxyClient]`. Значение по умолчанию: **ProxyClientsAddresses** = `inet:8066@CLIENT-IP`
ReceiverPoolOptions = {настройки пула}	Настройки пула потоков, отвечающих за передачу сообщения на проверку в **drweb-maild**. Потоки в пуле принимают запросы на проверку сообщения от **drweb-proxy-client**, создают локальный идентификатор для полученного сообщения и передают сообщение на проверку в **drweb-maild**. Затем по результатам проверки в **drweb-proxy-client** возвращается оригинальное или модифицированное сообщение. Значение по умолчанию: **ReceiverPoolOptions** = `auto`
SenderPoolOptions = {настройки пула}	Настройки пула потоков, отвечающих за отправление писем в **drweb-proxy-client** для отправки их через компонент **Sender**. Потоки в пуле принимают запросы на отправку почты от различных компонентов и затем передают их на обработку в **drweb-proxy-client**. Результат обработки передается обратно компонентам, запросившим отправку сообщения. Значение по умолчанию: **SenderPoolSettings** = `auto`

Статистика

В процессе работы комплекса может собираться статистика двух видов: общая статистика и статистика по заблокированным сообщениям. Общая статистика представляет из себя общую информацию о работе комплекса **Dr.Web для почтовых серверов UNIX** за определенный период: число проверенных сообщений, их размер, число сообщений, отмеченных как спам, и т.д. Статистика по заблокированным собщениям представляет собой информацию по конкретным письмам, в которых было обнаружено что-либо нежелательное, например, вирус.

Вся статистика хранится во внутренней базе данных **Dr.Web для почтовых серверов UNIX**. Общая статистика накапливается во внутреннем кэше и периодически (раз в 5 минут) сбрасывается в базу данных. Статистика по заблокированным сообщениям сохраняется непосредственно в базе данных и при необходимости может быть экспортирована.

Существует несколько уровней подробности статистики, которые задаются с помощью параметра **Detail** из секции `[Stat]`.

- `off` – полностью отключает сбор всякой статистики, что увеличивает производительность программного комплекса **Dr.Web для почтовых серверов UNIX**, но в результате этого функции отправки отчетов или экспорта статистики теряют смысл.
- `low` – включает сбор статистики по всей системе. В результате становится возможным пользоваться отчетами и экспортом статистики.

- `medium` – в дополнение к статистике на уровне `low` позволяет собирать статистику по всем группам, для которых ведение статистики явно не выключено. Доступ к статистике по группам можно получить либо через управляющий сокет, либо через веб-интерфейс.
- `high` – в дополнение к статистике на уровне `medium` позволяет собирать статистику по всем пользователям, внесенным во внутреннюю базу данных, для которых ведение статистики явно не выключено. Доступ к статистике для каждого пользователя можно получить либо через управляющий сокет, либо через веб-интерфейс.

Экспорт статистики

Существует возможность экспорта статистики как через компонент **Dr.Web Agent** (см. раздел **Dr.Web Agent**), так и средствами компонента **MailD core** с помощью типа `Storage`. Причем обе эти возможности можно включить одновременно.

Экспорт статистики через **Dr.Web Agent** по умолчанию выключен.

Когда вся статистика передается **Dr.Web Agent**, то он либо отсылает ее на сервер вирусной статистики компании **«Доктор Веб»** (подробнее смотрите параметры **`StatisticServerHost`**, **`StatisticServerPort`** и **`UUID`** в секции `[StandaloneMode]` конфигурационного файла `agent.conf`), либо отправляет ее на сервер централизованной зашиты **Dr.Web**, если работает в `Enterprise`-режиме (подробнее смотрите секцию `[EnterpriseMode]` конфигурационного файла `agent.conf`).

Для включения экспорта статистики через тип `Storage` надо указать `Yes` как значение параметра **`ExportStat`** из секции `[Stat]`, затем нужно задать значение, как минимум, для одного из следующих параметров секции `[Stat]`, указав в нем команды экспорта статистики:

- **`ExportBlockObjectsStorage`** – список объектов для экспорта статистики по заблокированным сообщениям.
- **`ExportStatStorage`** – экспорт статистики по общему числу сообщений, обработанных комплексом **Dr.Web для почтовых серверов UNIX**.
- **`ExportPluginStatStorage`** – экспорт статистики по числу обработанных сообщений для каждого подключаемого модуля.

Подробнее описание каждого из приведенных выше параметров можно найти в главе Секция [Stat].

Пожалуйста, обратите внимание на особенности в экспорте статистики через **Dr.Web Agent**, заключающиеся в том, что по отдельности формируется и передается два вида статистики:

- Вирусная статистика по всем найденным и обработанным вирусам и прочим вредоносным объектам;
- Статистика по всем обработанным почтовым сообщениям и действиям над ними.

Эти два вида статистики друг с другом никак не связаны, т.е. если было обработано письмо, во вложении которого было обнаружено 5 различных вирусов, и для каждого вируса будет применено некоторое действие (например – `cure`), то в статистике по вирусам будет отправлено 5 записей, а в статистике по обработке почтовых сообщений будет отправлена только одна запись с информацией о действии, примененном к письму (например – `pass` для данного письма).

Карантин

Письма попадают в **Карантин** как по запросу от основного модуля **`drweb-maild`**, так и по запросу от любого из используемых подключаемых модулей. Далее они сохраняются в каталоге `/quarantine/path/def/name/`, где `name` – название модуля, запросившего сохранение.

При сохранении письма в **Карантине** создаются два файла:

- Первый файл с именем `name` (которое формируется в соответствии с настройками **`FilenamesMode`** и **`FilenamesPrefix`**) содержит оригинальное тело сообщения (при этом все символы "_" заменяются на ".").
- Второй файл с именем `name.envelope` содержит оригинальный конверт сообщения в следующим формате:
  - `int4_t` – длина адреса отправителя;
  - `sN` – адрес отправителя;
  - `int4_t` – число получателей;
  - `int4_t sN` – для каждого получателя, где `int4_t` – 4-байтовое число со знаком в сетевом порядке байтов.

Также необходимо отметить, что если значение параметра **`MoveAll`** установлено в `Yes`, то вся проходящая через программный комплекс почта будет сохраняться в каталоге `/path/def/backup/`.

Кроме сохранения тела письма, в каталоге **Карантина** происходит регистрация сообщения во внутренней базе данных с сохранением там дополнительной информации о письме (например, сохраняется конверт письма, время перемещения письма в **Карантин**, указывается причина перемещения и т.д.).

Работу с **Карантином** можно осуществлять непосредственно через управляющий сокет, эффективно выполняя поиск, отправление, пересылку, удаление и другие операции с содержимым **Карантина**.

Можно задать максимальный срок хранения писем в каталоге **Карантина** через настройку параметра **`StoredTime`**, а также ограничить **Карантин** по максимальному размеру (с помощью параметра **`MaxSize)`** и максимальному числу хранимых сообщений (с помощью параметра **`MaxNumber)`**. Если одновременно задано несколько ограничений, то все они будут поддерживаться одновременно.

Ограничения на максимальный размер **Карантина** и число сообщений в нем проверяются при каждом сохранении сообщения в **Карантин**. Ограничение максимального времени хранения писем в **Карантине** проверяется периодически - период устанавливается через настройку параметра **`PulseTime`**.

Удалением устаревших сообщений и перемещением их во внешнее хранилище **DBI** занимается внешняя утилита **`drweb-qp`**, путь к которой указывается в настройке **`PathToDrwebQp`**. Для ее работы необходим **Perl** (версии не ниже 5.0). Запуск данной утилиты происходит с периодичностью, указанной в настройке **`PulseTime`**. Если значение параметра **`PulseTime`** установлено равным `0` и выключено использование управляющих писем, то запуск производится не будет.

Использование DBI

Существует возможность хранения сообщений, находящихся в **Карантине**, не в файловой системе, а в хранилище **DBI**. Используемое хранилище **DBI** должно быть предварительно настроено. Подробнее про установку и настройку модулей **DBI** для работы с базами данных можно посмотреть в документации по **DBI**. Кроме того, чтобы успешно сохранять письма целиком в БД, необходимо, чтобы она была создана с использованием наборов символов `SQL-ASCII`.

Работа с **DBI** осуществляется через **внешнюю утилиту**, путь к которой указывается в значении параметра **PathToDrwebQp** (в секции [Quarantine]).

По умолчанию используется поставляемая с **Dr.Web MailD** утилита **drweb-qp**, которая расположена в каталоге %bin_dir.

Для использования утилиты **drweb-qp** необходимы **Perl** (версии не ниже 5.0), установленные модули **Perl DBI** и **File::Temp**.

Для использования **DBI** необходимо установить **MoveToDBI** в Yes и настроить **DBISettings**, **DBIUsername** и **DBIPassword** соответствующим образом для доступа к хранилищу **DBI**.

Также надо настроить соответствующие SQL-команды для выполнения требуемых действий:

- **SQLInsertCommand** – команда добавления письма в хранилище **DBI**.
- **SQLRemoveCommand** – команда для удаления письма из **DBI**. Используется, если задано ограничение на время хранения писем в **Карантине**.
- **SQLSelectCommand** – команда доступа к сохраненному сообщению в хранилище **DBI**. Используется при запросе письма из **Карантина** (например, через управляющее письмо).

Возможные проблемы:

Если возникла ошибка вида:

```
maild ERROR Error in system call for
[/opt/drweb/drweb-qp --Level debug
--SyslogFacility Daemon --BaseDir /var/drweb/
--ProcessMail 1 --MoveToDBI 0
--StoredTime 86400 --SQLInsertCommand ""
--MDClient "def" >/dev/null 2>&1 &]
```

то попробуйте увеличить макимально доступное количество памяти для процесса **drweb-maild** (например, с помощью команды **ulimit** -m).

Использование управляющих писем

Доступ к **Карантину** можно получить через специальные *управляющие письма*, которые в поле Subject содержат команды, которые надо выполнить. Письма нужно отправлять на адрес, заданный значением параметра **FilterMail** (в секции [Notifier] конфигурационного файла **Dr.Web MailD** либо в локальных Правилах обработки почты, истинных для данного письма). Настройка разрешений для управляющих писем производится с помощью задания значения параметра **OnlyTrustedControlMails** секции [Maild] конфигурационного файла **Dr.Web MailD**.

Получение писем, сохраненных в **Карантине**, контролируется параметром **AccessByEmail** секции [Quarantine] конфигурационного файла **Dr.Web MailD**.

Для получения определенного письма из **Карантина** надо в поле Subject управляющего письма написать:

```
q:relative_path_to_file
```

где relative_path_to_file – относительный путь к файлу в **Карантине** (например, /def/drweb/F/00014F8F.DW_SHOT_PRODUCT.QqFpdH).

Письмо запрашивается из **Карантина**, только если отправитель или один из получателей совпадает с отправителем *управляющего письма*.

Управляющее письмо может автоматически генерироваться MUA пользователя при нажатии им соответствующей ссылки в уведомлениях, высылаемых **Dr.Web Notifier** при помещении

сообщения в **Карантин**.

Миграция на новую версию Карантина

Начиная с **Dr.Web MailD** версии 6.0 формат **Карантина** поменялся: в файловой системе теперь находится только тело письма, а конверт и дополнительная информация хранятся во внутренней базе данных.

Для перевода **Карантина** старых версий (в **Dr.Web MailD** версии 5.0 и ниже) в новый формат служит специальный скрипт **quarantine_migration.pl**, находящийся в каталоге `%bin_dir/maild/scripts/`. После запуска он определит все необходимые настройки по умолчанию и предложит запустить процесс миграции на новую версию **Карантина**. Миграция происходит полностью в автоматическом режиме. После окончания миграции скрипт выведет информацию по результатам работы: время начала и окончания работы, число обработанных, пропущенных и вызвавших ошибку писем.

Интерактивное управление

Во время работы программного комплекса **Dr.Web для почтовых серверов UNIX** возможно интерактивное управление некоторыми его частями. Для этого надо:

1. Включить данную возможность, задав в секции `[Maild]` конфигурационного файла **Dr.Web MailD** значение `Yes` для параметра **Control**;
2. Подключиться к адресу, указанному в значении параметра **ControlAddress** в той же секции, и в интерактивном режиме вводить требуемые команды (для этого следует использовать любой текстовый клиент, например, **telnet**).

Взаимодействие происходит по строкам, т.е. сперва пользователь вводит некую строку, а затем система выводит ему ответ. Таким образом, возможность использовать многострочные команды не предусмотрена (соответственно, при вводе Правил для пользователя или группы их также нужно будет вводить построчно).

Окончанием вывода информации от **Dr.Web MailD** служит пустая строка.

Одновременно поддерживается несколько соединений.

Независимо от значения параметра **Control** конфигурационного файла **Dr.Web MailD**, прослушивающий сокет для осуществления интерактивного управления всегда открывается по адресу `/значение_параметра_BaseDir/ipc/.ctl`. При задании адреса прослушивающего сокета поддерживается как формат IPv4, так и формат IPv6.

При помощи интерфейса интерактивного управления можно:

- Управлять общим состоянием программного комплекса;
- Управлять пользователями, объединять их в группы, задавать алиасы и параметры обработки писем;
- Управлять **Карантином** и его содержимым;
- Получать накопленную статистику;
- Проверять работу модуля генерации уведомлений **Notifier.**

Общие команды управления

Перечень общих команд с их описаниями:

Команда	Описание
help [<section_name> | <command_name> | all]	Вывод справки по имеющимся секциям команд. В качестве аргумента команды можно указать название секции, чтобы узнать справку по всем командам из нее, а также название конкретной команды, чтобы увидеть справку только по ней. Список всех команд можно получить, введя команду **help** all.
option [regex]	Вывод значений настроек, с которыми работает как **Dr.Web MailD**, так и загруженные подключаемые модули (которые получили свои настройки через **Dr.Web MailD**), и имена которых совпадают с заданным регулярным выражением. Если регулярное выражение не задано, то выводятся все настройки.
db-state	Вывод текущего состояния внутренней БД **Dr.Web MailD**. Состояние БД выводится в формате: Number: NC/NM Size: SC/SM где NC и NM – текущее и максимальное число сообщений в БД, а SC и SM – текущий и максимальный размер БД в байтах. Если NM или SM равно 0, то это означает отсутствие ограничений на максимальное количество сообщений или размер БД (ограничения можно задать в настройках).
queue-state	Выводит текущее состояние сообщений, находящихся во внутренней очереди для обработки. Выводится как общее число сообщений, так и информация по каждому сообщению. Большое число сообщений в очереди может указывать на нехватку потоков во втором пуле **Dr.Web MailD** (контролируемом параметром **OutPoolOptions**);
send-stat	Форсирует отправку/экспорт статистики (как если бы истекло время ожидания, заданное в параметре **SendPeriod** секции [Stat] конфигурационного файла **Dr.Web MailD**). Для выполнения необходимо, чтобы значение параметра **Send** из той же секции было установлено в Yes. Производится передача статистики **Dr.Web Agent**.
send-report [period]	Форсирует отправление письма с отчетом по работе подключаемых модулей (как если бы истекло время ожидания, заданное в параметре **SendTimes** секции [Reports] конфигурационного файла **Dr.Web MailD**). Для выполнения необходимо, чтобы значение параметра **Send** из той же секции было установлено в Yes. При этом period указывает интервал, за который надо отправлять отчет (в формате {time}). Если значение не указано, то отчет отсылается за последние 24 часа.
backup	Форсировать выполнения резервного копирования внутренней БД
quarantine-pulse	Форсирует запуск утилиты **drweb-qp** по обработке **Карантина**, как если бы истекло время ожидания, заданное в параметре **PulseTime** секции [Quarantine] конфигурационного файла **Dr.Web MailD**.
dump-cache-stat	Сброс всей кэшированной статистической информации из памяти во внутреннюю БД.
get [(id1|-|id1-[id2]) [(plugin_name|-)]]	Вывод информации по сохраненным во внутренней БД сообщениям. При этом:

Команда	Описание
	• id – номер запрашиваемого сообщения, • id1-id2 – вывод сообщений с номерами в запрашиваемых диапазонах, • id1- – вывод всех сообщений, начиная с номера id1 (номера должны задаваться в шестнадцатеричном виде), • plugin_name – имя модуля, который попросил сохранения сообщения в БД. Символ "–" эквивалентен отсутствию параметра. При отсутствии параметров выводятся все сохраненные в БД сообщения. **Пример**: **get** - drweb – вывод всех сообщений, сохраненных подключаемым модулем **Drweb**. **get** – вывод всех сохраненных сообщений.
send [(id1|-|id1-[id2]) [(plugin_name|-)] [force]]	Отправление заданных сообщений получателям из конверта. Отправляются только еще не отосланные сообщения (т.е. такие сообщения, для которых состояние send=no в выводе команды **get)**. Параметры аналогичны команде **get**, за исключением нового параметра force, который заставляет отправить сообщения, для которых флаг send установлен в Yes.
export [(id1|-|id1-[id2]) [(plugin_name|-)] [(dir_name|-)] [env]	Сохранение заданных сообщений из БД во внешних файлах. Параметры аналогичны команде **get**, за исключением новых параметров: • dir_name – название каталога, в который нужно производить сохранение файлов. Если каталог не указан, то используется значение параметра **BaseDir** секции [General] конфигурационного файла **Dr.Web MailD**; • env – если указано, то экспортируется и конверт в формате: ○ первая строка – отправитель; ○ вторая строка – получатели, разделенные запятыми. Имя файла составляется из номера сообщения и расширения .eml, а имя файла конверта – из номера сообщения и расширения .envelope. **Пример**: **export** 00002D94 vaderetro /t env Success export body to /t/00002D94.eml and envelope to /t/00002D94.envelope
remove [(id1|-|id1-[id2]) [(plugin_name|-)]	Удаляет заданные сообщения из БД. Параметры аналогичны команде **get**. **Пример**: **remove** 00002D93 Success remove record 00002D93
send_and_remove [(id1|-|id1-[id2]) [(plugin_name|-)] [force_send] [ignore_send_error]	Отправление и удаление заданных сообщений. Значение параметра force_send аналогично параметру force команды **send**. Если командой **send_and_remove** сообщение было успешно отправлено, или для него не требуется отправка (т.е. оно было отправлено ранее), то оно удаляется. Если задан параметр ignore_send_error, то сообщение удаляется (независимо от успешности отправки).
version	Вывод текущей версии продукта.
stop	Остановка продукта.
reload	Отправление процессу **drweb-maild** сигнала SIGHUP.

Управление пользователями, группами и алиасами

Понятие пользователя, группы и алиаса

Пользователем в **Dr.Web для почтовых серверов UNIX** называется владелец одного или более почтовых адресов, вся корреспонденция которого должна обрабатываться с использованием отдельных настроек. Если с пользователем связано более одного почтового адреса, то все адреса, используемые пользователем, называются *алиасами* (при этом один из адресов является главным).

С пользователем могут быть связаны индивидуальные Правила обработки его писем, аналогично тому, как они задаются в общих Правилах обработки в конфигурационном файле (в секции `[Rules]`). Идентификатором пользователя с точки зрения **Dr.Web для почтовых серверов UNIX** является его почтовый адрес (или любой алиас). Все адреса пользователя считаются единым целым и для них используется одинаковый набор настроек и собирается единая статистика.

Кроме едничных пользователей, в базе данных можно также создавать *группы пользователей*, к которым также можно привязывать свои Правила обработки писем. При этом любой пользователь может входить в произвольное число групп. Как для пользователя, так и для группы может быть установлено два флага:

- **Флаг активности** (**A**) – активен ли пользователь (группа). Если да, то заданные для пользователя Правила будут использоваться при обработке его писем. В противном случае эти Правила будут игнорироваться.
- **Флаг статистики** (**S**) – включен ли для пользователя (группы) сбор статистики. Чтобы была возможность вести статистику по каждому пользователю отдельно, уровень подробности общей статистики должен быть установлен в `high`.

Алгоритм поиска значений параметров при обработке писем

При выборе значения параметра, которое следует применить к обрабатываемому письму, используется следующий алгоритм:

- При выборе значений параметров, которые следует применить к обрабатываемому письму, используется следующий алгоритм:
  - Просматриваются Правила, имеющиеся во встроенной базе данных и связанные с **получателем** данного письма (получатель определяется по заданному отправителем `RCPT TO`).
  - Просматриваются Правила, имеющиеся во встроенной базе данных и связанные со всеми группами, к которым относится пользователь-получатель. Просмотр Правил групп производится в обратном порядке: с настроек самой последней группы и до первой в списке группы.
  - Просматриваются Правила, заданные в секции `[Rules]` основного конфигурационного файла.

  Обратите внимание на порядок обхода Правил:
  - Все Правила в текущей просматриваемой группе Правил всегда проверяются в порядке их задания.
  - Для каждого проверяемого Правила проверяется условие `CONDITION` – и если оно истинно, то значение требуемого параметра ищется среди элементов секции `SETTINGS` этого правила.
  - Если условие `CONDITION` оказалось ложно, то просмотр Правила заканчивается и происходит переход к поиску значения в следующем Правиле.
  - Если условие `CONDITION` истинно и после него стоит директива `cont`, то происходит переход к проверке следующего Правила. Если же после истинного `CONDITION` стоит директива `stop`, то просмотр Правил заканчивается вне зависимости от того, было

найдено значение требуемого параметра или нет.

Значение параметра по результатам просмотра Правил всегда определяется следующим образом:

- Если искомый параметр встретился в одном из сработавшем правил, то используется его значение, извлеченное из части `SETTINGS` (обратите внимание, что при срабатывании нескольких Правил для одного и того же параметра, результирующее значение этого параметра зависит от его семантики. Подробнее об этом см. в разделе Правила обработки писем).
- Если Правила отсутствуют, или ни одно Правило не сработало, или ни в одном из сработавших Правилах параметр не нашелся, то извлекается значение этого параметра, заданное в соответствующей секции конфигурационного файла.
- Если в конфигурационном файле искомый параметр не задан, то используется его значение по умолчанию.

Из этого следует, что порядок указания групп для пользователя важен и определяет, какие настройки будут применяться к данному адресу.

Если в письме указано больше одного получателя и для каждого из них найдены разные значения для одного и того же параметра, то возможны два варианта действий:

1. Для ряда настроек происходит клонирование письма (т.е. создаются две копии с разными получателями в конвертах), и для каждой копии применяется свое значение настройки.
2. Настройки, не предусматривающие клонирование письма, просто игнорируются, и в этом случае используется либо значение из настроек пользователя, либо глобальная настройка из конфигурационного файла, либо берется значение параметра по умолчанию.

При поиске настроек все Правила (как для конкретного пользователя, так и для группы, в которую он входит) рассматриваются одним списком (пользовательские правила в начале списка, групповые – в конце). Соответственно, при проверке этих списков для разных получателей одного письма настройка из "пользовательской" части списка для одного получателя может совпасть с настройкой из "групповой" части списка для другого получателя, а после обнаружения такого совпадения применяется вышеописанный алгоритм.

Для работы с пользователями, алиасами и группами можно использовать как интерфейс управляющего сокета, так и веб-интерфейс.

Команды интерфейса интерактивного управления для просмотра перечня имеющихся пользователей и групп

- **email-info**

  Выводит всю информацию о пользователе. Для каждого пользователя в базе данных кроме, собственно, его Правил, сохраняется дополнительная информация. Формат вывода этой информации следующий:

```
[client-id/]email A=0|1 S=0|1
name: username
aliases: alias1 alias2 ...
groups: group1 group2 ...
rules:
1: SETTINGS1
2: SETTINGS2
...
custom:
tag1: info1..
tag2: info2..
...
```

  Здесь:

  - `client-id` – пустая строка;
  - `A` – активен пользователь или нет. Если пользователь не активен, то все связанные с ним Правила игнорируются;
  - `S` – вести для пользователя отдельную статистику или нет. Чтобы была возможность вести статистику по каждому пользователю отдельно, уровень подробности общей статистики должен быть установлен в `high`;
  - `name: name1` – имя пользователя (используется, в основном, в веб-интерфейсе);
  - `aliases:`, `groups:`, `rules:`, `custom:` – информация об алиасах, группах, Правилах и прочих настройках пользователя.

- **groups-info**

  Выводит всю информацию о группе пользователей. С каждой группой связан такой же набор настроек, что и для пользователя (пустая строка `client-id`, название группы, активна ли группа, Правила группы, вести ли для нее отдельную статистику, список пользователей, входящих в группу, а также дополнительная служебная информация). Формат вывода этой информации следующий:

```
[client-id1/]group A=0|1 S=0|1
emails:
email1
email2
...
custom:
tag1: info1..
tag2: info2..
...
```

Команды интерфейса интерактивного управления для управления пользователями и группами

Управление пользователями, группами пользователей и алиасами осуществляется с помощью специальных команд. В командах используются следующие общие понятия:

- **email** – почтовый адрес пользователя (в соответствии с RFC 5322). Он может быть заключен в угловые скобки (`<>`). Также его можно заключать в одинарные кавычки (`'`). Длина его не может превышать 1024 байт.
- **client-email** – пара значений `[client-id/]`**email**, где `client-id` для **Dr.Web**

MailD всегда является пустым.

- **emails-list** – список **client-email**, разделенных пробелами.
- **group** – имя группы, заключенное в одинарные кавычки. Если в подстроке нет пробелов, то окружающие кавычки можно опустить. Если кавычки присутствуют, то когда в имени встречается символ ', то перед ним должен ставиться повторный символ '. Имя группы не может превышать 1024 байт.
- **client-group** – пара значений [client-id/]**group**, где client-id для **Dr.Web MailD** всегда является пустым.
- **ext-client-group** = [client-id/]**group** | client-id/ - аналог **client-group**, где client-id для **Dr.Web MailD** всегда является пустым.
- **group-list** – список **client-group**, разделенных пробелами.
- **ext-group-list** – список **ext-client-group**, разделенных пробелами.
- **RULE** – Правило обработки писем. Если в значении параметра встречается запятая и она не заключена в кавычки, то перед ней надо ставить "\" – если параметр не разбивается запятыми (т.е. может иметь только одно значение, а не несколько значений, перечисленных через запятую), и "\\\" - если параметр разбивается запятыми (т.е. необходимо экранировать дважды).

  Примеры:

  ```
  true cont headersfilter/RejectCondition = FileName = \"\.e\\\,e\"\, FileName = \"\.com\", headersfilter/RejectPartCondition = FileName = \"\.e\\\,e\"\, FileName = \"\.com\"
  true cont vaderetro/action = discard\, quarantine
  ```

- **tag** – произвольная строка, состоящая только из символов [a-zA-Z0-9_-]. Она является тегом для поиска произвольной информации, связанной с пользователем или группой. Для Web-интерфейса устанавливается в значение web.
- **info** – представляет собой всю строку (вплоть до перевода строки) - т.е. не может содержать внутри себя переводы строки или нулевые символы.
- **settings** – набор настроек для объекта (пользователя или группы). Можно задавать в виде пар имя_параметра=значение. Параметры должны быть разделены пробелами.

  Сейчас доступны следующие параметры:

  - A (active) – может принимать значение 0 (не активирован) или 1 (активирован). Если объект не активирован, то все Правила, связанные с ним, не учитываются. По умолчанию (если параметр не указан) объект считается активным.
  - S (stat) – контролирует ведение статистики для объекта. Может принимать значение 0 (не активирован) или 1 (активирован). Деактивация параметра означает только прекращение ведения статистики для объекта – при этом, если для объекта уже есть статистика в БД, то к ней по-прежнему есть доступ и она не удаляется. По умолчанию сбор статистики для объекта ведется.
  - N (name) – расширенное имя пользователя (для групп данный параметр игнорируется). Может быть заключено в одинарные кавычки так же, как и **group**. Если параметр не указан, то имя пользователя устанавливается пустым. Максимальная длина имени составляет 1000 байт.

    Примеры:

    ```
    S=1 A=0 N='Some user'
    S=0
    ```

Пожалуйста, обратите внимание, что с целью поддержки порядка следования групп для конкретного **client-email**, управление осуществляется набором групп для **client-email**, а не набором **client-email** для группы.

Команды для управления пользователями

При работе с управляющим сокетом "пользователем" считается каждый отдельный e-mail адрес, внесенный в систему. Управлять адресами можно с помощью следующих команд:

- **email-set** client-email [settings] – создание или обновление адреса email, заданного в client-email. Если адрес не существует, то он будет создан. Если в settings указаны не все настройки, то для отсутствующих настроек будут установлены значения по умолчанию. Если для адреса задан алиас, то при обновлении можно указывать именно его в качестве client-email.
- **email-remove** client-email – удаление адреса email, заданного в client-email. Также происходит удаление пользователя из всех групп, в которые он входил. Если такого адреса не существовало или он является алиасом, то выводится ошибка.
- **email-rename** client-email email – изменение основного адреса пользователя, указанного в первом параметре, на адрес, указанный во втором параметре. Если адреса из первого параметра не существует, или он является алиасом, или адрес с новым именем уже существует, то выводится ошибка и никакие действия не выполняются.
- **email-set-groups** client-email [list-of-groups] – задание списка групп, в которые входит адрес client-email. Порядок групп имеет значение (больший приоритет имеют настройки из групп в конце списка). Если list-of-groups пустой, то весь список групп для адреса client-email очищается. В списке list-of-groups группы разделяются пробелами. Если client-email или какая-либо из групп в списке не существуют, то выводится ошибка и операция не выполняется. Если одна и та же группа встречается в списке два раза, то выводится ошибка. Если client-email является алиасом, то обновляется оригинальный получатель. Если для группы в list-of-groups указан client-id, то он должен совпадать с client-id из адреса client-email, в противном случае выводится ошибка. Если в алиасе из list-of-groups client-id не указан, то он принимается равным client-id, указанному в client-email.
- **email-get-groups** emails-list – получения списка групп для всех адресов из списка emails-list. Если какой-либо адрес из списка отсутствует, то выводится ошибка, но операция продолжается. Если client-email является алиасом, то выводится информация для оригинального получателя.

  Формат вывода:

  ```
  client-id/email1: group1 group2 group3 ...
  client-id/email2: group21 group22 group23 ...
  ```

  Здесь groupN может быть заключен в одинарные кавычки, если в имени группы используются пробелы.
- **email-get-rules** emails-list – получение Правил для всех адресов из списка emails-list. Если какого либо адреса в списке не существует, то выводится ошибка, но операция продолжается. Если передан алиас, то выводятся настройки для оригинального получателя. Для каждого несуществующего адреса выводится ошибка.

  Формат вывода:

  ```
  [client-id1/]email1
  1: rule1
  2: rule2
  ...
  [client-id2/]email2
  1: rule21
  2: rule22
  ...
  ```
- **email-insert-rule** client-email index RULE – вставка нового Правила перед Правилом с порядковым номером index для адреса email, заданного в client-email. Если email не существует, то выводится ошибка. Нумерация (index) начинается с 1. Если

значение `index` больше максимального числа Правил для указанного `email`, то новое Правило `RULE` добавляется в конец списка Правил. При этом ему присваивается `index` по порядку (т.е. если для `email` задано всего два Правила, то при попытке добавить новое правило с `index`, равным `10`, правило добавится в конец списка с `index`, равным `3`). Если `index ≤ 0`, то выводится ошибка. Если `RULE` пустое (т.е. Правило не указано), то выводится ошибка. После успешной модификации выводятся Правила для данной группы в формате вывода **`email-get-rules`**.

- **`email-remove-rule`** `client-email index` – удаление Правила с порядковым номером `index` для адреса `email`, заданного в `client-email`. Нумерация (`index`) начинается с `1`. Если `client-email` не существует, то выводится ошибка. Если значение `index` больше максимального числа Правил для указанного `email` или `index ≤ 0`, то выводится ошибка. Если передан алиас, то обновляются настройки для оригинального адреса. После успешной модификации выводятся Правила в формате вывода **`email-get-rules`**.
- **`email-get-custom`** `-|tag emails-list` – получение информации с тегом `tag`, связанной с каждым из пользователей, перечисленных в `emails-list`. Если какого-либо адреса в списке не существует, то выводится ошибка, но операция продолжается. Если информации, связанной с тегом `tag`, не существует, то выводится пустая строка. Информация по каждому адресу разделяется переводом строки. Если вместо тега указан символ "-", выводится информация по всем тегам.

  Формат вывода:

  ```
  [client-id1/]email1
  tag: info..
  [client-id2/]email2
  tag2: info2..
  ```

- **`email-set-custom`** `tag client-email [info]` – установка текста `info`, связанного с тегом `tag` для пользователя `client-email`. Если пользователь не найден, то выводится ошибка. Если `info` не указан, то тег со всей информацией, связанной с ним, удаляется.
- **`email-info`** `emails-list` – получение полной информации по всем адресам из списка `emails-list`. Если какого-либо адреса в списке не существует, то выводится ошибка, но операция продолжается. Правила для адреса выводятся в скомпилированном виде для всех групп и личных настроек адреса. Для алиаса информация по группам и настройкам берется из оригинального адреса. Настройки Правил выводятся в следующем порядке: сначала пользовательские настройки, затем настройки групп в порядке, обратном порядку следования этих групп. При компиляции Правил учитывается настройка активности групп и пользователя.

Формат вывода:

```
[client-id1/]email1 A=active1 S=stat1
name: name1
aliases: alias1 alias2 ..
groups: group1 group2
rules:
1: rule11
2: rule12
...
 custom:
tag1: info1..
tag2: info2..
...
[client-id2/]email2 A=active2 S=stat2
name: name2
aliases: alias12 alias22 .. | alias for email2
groups: group3
rules:
1: rule21
2: rule22
...
custom:
tag21: info21..
tag22: info22..
...
```

Здесь `groupN` может быть заключен в одинарные кавычки, если в имени группы используются пробелы.

Формат вывода для алиаса:

```
[client-id1/]email1
aliases: alias for email
```

- **email-search** `[range:START/NUMBER] [email:part-of-email] [name:'part-of-name'] [ignore:alias|nonalias]` – поиск по адресу или части адреса. Выводит адреса, начиная со `START` (нумерация начинается с `0`), и в количестве `NUMBER` элементов. Если `START` и `NUMBER` не указаны, то выводятся все найденные адреса. Если `START` или `NUMBER` отрицательные, то выводится ошибка. Если значения `START` или `NUMBER` превышают количество найденных адресов, то их значения считаются не ограниченными (соответственно, для "неограниченного" `START` выводятся адреса с самого первого в списке, а для "неограниченного" `NUMBER` - все имеющиеся в списке адреса).
  - `part-of-email` – подстрока в почтовом адресе или алиасе, по которой производится поиск. Если `part-of-email` не указана, то выводятся все известные адреса и алиасы. Формат вывода совпадает с выводом **email-info**. Уникальный идентификатор пользователя в `part-of-email` должен быть указан полностью.
  - `part-of-name` – подстрока в имени пользователя (если в имени встречается одинарная кавычка `'`, то перед ней должен ставиться тот же символ `'`; если в подстроке нет пробелов, то окружающие кавычки можно опустить) – выводятся только те пользователи, имена которых содержат указанную подстроку.
  - `ignore` – определяет, какого типа записи следует игнорировать: `alias` – алиасы (то есть поиск будет проводиться только среди обычных адресов), `nonalias` – обычные адреса (то есть поиск будет проводиться только среди алиасов).

  Если задано одновременно `email` и `name`, то выводятся только пользователи, удовлетворяющие обоим ограничениям. Так как для алиасов не хранится имя пользователя, то использование в поиске одновременно подстроки для алиаса и имени пользователя является бессмысленным.
- **email-count** `[range:START/NUMBER] [email:part-of-email] [name:'part-of-name'] [ignore:alias|nonalias]` – обработка осуществляется аналогично **email-**

search, но выводится только число найденных адресов.

Команды для управления алиасами

Алиасами можно управлять с помощью следующих команд:

- **aliases-get** `emails-list` – получение списка алиасов для всех адресов из списка `emails-list`. Если в `emails-list` есть несуществующие адреса или другие алиасы, то для них выводится ошибка, но операция продолжается. Если один и тот же адрес встречается два раза, то выводится ошибка.

  Формат вывода:

  ```
  [client-id1/]email1: alias1 alias2 alias3 ...
  [client-id2/]email2: alias21 alias22 alias23 ...
  ```

- **aliases-set** `client-email [emails-list]` – задание списка алиасов для адреса `email`, заданного в `client-email`. Если `client-email` не существует или сам является алиасом, то выводится ошибка. Если список `emails-list` не указан, то все алиасы, связанные с `client-email`, удаляются. Если в списке есть хоть один адрес, который уже является зарегистрированным адресом или алиасом для другого адреса, то выводится ошибка и операция прекращается. Если для адреса в `emails-list` указан `client-id`, то он должен совпадать с `client-id` из адреса `client-email`, в противном случае выводится ошибка. Если в алиасе из `emails-list client-id` не указан, то он принимается равным `client-id`, указанному в `client-email`.

Команды для управления группами

Управлять группами пользователей можно с помощью следующих команд:

- **groups-set** `client-group [settings]` – создание или обновление группы с именем `group`, заданной в `client-group`. Если группа не существует, то она будет создана. Если в `settings` указаны не все настройки, то для отсутствующих будет установлено значение по умолчанию.
- **groups-remove** `client-group` – удаление группы с именем `group`, заданной в `client-group`. Если заданная группа не существует, то выводится ошибка. Для каждого из пользователей, входивших в удаляемую группу, данная группа будет удалена из списка групп, в которые пользователь входит.
- **groups-rename** `client-group group` – переименование группы, указанной в качестве первого параметра, с использованием имени, указанного в качестве второго параметра. Если указанной группы не существует или группа с новым именем уже существует, то выводится ошибка и никакие действия не выполняются.
- **groups-get-rules** `[group-list]` – получение Правил для всех групп из списка `group-list`. Если какая-либо группа из `group-list` не существует, то выводится ошибка, но операция продолжается.

  Формат вывода:

  ```
  [client-id1/]group1
  1: rule1
  2: rule2
  ...
  [client-id2/]group2
  1: rule21
  2: rule22
  ...
  ```

- **groups-insert-rule** `client-group index RULE` – вставка нового Правила перед Правилом с порядковым номером `index` для группы с именем `group`, заданной в `client-group`. Если группы с таким именем нет, то выводится ошибка. Нумерация (`index`)

начинается с 1. Если значение index больше максимального числа Правил для указанной группы, то новое Правило RULE добавляется в конец списка Правил. При этом ему присваивается index по порядку.

Пример:

Если для группы задано всего два Правила, то при попытке добавить новое Правило с index, равным 10, правило добавится в конец списка с index, равным 3).

Если index ≤ 0, то выводится ошибка. Если RULE пустое (т.е. Правило не указано), то выводится ошибка. После успешной модификации выводятся Правила для данной группы в формате вывода **groups-get-rules**.

- **groups-remove-rule** client-group index – удаление Правила с порядковым номером index для группы group, заданной в client-group. Нумерация (index) начинается с 1. Если group не существует, то выводится ошибка. Если значение index больше максимального числа Правил для указанной группы или index ≤ 0, то выводится ошибка. После успешной модификации выводятся Правила для данной группы в формате вывода **groups-get-rules**.
- **groups-info** [ext-group-list] – вывод всех пользователей, которые входят в группы из списка ext-group-list, а также информации об активности и произвольной информации. Если какая-либо группа из ext-group-list не существует, то выдается ошибка, но операция продолжается. Если ext-group-list не указано, то выводится информация по всем имеющимся группам. Алиасы в списках адресов не выводятся.

  Формат вывода:

```
[client-id1/]group1 A=active1 S=stat1
emails:
email1
email2
...
custom:
tag1: info1..
tag2: info2..
...
[client-id2/]group2 A=active2 S=stat2
emails:
email21
email22
...
custom:
tag21: info21..
tag22: info22..
...
```

- **groups-count** [ext-group-list] – команда работает аналогично **groups-info**, только выводит число найденных групп.
- **groups-get-custom** -|tag group-list – получение информации с тегом tag, связанной с каждой группой, перечисленной в group-list. Если какая-либо группа из group-list не существует, то выводится ошибка, но операция продолжается. Если информации, связанной с тегом tag не существует, то выводится пустая строка. Информация по каждой группе разделяется переводом строки. Если вместо тега указан символ "-", то выводится информация по всем тегам.

  Формат вывода:

```
[client-id1/]group1
tag: info..
[client-id2/]group2
tag2: info2..
```

- **groups-set-custom** tag client-group [info] – установка текста info, связанного с тегом tag для группы client-group. Если группа не найдена, то выводится ошибка.

Если info не указано, то тег со всей информацией, связанной с ним, удаляется.

Работа с Карантином

Работа с **Карантином** через управляющий сокет осуществляется с помощью специальных команд. В командах используются следующие общие понятия:

- **id** – путь относительно каталога, указанного в значении параметра **Path** секции [Quarantine], к файлу с телом письма.

  Например, если в значении параметра **Path** секции [Quarantine] указано /var/drweb/infected (значение по умолчанию), то путь <id>/drweb/E/00020EBE.maild.xeAX4u ссылается на письмо, тело которого расположено в файле /var/drweb/infected/<id>/drweb/E/00020EBE.maild.xeAX4u.

  Здесь:

  - <id> – строка "def";
  - drweb – подключаемый модуль, заблокировавший письмо (**Drweb** в данном случае). Если письмо заблокировано компонентом **MailD core**, то значение устанавливается в maild. Если письмо помещено в архив, то значение устанавливается в backup.

- **id-like** – то же, что и **id**, только при задании данных идентификаторов можно использовать специальные символы: "%" – ноль или более произвольных символов, "_" – один произвольный символ.

  Пример:

  def/%00014F7F% – все сообщения с номером 00014F7F, сохраненные в **Карантине**;

  def/drweb/% – все сообщения, сохраненные подключаемым модулем **Drweb**.

  Тема письма сохраняется в базе данных в декодированном виде (в кодировке UTF-8), и все управляющие символы (ASCII 0..21 и 127), за исключением табуляции, заменяются на пробел.

Вывод результата выполнения каждой команды заканчивается пустой строкой.

Команды для управления Карантином

Для управления **Карантином** используются следующие команды:

- **quarantine-search** [range:START/NUMBER] [sort:SORT_TYPE] [sender:EMAIL_SUBSTR] [rcpt:EMAIL_SUBSTR]* [period:DATE1[/DATE2]] [size:SIZE] [subject:'SUBJECT_SUBSTR'] [id:id-like] [order:ascent|descent] – поиск сообщений в **Карантине** по заданным критериям. Выводит письма, начиная со START (нумерация начинается с 0), и в количестве NUMBER элементов. Если START и NUMBER не указаны, то выводятся все найденные письма, удовлетворяющие остальным критериям. Значение NUMBER 0 означает вывод всех элементов.

  Используемые параметры:

  - SORT_TYPE – тип сортировки. Возможные значения:
    - date (по умолчанию) – сортировка по дате поступления писем в **Карантин**;
    - size – сортировка по размеру письма;
    - sender – сортировка по адресу отправителя;
    - subject – сортировка по теме письма.
  - EMAIL_SUBSTR – подстрока для поиска в полях rcpt или sender.
  - period – период, за который будут выводиться письма. Если он не указан, то выводятся письма за весь период.
  - DATE1 – выводятся письма, которые попали в **Карантин** только после этого времени

(включительно).

- DATE2 – верхняя граница времени попадания письма в **Карантин** (включительно). Формат DATE соответствует ISO формату – YYYYMMDDTHHMMSS, где T – разделитель между временем и датой. Время задается и выводится как локальное время для хоста, на котором работает **Dr.Web MailD**.
- SIZE – задает максимальный размер в байтах и возвращает только письма, размер которых превышает указанное значение. При значении 0 размер не ограничен.
- SUBJECT_SUBSTR – заключенная в кавычки подстрока в оригинальной теме письма (т.е. до модификации письма компонентами программного комплекса). Если в подстроке нет пробелов, то окружающие кавычки можно опустить. Если кавычки присутствуют, то когда в имени встречается ', то перед ним должен ставиться повторный символ '.
- order – порядок, в котором возвращаются результаты (ascent – возрастающий, descent – убывающий). Значение по умолчанию: descent.

Если в каком-либо из параметров допущена ошибка, то команда поиска не будет исполнена. Если задано несколько шаблонов получателей, то выдаются только письма, в которых содержатся все шаблоны (аналогично действию логического оператора AND). Для всех параметров, кроме rcpt, будет использоваться последнее заданное в командной строке значение, а для rcpt набор получателей будет увеличиваться с каждым новым введенным значением.

Формат вывода:

```
N. id SENDER RCTPS
SIZE DATE SUBJECT
BLOCK_OBJECT1
BLOCK_OBJECT2
...
```

где:

- N – порядковый номер найденного сообщения;
- SENDER – отправитель письма из конверта;
- RCPTS – получатели письма из конверта;
- SUBJECT – тема письма. Выводится в кодировке UTF8;
- SIZE – размер письма в байтах;
- DATE – дата помещения письма в **Карантин**;
- BLOCK_OBJECTN – блокирующий объект для этого письма.

Примеры:

```
# quarantine-search
```

Возвращает список всех писем в **Карантине**, начиная с самых новых.

```
# quarantine-search range:45/15 id:def/drweb/%
```

Возвращает 15 самых новых сообщений в **Карантине**, пропустив первые 45 для подключаемого модуля **Drweb.**

```
# quarantine-search rcpt:vasya@pupkin.com
```

Возвращает все письма в **Карантине**, начиная с самых новых, среди получателей которых есть vasya@pupkin.com.

```
# quarantine-search sort:size sender:
period:20090101T100001/20090102T100000 size:5242880 id:def/vaderetro/%
```

Выводятся в порядке уменьшения размера все сообщения, полученные для подключаемого модуля **Vaderetro** 1 января 2009 года с 10 утра до 10 утра следующего дня, и размер которых больше 5 мегабайт.

Пример вывода:

```
# quarantine-search
0.           def/drweb/9/00021569.maild.BMED3y           <ai@drweb.com>
<alias_ai81@drweb.com>
829 20091117T102126 [EICAR] test2
EICAR Test File (NOT a Virus!)
1.           def/backup/9/00021569.maild.3PLb8e          <ai@drweb.com>
<alias_ai81@drweb.com>
828 20091117T100213 [EICAR] test
```

- **quarantine-count** [range:START/NUMBER] [sort:SORT_TYPE] [sender:EMAIL_SUBSTR] [rcpt:EMAIL_SUBSTR]* [period:DATE1[/DATE2]] [size:SIZE] [subject:'SUBJECT_SUBSTR'] [id:id-like] [order:ascent|descent] – данная команда работает аналогично команде **quarantine-search**, только вместо самих сообщений выводится общее число найденных сообщений.

  Пример вывода:

  ```
  # quarantine-count
  234
  ```

- **quarantine-remove** id-like [part-of-email1, part-of-email2, ..] – удаляет заданных получателей (ищутся как подстрока) part-of-email1, part-of-email2, ... из конвертов писем, идентификатор которых совпадает с id-like (в конверте должны присутствовать все заданные получатели). Если у письма получателей не осталось или их список для удаления не указан, то письмо целиком удаляется из **Карантина**.

  Примеры:

  ```
  # quarantine-remove %/backup/% drweb.com>
  ```

  Из **Карантина** и бэкапа удаляются письма всех получателей, адреса которых заканчиваются на drweb.com.

  ```
  # quarantine-remove % <foo@dwreb.com> <foo2@dwreb.com>
  ```

  Из **Карантина** и бэкапа удаляются все письма, в получателях которых присутствуют одновременно foo@dwreb.com и foo2@dwreb.com.

- **quarantine-limits** – вывод текущих ограничений, установленных для **Карантина**.

  Формат вывода:

  ```
  client-id: NUMBER/MAX-NUMBER SIZE/MAX-SIZE
  ...
  total: NUMBER/MAX-NUMBER SIZE/MAX-SIZE
  ```

  где:

  - NUMBER/MAX-NUMBER – текущее/максимальное число сообщений. Если максимальное значение не установлено, то выводится 0.
  - SIZE/MAX-SIZE – текущий/максимальный размер сообщений в **Карантине** (в байтах). Если максимальное значение не установлено, то выводится 0.
  - client-id – строка "def".
  - total – информация по всей базе данных.

- **quarantine-send** id-like [email1 email2 ...] – отправляет сообщения из **Карантина** заданным получателям (email1 email2 ...). Если получатели не заданы, то письма отправляются оригинальным получателям из конверта. Результат отправки письма выводится в следующем формате:

  ```
  RES in sending (to RCPTS_LIST): id
  ```

  где:

  - RCPTS_LIST – фактический список получателей письма.
  - RES – OK или ERROR, в зависимости от результата отправки.
  - id – путь к файлу с телом письма.

  Пример вывода:

  ```
  OK in sending (to <ai@drweb.com> <as@sd>): def/
  backup/6/00004DD6.maild.VQ80Ro
  OK in sending (to <ai@drweb.com> <as@sd>): def/
  backup/6/00004DC6.maild.PWfqe3
  ```

- **quarantine-add** id from rcpt1 rcpt2... – добавляет заданный файл в **Карантин**. Здесь from – отправитель письма, rcptN – получатели. Адреса могут быть заключены в угловые скобки <>. Если файла с указанным id не существует, то выводится ошибка.

Получение статистической информации

С помощью командного интерфейса управляющего сокета можно получать статистику по работе **Dr.Web для почтовых серверов UNIX** для пользователей и групп. Получение статистической информации осуществляется с помощью специальных команд. В командах используются следующие общие понятия:

- **email** – почтовый адрес пользователя (в соответствии с RFC5322). Он может быть заключен в угловые скобки (<>). Также его можно заключать в одинарные кавычки (''). Длина его не может превышать 1024 байт.
- **client-email** – пара значений [client-id/]email, где client-id для **Dr.Web MailD** всегда является пустым.
- **group** – имя группы, заключенное в одинарные кавычки. Если в подстроке нет пробелов, то окружающие кавычки можно опустить. Если кавычки присутствуют, то когда в имени встречается символ ', то перед ним должен ставиться повторный символ '. Имя группы не может превышать 1024 байт.
- **client-group** – пара значений [client-id/]group, где client-id для **Dr.Web MailD** всегда является пустым.

При работе со статистикой следует учитывать, что статистика для пользователей и групп сохраняется непосредственно в соответствующие записи внутренней базы данных. При этом, если при попытке сохранить следующую информацию окажется, что запись находится в базе данных больше пяти минут, то создается новая запись, в которую и происходят дальнейшие сохранения.

Так как приведенные ниже команды работают только со внутренней базой данных, то из этого следует, что последняя по времени запись для пользователей и групп содержит статистику от момента своего создания до текущего момента.

Команды для работы со статистикой

У команд для работы со статистикой есть ряд общих параметров:

- **period** = period:DATE1[/DATE2] – выводить статистику только за указанный период, включая границы временного интервала.

  Здесь:

  - DATE1 – нижняя граница периода. Формат вывода описан ниже. Формат времени описан ниже.
  - DATE2 – верхняя граница периода. Если данный параметр не указан, то принимается текущее время. Формат времени описан ниже.

  Если период не указан, то выводится вся доступная статистика.
- **ignore** = ignore:total|block – фильтрация выводимой статистики.
  - **total** – не выводить общую статистику по проверенным сообщениям.
  - **block** – не выводить статистику по заблокированным сообщениям. Если данный параметр не указан, то выводятся все виды статистики.
- **plugin** = plugin:name – выводить информацию только для заданного подключаемого модуля, где name – имя модуля, по работе которого необходимо предоставить статистику. Если параметр **plugin** не указан, то выводится информация по всем подключаемым модулям. Если указан несуществующий модуль, то выводится ошибка и выполнение команды прерывается. Если указано *, то выводится только общая статистика.

Если указано несколько одинаковых параметров, то статистика будет выводиться только по последнему указанному параметру.

Доступны следующие команды:

stat-client – [period] [ignore] [plugin]

Команда предназначена для получения статистики.

Факультативные параметры могут идти в произвольном порядке.

stat-group client-group [period] [ignore] [plugin]

Команда предназначена для получения статистики для группы client-group.

- Если группа не существует, то выводится ошибка и выполнение команды прерывается.

Факультативные параметры могут идти в произвольном порядке.

stat-email client-email [period] [ignore] [plugin]

Команда предназначена для получения статистики для конкретного пользователя client-email.

- Если для указанного адреса нет статистики (например, адрес указан неверно), то выводится пустая строка.
- Если адрес является алиасом, то выводится статистика по основному адресу.

Факультативные параметры могут идти в произвольном порядке.

stat-remove-client – [period] [ignore] [plugin]

Команда предназначена для удаления статистики .

По результатам выводится количество удаленных записей.

stat-remove-group client-group [period] [ignore] [plugin]

Команда предназначена для удаления статистики для группы client-group.

В результате выводится количество удаленных записей.

stat-remove-email client-email [period] [ignore] [plugin]

Команда предназначена для удаления статистики для конкретного пользователя client-email.

В результате выводится количество удаленных записей.

remove-old-stat [time]

Команда предназначена для удаления всей статистики по всем группам и пользователям, если она старше времени, указанного в time (тип {time}).

Если значение не указано, то будет удалена вся статистика **старше 24 часов**.

dump-cache-stat

Команда предназначена для сброса внутреннего кэша общей статистики во внутреннюю базу данных.

Данная функция периодически вызывается самим комплексом. Также она вызывается при получении сигнала HUP и остановке комплекса.

Вывод статистики

Статистика по работе каждого из подключаемых модулей выводится в определенном формате. Для каждого подключаемого модуля вывод состоит из двух частей:

1. общая статистика по проверенным сообщениям:

```
PLUGIN DATE [P][R][D][T][Q][RE][N][C][S][U][F][I][DI][DM][DSV][DC][DD][DSK][DAR][DE]
[DTA][DTD][DTJ][DTR][DTH][PS][RS][DS][TS][QS][RES][NS][CS][SS][US][FS][IS][WT]...
```

2. статистика по заблокированным сообщениям:

```
PLUGIN DATE FROM|- IP|- 'BLOCK1' TYPE1 'BLOCK2' TYPE2 ...
```

Здесь:

- PLUGIN – название подключаемого модуля, по которому выводится статистика. Если указано *, то выводится общая статистика по всему комплексу (например, здесь учитываются сообщения, которые не были пропущены ни через один подключаемый модуль).
- DATE – время, когда была создана запись. Для общей статистики по проверенным сообщениям обозначает начало периода, за который была сохранена статистика, при этом конец периода - это начало следующей записи, или, если следующей записи нет, то это начало периода плюс 5 минут. Формат соответствует ISO формату - YYYYMMDDTHHMMSS, где T – разделитель между временем и датой. Время задается и выводится как локальное время для хоста, на котором работает **Dr.Web для почтовых серверов UNIX**.

Следующие значения до WT включительно выводятся в формате **NAME**=VAL, где **NAME** – собственно имя значения (P, PS...), а VAL – его числовое значение. Если какое-либо из этих значений не указано, то оно считается равным нулю.

- P/PS – <количество>/<размер в байтах> сообщений, для которых было выполнено действие pass;
- R/RS – <количество>/<размер в байтах> сообщений, для которых было выполнено действие reject;

- D/DS – <количество>/<размер в байтах> сообщений, для которых было выполнено действие discard;
- T/TS – <количество>/<размер в байтах> сообщений, для которых было выполнено действие tempfail;
- Q/QS – <количество>/<размер в байтах> сообщений, для которых было выполнено действие quarantine;
- RE/RES – <количество>/<размер в байтах> сообщений, для которых было выполнено действие redirect;
- N/NS – <количество>/<размер в байтах> сообщений, для которых было выполнено действие notify;
- C/CS – <количество>/<размер в байтах> чистых сообщений;
- S/SS – <количество>/<размер в байтах> сообщений, помеченных как спам;
- U/US – <количество>/<размер в байтах> сообщений, помеченных как безусловный спам;
- F/FS – <количество>/<размер в байтах> сообщений, заблокированных фильтром;
- I/IS – <количество>/<размер в байтах> сообщений, содержащих вирусы;
- DI – количество инфицированных вложений;
- DM – количество вложений, зараженных модификацией известного вируса;
- DSV – количество вложений, зараженных неизвестным вирусом;
- DC – количество излеченных вложений;
- DD – количество вложений, которые были удалены;
- DSK – количество вложений, которые по различным причинам были пропущены без антивирусной проверки;
- DAR – количество вложений, которые были пропущены без антивирусной проверки из-за ограничений на архивы;
- DE – количество вложений, при проверке которых произошла ошибка;
- DTA – количество вложений, содержащих программы для показа рекламы;
- DTD – количество вложений, содержащих программы дозвона;
- DTJ – количество вложений, содержащих программы-шутки;
- DTR – количество вложений, содержащих потенциально опасные программы;
- DTH – количество вложений, содержащих программы, предназначенные для получения несанкционированного доступа к компьютерным системам;
- WT – время в миллисекундах, потраченное подключаемым модулем на обработку сообщений.

Для заблокированных писем выводится список, состоящий из следующих полей:

- BLOCK[12..] – название блокирующего объекта (например, вируса). Оно заключается в кавычки по аналогии с тем, как это делается для групп (см. выше).
- TYPE[12..] – тип блокирующего объекта. Названия берется из **NAME**, описанного выше. Доступные значения: DI–DTH, F, S, U.
- FROM – отправитель письма из конверта.
- IP – IP-адрес отправителя письма.

Примеры запросов статистики

1. Запрос всей накопленной статистики для всех обработанных писем (по всем подключаемым модулям):

```
> stat-client -
```

Формат команд соответствует описанию, статистика выдается в формате, указанном выше.

2. Запрос статистики по незаблокированным письмам (по всем подключаемым модулям):

```
> stat-client - ignore:block
```

Пример вывода статистики:

```
* 20120307T145754 P=1 C=1 PS=194562 CS=194562
headersfilter 20120311T163250 R=4 F=4 RS=39848 FS=39848
headersfilter 20120311T164250 R=1 F=1 RS=539 FS=539
headersfilter 20120311T165400 P=1 F=1 PS=539 FS=539 WT=32
drweb 20120311T165400 R=1 Q=1 C=1 DE=1 RS=539 QS=539 CS=539 WT=13
headersfilter 20120311T165727 P=1 F=1 PS=539 FS=539 WT=32
drweb 20120311T165727 P=1 C=1 DE=1 PS=539 CS=539 WT=11
* 20120311T165953 P=1 U=1 F=1 DE=1 PS=539 US=539 FS=539 WT=51
vaderetro 20120311T165953 P=1 U=1 PS=539 US=539 WT=5
modifier 20120311T165953 P=1 C=1 PS=539 CS=539 WT=3
headersfilter 20120311T170453 P=1 F=1 PS=539 FS=539
drweb 20120311T170453 P=1 C=1 DE=1 PS=539 CS=539 WT=11
* 20120311T171208 P=1 U=1 F=1 PS=539 US=539 FS=539 WT=52
vaderetro 20120311T171208 P=1 U=1 PS=539 US=539 WT=5
modifier 20120311T171208 P=1 C=1 PS=539 CS=539 WT=3
headersfilter 20120311T171412 P=1 F=1 PS=539 FS=539 WT=33
drweb 20120311T171412 P=1 C=1 DE=1 PS=539 CS=539 WT=5
```

3. Запрос статистики только по заблокированным письмам (по всем подключаемым модулям):

```
> stat-client - ignore:total
drweb 20120406T194038 root@testlab-solaris.i.drweb.ru 10.3.0.91 'No such
file or directory' DE - DE 'No such file or directory' DE - DE Trojan.Grab
DI 'No such file or directory' DE - DE
drweb 20120406T194039 root@testlab-solaris.i.drweb.ru 10.3.0.91 'No such
file or directory' DE - DE 'No such file or directory' DE - DE Trojan.Grab
DI 'No such file or directory' DE - DE
drweb 20120406T194040 root@testlab-solaris.i.drweb.ru 10.3.0.91 'No such
file or directory' DE - DE 'No such file or directory' DE - DE Trojan.Grab
DI 'No such file or directory' DE - DE
```

4. Запрос статистики только по письмам, заблокированным антивирусным модулем **Drweb**:

```
> stat-client - ignore:total plugin:drweb
```

5. Запрос статистики только по письмам, пропущенным антивирусным модулем **Drweb**:

```
> stat-client - ignore:block plugin:drweb
```

Проверка генерации уведомлений

Для проверки генерации компонентом **Notifier** уведомлений на основании шаблонов в интерфейсе интерактивного управления доступна специальная команда **notify**.

Формат команды

Команда имеет следующий синтаксис:

```
notify type [mode] [-] [options]
```

здесь:

- `type` – Тип уведомления. Совпадает с именем файла используемого шаблона. Разрешается использовать любой из типов, кроме `report.msg`.
- `mode` – Режим работы с уведомлением. Существуют следующие варианты:
  - `show` – сгенерировать письмо на основе шаблона и вывести его текст.

Формат вывода команды:

```
SIZE <FROM> <RCPT1> <RCPT2>...
BODY
```

где: `SIZE` – размер `BODY` в байтах. Если `BODY` не заканчивается переводом строки, то он добавляется (но в `SIZE` не включается); `FROM` – отправитель письма (из конверта); `RCPT` – получатели письма (из конверта); `BODY` – тело самого письма. Механизм определения отправителя и получателей письма описан ниже.

- `sync` – сгенерировать письмо на основе шаблона и отправить его через компонент **Sender** в синхронном режиме (т.е. **Sender** должен выполнить отправку письма, прежде чем вернуть результат). Если используемый **Sender** не поддерживает данный режим, то отправка будет произведена в асинхронном режиме (см. ниже). По результатам работы команды выводится строка об успешном или неудачном отправлении сгенерированного уведомления.
- `async` – сгенерировать письмо на основе шаблона и отправить его через компонент **Sender** в асинхронном режиме (т.е. **Sender** должен принять письмо на обработку, но фактическое отправление может сделать позднее). Если используемых **Sender** не поддерживает данный режим, то отправка будет произведена в синхронном режиме (см. выше). Компонент **Sender** должен поддерживать хотя бы один из режимов. По результатам работы команды выводится строка об успешном или неудачном отправлении.

Если режим `mode` не указан, то он устанавливается в `show`.

- `client-id` – идентификатор **Клиента**, в рамках которого обрабатывается уведомление. Если указан `'-'`, то идентификатор принимается пустым. Таким образом с помощью команды **`notify`** можно работать только с одним **Клиентом**.
- `options` – Список инициализации макросов для шаблона (присвоенные значения будут использоваться при генерации сообщения). Формат списка – набор пар вида:

  ```
  NAME=VAL
  ```

  где `NAME` – имя макроса (без окружающих $), а `VAL` – присвоенное значение, заключенное в одинарные кавычки. Регистронезависимый. Если в VAL нет пробельных символов, то окружающие кавычки можно опустить. Если кавычки присутствуют и в VAL встречается символ ', то перед ним должен ставиться повторный символ '.

  Между NAME, '=' и VAL пробельных символов быть не может. Если указано несколько пар NAME=VAL с одинаковым NAME, то для макроса NAME устанавливается списковый тип и он в шаблоне обрабатывается соответствующим образом. Если в шаблоне предполагается, что макрос не имеет спискового типа, а он указан в команде несколько раз, то либо берется первое значение, либо все значения объединяются через запятую (что в большинстве случаев неверно).

После вывода команды всегда добавляется пустая строка.

Инициализация макросов

В списке инициализации могут использоваться любые макросы из списка. Для всех макросов, кроме описанных ниже, если они не указаны, будет установлено значение по умолчанию.

Не будут автоматически установлены значения макросов, используемых в отчетах (макросы с именем `$RP_*$`). Эти макросы требуется явно инициализировать в команде. Если какой-либо макрос, необходимый для генерации письма из шаблона, не был указан, то в журнале компонента **Notifier** будет выведена соответствующая ошибка.

В качестве параметров для инициализации, наряду с макросами, можно также использовать следующие специальные значения:

- `_FROM` – задает отправителя сообщения в конверте (в том числе используется для поиска

параметров в Правилах, использующих условие `sender:`) Для шаблонов DSN отправитель всегда сбрасывается. Механизм определения отправителя письма (для конверта) описан ниже. Если задано несколько значений, то используется первое.

- `_RCPTS` – задает получателей сообщения (в том числе используется и для поиска параметров в Правилах, использующих условие `rcpt:`) Имеет списковый тип. Механизм определения получателей (для конверта) описан ниже.
- `_BLOCK` – задает названия вредоносных объектов, заблокировавших виртуальное сообщение, для которого генерируется уведомление. Этот параметр имеет списковый тип и используется для поиска параметров в Правилах, использующих условие `block:`.
- `_SIZE` – задает размер виртуального сообщения, для которого генерируется уведомление, в байтах . Используется для поиска параметров в Правилах, использующих условие `size:`.
- `_SCORE` – задает счет сообщения, для которого генерируется уведомление. Используется для поиска параметров в Правилах, использующих условие `score:`.

Механизм определения адресатов

Если заданы специальные значения `_FROM` или `_RCPTS`, то отправитель и получатели в конверте определяются только на их основе (кроме шаблонов DSN, где отправитель всегда устанавливается пустым). В противном случае используются следующие правила определения адресатов:

Для шаблонов `welcome_user.msg` и `password_user.msg`:

- адрес отправителя устанавливается на основе параметра **`AdminMail`** из Правил;
- адреса получателей – на основе макроса `$RCPTS$` (поэтому он обязательно должен быть инициализирован, в противном случае возникнет ошибка).

Для шаблонов `welcome_client.msg` и `password_client.msg`:

- адрес отправителя устанавливается на основе первого значения параметра **`AdminMail`** из конфигурации, без учета сработавших Правил.
- адреса получателей устанавливаются на основе значения параметра **`AdminMail`** из сработавших Правил.

Для шаблонов DSN (`dsn.msg` или `dsn_for_exchange.msg`):

- адрес отправителя всегда пустой
- если задан параметр `_FROM`, то адрес получателя устанавливается в его первое значение, в противном случае адреса получателей берутся из макроса `$RCPTS$`. Если не задан параметр `_FROM` и не инициализирован макрос `$RCPTS$`, то возникнет ошибка.

Остальные шаблоны:

- адрес отправитель устанавливается в значение параметра **`FilterMail`** из сработавших Правил.
- адреса получателей – на основе макроса `$RCPTS$` (поэтому он обязательно должен быть инициализирован, в противном случае возникнет ошибка).

Примеры использования команды

Показать текст письма, генерируемого как DSN для получателей `ai@drweb.com` и `test@drweb.com`

```
notify dsn.msg show - _RCPTS=ai@drweb.com FULLHEADERS='From: <fake>'
_RCPTS='test@drweb.com'
```

В результате выводится текст сформированного письма.

Сформировать и отправить в синхронном режиме письмо, сгенерированное на основе шаблона `admin_virus.msg`

```
notify admin_virus.msg sync - RCPTS=test FULLHEADERS='From: <fake>'
```

Утилиты

Для удобства настройки и управления работой комплексного компонента **Dr.Web MailD**, в состав комплекса **Dr.Web для почтовых серверов UNIX** входит набор дополнительных утилит:

- Утилита **drweb-qp** – предназначена для сопряжения **Карантина** с **DBI**. Не предназначена для запуска вручную
- Утилита **drweb-qcontrol** – предназначена для работы с содержимым **Карантина**;
- Утилита **drweb-lookup** – предназначена для проверки корректности выражений, используемых в Lookup;
- Утилита **drweb-inject** – предназначена для отправки вручную писем (в т.ч. «потерянных» в следствие ошибок, возникших при обработке) через компонент **Sender** или установленную почтовую систему.

drweb-qcontrol: Управление Карантином

Утилита **drweb-qcontrol** предназначена для управления **Карантином** и осуществления поиска в нем писем. Интерфейс запуска утилиты в большинстве случаев не зависит от того, где хранятся письма в файлах (в каталогах хранилища на диске или в хранилище **DBI**).

Если **Карантин** хранится в каталогах, то для работы утилиты с ним требуется, чтобы модуль **drweb-maild** был предварительно запущен.

Формат запуска утилиты:

```
drweb-qcontrol [параметры] команда [, команда, ...] <идентификаторы>
```

1. Параметры командной строки

Доступны следующие `параметры` командной строки:

Краткий вариант	Расширенный вариант	Аргументы
-h	--help	
Описание: Вывод на консоль краткой справки по параметрам командной строки и завершение работы утилиты		
	--version	
Описание: Вывод на консоль информации о версии и завершение работы утилиты		
-v	--verbose	
Описание: Предписание выводить на консоль всю информацию о действиях, совершаемых утилитой в процессе работы		
-l	--level	<уровень подробности>
Описание: Установка уровня подробности записи в журнал. Возможные значения: Quiet, Error, Alert, Info, Debug		

Краткий вариант	Расширенный вариант	Аргументы
-i	--ipc-level	<уровень подробности>
Описание: Установка уровня подробности записи в журнал для подсистемы IPC (взаимодействие с модулем **drweb-maild**). Возможные значения: Quiet, Error, Alert, Info, Debug		
	--syslogfacility	<метка syslog>
Описание: Установка типа подсистемы, через которую системный сервис **syslog** (если журналирование работы утилиты ведется через него, см. следующий параметр) выдает сообщения о событиях. Возможные значения: Daemon, Mail, Local0, ..., Local7		
	--log-filename	<имя файла>
Описание: Установка имени файла журнала или значение syslog, если журналирование работы утилиты должно осуществляться с помощью системного сервиса **syslog**		
	--sendmail	<путь к файлу>
Описание: Установка пути к исполняемому файлу утилиты отправки писем **drweb-inject** (по умолчанию, если параметр не задан, используется путь %bin_dir%/drweb-inject)		
-s	--socket	<путь к файлу>
Описание: Установка пути к управляющему сокету **Dr.Web MailD** (по умолчанию, если параметр не задан, используется путь %var_dir%/ipc/.ctl)		
	--agent	<путь к файлу>
Описание: Установка пути к сокету компонента **Dr.Web Agent** для получения от него конфигурационной информации (по умолчанию, если параметр не задан, используется путь %var_dir%/ipc/.agent). Если указать ключ без параметра, обращение к **Dr.Web Agent** производиться не будет		
	--timeout	<период времени>
Описание: Установка максимального допустимого времени ожидания ответа от **Dr.Web Agent** при получении конфигурации		

2. Команды

Команды опередяляют, какие действия следует выполнить с письмами, выбранными из **Карантина** по указанным условиям. Список писем, над которыми выполняются действия, определяется через уникальные идентификаторы, в качестве которых используются относительные пути файлов писем, сохраненных в **Карантине**. При задании этих идентификаторов можно использовать специальные символы шаблонов:

- "%" – соответствует последовательности из нуля или более произвольных символов;
- "_" – соответствует ровно одному произвольному символу.

Обратите внимание, что в начале любого идентификатора необходимо указать префикс def/.

Пример:

def/%00014F7F% – все сообщения, помещенные в **Карантин**, номер которых содержит внутри себя последовательность цифр 00014F7F, или совпадает с ней;

def/drweb/% – все сообщения, помещенные в **Карантин** подключаемым модулем **Drweb**.

Идентификаторы файлов, над которыми требуется выполнить операции, берутся из командной строки или из указанных при запуске условий поиска (см. ниже) (при этом, если одновременно указаны условия поиска и непосредственно идентификаторы писем в командной строке, то они объединяются). Если не было указано ни одного условия поиска, и в командной строке не было указано ни одного идентификатора писем, то ожидается ввод идентификатора со стандартного

потока ввода.

Возможные команды:

- `--view` – просмотреть содержимое писем с заданными идентификаторами при помощи утилиты, указанной в переменной окружения `PAGER`. Если в переменной окружения `PAGER` не указано значение, то по умолчанию используется утилита **`cat`**.
- `--send` – отправить все письма с заданными идентификаторами оригинальным получателям. Для отправления используется утилита **`drweb-inject`**.
- `--redirect [list_of_rcpts]` – переслать все письма с заданными идентификаторами на адреса из заданного списка `list_of_rcpts`. Для отправления используется утилита **`drweb-inject`**.
- `--remove` – удалить все письма с заданными идентификаторами из **Карантина**.
- `--stat` – вывод статистической информации о найденных в **Карантине** сообщениях с заданными идентификаторами.

Пример:

```
drweb-qcontrol --stat def/%
1. def/backup/B/00014F8B.DW_SHOT_PRODUCT.U0dshM  from: ai@1; to: ai@fff;
time: 2008-08-14 12:10:57
2. def/drweb/F/00014F8F.DW_SHOT_PRODUCT.QqFpdH  from: ai@4; to: ai@fff;
time: 2008-08-14 13:00:50
3. def/backup/C/00014F8C.DW_SHOT_PRODUCT.A39xp7  from: ai@2; to: ai@fff;
time: 2008-08-14 13:00:50
4. def/backup/F/00014F8F.DW_SHOT_PRODUCT.tMi6W2  from: ai@4; to: ai@fff;
time: 2008-08-14 13:00:50
5. def/drweb/3/00014F93.DW_SHOT_PRODUCT.n9xPjU  from: ai@3; to: ai@fff;
time: 2008-08-14 13:30:49
6. def/backup/3/00014F93.DW_SHOT_PRODUCT.ewYFVA  from: ai@3; to: ai@fff;
time: 2008-08-14 13:30:49
7. def/backup/4/00014F94.DW_SHOT_PRODUCT.JQ3sLH  from: ai@3; to: ai@fff;
time: 2008-08-14 13:30:49
```

Действия будут выполняться именно в том порядке, в котором они описаны. Т.е. в одной команде можно указывать сразу несколько действий.

Пример:

```
drweb-qcontrol --send --remove def/backup/F/00014F7F.DW_SHOT_PRODUCT.yv4ro9
```

отправит письмо с идентификатором `def/backup/F/00014F7F.DW_SHOT_PRODUCT.yv4ro9` оригинальным получателям, а затем удалит его из **Карантина**.

Если программный комплекс настроен на хранение **Карантина** в хранилище **DBI**, то для выполнения удаления писем дополнительно потребуется указание в командной строке SQL-команды удаления. Для этого используется дополнительная команда:

- `--sql-remove-command` – команда удаления письма из **Карантина** по его файловому идентификатору. Единственным параметром тут является файловый идентификатор письма.

  Пример:

  ```
  drweb-qcontrol --sql-remove-command "DELETE FROM mail_export WHERE filename
  LIKE ?"
  ```

3. Поиск писем

Утилита **`drweb-qcontrol`** предоставляет также простой интерфейс для поиска в письмах, помещенных в **Карантин**. Доступные критерии поиска:

- `--search-from {адрес}` – поиск по отправителю в конверте письма;
- `--search-to {адрес}` – поиск по получателю в конверте письма;

- `--search-headers {header_name[:value]}` – поиск в заголовках верхнего уровня письма.

  Здесь `header_name` – имя искомого заголовка (допускается только полное соответствие). Если `value` не указано, то для совпадения достаточно одного факта присутствия этого заголовка. Если указано `value`, то оно ищется в значении данного заголовка как подстрока. Поиск имени заголовка и его значения является регистронезависимым;
- `--search-inbody {строка}` – осуществляет поиск заданных подстрок в теле сообщения. Тело сообщения воспринимается как единое целое, и не осуществляется никакого MIME-декодирования. Поиск является регистронезависимым.

Обратите внимание, что если в качестве аргументов для параметров `--search-headers` и `--search-inbody` указываются спецсимволы `*`, `^`, `$`, то их необходимо экранировать символом "\".

Пример:

```
drweb-qcontrol --search-inbody \* --stat
```

Вывод статистики по письмам, найденным по заданным условиям поиска в теле.

Каждый из указанных критериев поиска проверяется независимо, т.е. они объединяются по принципу `OR`.

Пример:

```
drweb-qcontrol --search-to addr1 --search-to addr2
```

будет искать письма, в конверте которых среди получателей есть либо `addr1`, либо `addr2`.

Пример:

```
drweb-qcontrol --search-from from@drweb.com
--search-to to@drweb.com —search-headers
"Subject: [SPAM]" --search-inbody "spam"
```

найдет все письма в **Карантине**, отправителем которых является `from@drweb.com`, или получателем которых является `to@drweb.com`, или в теме которых есть строка `[SPAM]`, или в теле которых есть слово `spam`.

Обратите внимание, что если в параметрах утилиты одновременно указаны какой-либо критерий поиска и список файловых идентификаторов, то поиск будет производиться исключительно в списке файлов из командной строки (происходит пересечение подмножеств по принципу `AND`).

Пример:

```
drweb-qcontrol --stat --search-from ai@5 def/backup/%
```

выводит на консоль статистику обо всех заархивированных сообщениях (имеющих идентификатор, отвечающий указанному шаблону `def/backup/%`), отправителем которых является `ai@5`:

```
1. def/backup/5/00014F95.DW_SHOT_PRODUCT.1LXzg1
from: ai@5; to: to@drweb.com; time:
2008-8-14 15:1:46
```

drweb-lookup: Утилита проверки Lookup

Утилита **drweb-lookup** предназначена для проверки правильности результатов поиска при помощи выражений Lookup, заданных в настройках **Dr.Web MailD**.

Формат запуска утилиты:

```
drweb-lookup [параметры] <запрос>
```

где `<запрос>` – это различные типы `Lookup`, где будет производиться поиск, а `[параметры]` – это параметры командной строки.

Доступны следующие `параметры`:

Краткий вариант	Расширенный вариант	Аргументы
`-h`	`--help`	
Описание: Вывод на консоль краткой справки по параметрам командной строки и завершение работы утилиты		
`-v`	`--version`	
Описание: Вывод на консоль информации о версии и завершение работы утилиты		
`-l`	`--level`	`<уровень подробности>`
Описание: Установка уровня подробности записи в журнал. Возможные значения: `Quiet`, `Error`, `Alert`, `Info`, `Debug`		
`-i`	`--ipc-level`	`<уровень подробности>`
Описание: Установка уровня подробности записи в журнал для подсистемы IPC (взаимодействие с модулем **drweb-maild**). Возможные значения: `Quiet`, `Error`, `Alert`, `Info`, `Debug`		
	`--syslogfacility`	`<метка syslog>`
Описание: Установка типа подсистемы, через которую системный сервис **syslog** (если журналирование работы утилиты ведется через него, см. следующий параметр) выдает сообщения о событиях. Возможные значения: `Daemon`, `Mail`, `Local0`, `...`, `Local7`		
	`--log-filename`	`<имя файла>`
Описание: Установка имени файла журнала или значение `syslog`, если журналирование работы утилиты должно осуществляться с помощью системного сервиса **syslog**		
`-a`	`--agent`	`<путь к файлу>`
Описание: Установка пути к сокету компонента **Dr.Web Agent** для получения от него конфигурационной информации (по умолчанию, если параметр не задан, используется путь `%var_dir%/ipc/.agent`). Если указать ключ без параметра, обращение к **Dr.Web Agent** производиться не будет		
`-t`	`--timeout`	`<период времени>`
Описание: Установка максимального допустимого времени ожидания ответа от **Dr.Web Agent** при получении конфигурации		
`-q`	`--query`	`<искомая строка>`
Описание: Строка, значение которой ищется. Если указано значение "-", то утилита производит чтение искомого значения со стандартного потока ввода		
`-e`	`--exist`	
Описание: Указание, что требуется только проверка наличия запрашиваемого элемента в `Lookup` без получения значения для него (ответом утилиты будет вывод на консоль сообщения `FOUND` или `NOT FOUND` в зависимости от результатов выполнения запроса)		

Примеры:

```
drweb-lookup -q q -e e,w
q NOT FOUND
```

```
drweb-lookup -q q -e q,q
FOUND q
```

```
drweb-lookup -q test@drweb.com -e
'ldap:///?displayName?sub?(mail=$s)'
FOUND test@drweb.com
```

```
drweb-lookup -q test@drweb.com
'ldap:///?displayName?sub?(mail=$s)'
notify.virus=block, notify.virus=allow(rcpt),
drweb/ProcessingErrors = pass
```

```
drweb-lookup -q test@drweb.com
"odbc:select rules from maild where a='\$s'"
scan = all:-drweb
```

drweb-inject: Утилита отправки писем

Утилита **drweb-inject** служит для доставки локальной почты через компонент **Sender**. Она принимает тело сообщения через стандартный поток ввода и завершается с кодом возврата 0 при успехе и <>0 при ошибке. Отправляемое письмо может быть передано утилите либо через конвейер | (например, как результат вызова команды типа **cat**), либо при помощи перенаправления файла на стандартный поток ввода <.

Доступны следующие параметры командной строки:

Краткий вариант	Расширенный вариант	Аргументы
	--help	
Описание: Вывод на консоль краткой справки по параметрам командной строки и завершение работы утилиты		
	--version	
Описание: Вывод на консоль информации о версии и завершение работы утилиты		
	--agent	<путь к файлу>
Описание: Установка пути к сокету компонента **Dr.Web Agent** для получения от него конфигурационной информации (по умолчанию, если параметр не задан, используется путь %var_dir%/ipc/.agent). Если указать ключ без параметра, обращение к **Dr.Web Agent** производиться не будет		
	--timeout	<период времени>
Описание: Установка максимального допустимого времени ожидания ответа от **Dr.Web Agent** при получении конфигурации		
	--id	<идентификатор>

Краткий вариант	Расширенный вариант	Аргументы
Описание: Задание уникального идентификатора компонента **Sender**, который должен использоваться для отправки сообщения. Если параметр не указан, будет использован **Sender** по умолчанию		
-f	--env-from	<адрес>
Описание: Задание отправителя сообщения (для конверта письма). В случае, когда отправитель не указан, используется имя пользователя, с правами которого выполняется утилита. Если имени пользователя найти не удалось, то утилита завершается с ненулевым кодом ошибки.		
-F	--from	<адрес>
Описание: Задание значения поля From отправляемого письма, если в отправляемом письме нет поля From		
-i	--ignore-dot	
Описание: Предписание не воспринимать строку с единственным символом-точкой (".") как признак завершения ввода тела сообщения		
-t	--extract-recipients	
Описание: Предписание добавлять к получателям в конверте всех получателей из поля To письма		

Пример отправки письма с помощью утилиты **drweb-inject**:

```
cat /var/drweb/msgs/out/failed/00000A59tNvGZ8
| drweb-inject -f sender@domain rcpt@domain
```

Будет отправлено письмо, сохраненное в файл **00000A59tNvGZ8**, и находящееся в каталоге ошибочных ("потерянных") писем msgs/out/failed. Этот каталог хранит письма, которые не удалось отправить, в том числе – отложенно. В качестве значения ключа **-f** задается отправитель, а далее – получатель письма.

Обратите внимание, что в случае отправки писем из базы данных почтовых сообщений **Dr.Web MailD** используются именно файлы писем, а не файлы конвертов (у них такой же ID, но расширение .envelope)!

Если список получателей нужно извлечь из тела отправляемого письма, то используется ключ **-t** (в этом случае нужно указать только отправителя):

```
cat absGRjJ0to1Ubye |
drweb-inject -t -f sender@domain
```

Правила обработки писем

Назначение Правил обработки писем

Правила обработки писем предназначены для тонкой настройки параметров, которые будут использоваться при обработке и доставке разных сообщений в зависимости от набора условий, которым они будут соответствовать. Каждое условие проверяет некоторые характеристики сообщения, такие, как адреса отправителя и получателя, названия найденных в письме вирусов и других угроз, размер письма и т.п. Имеется возможность задавать различные комбинации проверок данных параметров, и тем самым менять процедуру обработки сообщений.

Могут быть заданы как общие Правила, проверяемые для всех обрабатываемых писем, так и Правила, связанные с конкретными пользователями, их алиасами и группами пользователей.

Общие Правила обработки писем задаются в основном конфигурационном файле **Dr.Web MailD**,

в секции `[Rules]`, а Правила обработки писем, связанные с пользователями и группами пользователей, задаются во встроенной базе данных.

Помещение Правил во встроенную базу данных наиболее оправдано, если число пользователей с разными настройками обработки писем велико, поскольку обработка Правил в конфигурационном файле становится неэффективной, так как сложность поиска настроек в нем пропорциональна количеству Правил. Поиск сработавших Правил во встроенной базе данных в этом случае работает более эффективно, и, кроме того, оптимизируется использование памяти.

Порядок обработки писем и просмотра Правил

При обработке письма некоторым подключаемым модулем или другим компонентом, он может запросить у **MailD core** значение какого-либо параметра. В этом случае подходящее значение параметра выбирается согласно следующему алгоритму:

- Просматриваются Правила, имеющиеся во встроенной базе данных и связанные с получателем данного письма (получатель определяется по заданному отправителем `RCPT TO`).
- Просматриваются Правила, имеющиеся во встроенной базе данных и связанные со всеми группами, к которым относится пользователь-получатель. Просмотр Правил групп производится в обратном порядке: с настроек самой последней группы и до первой в списке группы.
- Просматриваются Правила, заданные в секции `[Rules]`.

  Обратите внимание на порядок обхода Правил:

  - Все Правила в текущей просматриваемой группе Правил всегда проверяются в порядке их задания.
  - Для каждого проверяемого Правила проверяется условие `CONDITION` – и если оно истинно, то значение требуемого параметра ищется среди элементов секции `SETTINGS` этого правила.
  - Если условие `CONDITION` оказалось ложно, то просмотр Правила заканчивается и происходит переход к поиску значения в следующем Правиле.
  - Если условие `CONDITION` истинно и после него стоит директива `cont`, то происходит переход к проверке следующего Правила. Если же после истинного `CONDITION` стоит директива `stop`, то просмотр Правил заканчивается вне зависимости от того, было найдено значение требуемого параметра или нет.

Значение параметра по результатам просмотра Правил всегда определяется следующим образом:

- Если искомый параметр встретился в одном из сработавшем правил, то используется его значение, извлеченное из части `SETTINGS` (обратите внимание, что при срабатывании нескольких Правил для одного и того же параметра, результирующее значение этого параметра зависит от его семантики. Подробнее об этом см. в подразделах Формат Правил и Особые случаи Правил).
- Если Правила отсутствуют, или ни одно Правило не сработало, или ни в одном из сработавших Правилах параметр не нашелся, то извлекается значение этого параметра, заданное в соответствующей секции конфигурационного файла.
- Если в конфигурационном файле искомый параметр не задан, то используется его значение по умолчанию.

Формат Правил

Формат Правил

Каждое Правило обработки писем записывается в виде строки и имеет вид:

```
CONDITION stop|cont [SETTINGS]
```

где:

- **CONDITION** – условие, которое должно быть истинно для некоторого письма, чтобы при его проверке использовались настройки и/или действия, указанные в части **SETTINGS**.
- **SETTINGS** – настройки и/или действия, которые нужно выполнить при проверке письма, если для него оказалось истинным условие **CONDITION**.

  Обратите внимание, что раздел **SETTINGS** в Правиле может отсутствовать. Это может быть в двух случаях:

  1. Если Правило не несет в себе специфических настроек, а является только "правилом-фильтром";
  2. Если используемые для проверки письма настройки загружаются прямо в процессе выполнения проверке **CONDITION** из внешнего источника (используется Lookup из LDAP, БД, файл и т.п.).
- Директива, стоящая после **CONDITION**, определяет, какие действия следует выполнить, если условие в **CONDITION** оказалось истинным для письма:
  - **stop** – прекратить поиск подходящих Правил ниже;
  - **cont** – продолжить просматривать список Правил вниз.

  Если условие **CONDITION** для некоторого письма оказалось ложным, то указанная после него директива не оказывает никакого влияния: будет продолжен просмотр списка Правил.

Если при записи Правило получается слишком длинным и не помещается в одну строку, в конце строки ставится знак "\", и описание Правила продолжается на следующей строке.

Условия (CONDITION) Правил

Каждое из условий `CONDITION` представляется в виде логической комбинации единичных логических термов:

```
BOOL_TERM [<log_op> BOOL_TERM]
```

где **BOOL_TERM** – логический терм, либо принимающий в значение «ложь» или «истина» в результате проверки некоторого параметра письма (задается выражением `[param_name:][value]`), либо тождественно равный `true` или `false`. Формально логический терм представляется в виде:

```
<[param_name:][value]> | true | false
```

`param_name` – имя параметра, а `value` – его значение.

Между именем параметра и значением (т.е. сразу после двоеточия) в логическом терме не должно быть пробелов.

Названия параметров, которые могут быть использованы в условиях Правил, представлены в таблице:

Имя параметра	Описание	Тип значения
any	Либо адрес отправителя, либо адрес получателя сообщения.	Lookup

Имя параметра	Описание	Тип значения
	Пример: `any:regex:*@domain.com` Или отправитель, или получатель сообщения должны иметь адрес в домене `domain.com`.	
from, **sender**	Адрес отправителя сообщения. **Пример**: `from:admin@domain.com` Отправителем письма должен быть владелец адреса `admin@domain.com`.	Lookup
to, **rcpt**	Адрес получателя сообщения. **Пример**: `"rcpt:ldap:///??sub?(mail=$s)"` Все получатели письма должны быть найдены в LDAP по полю `mail`.	Lookup
block	Имя объекта или причины, вызвавшей блокировку письма некоторым подключаемым модулем. Блокировкой письма считается применение к нему каким-либо модулем действия `reject`. В этом случае модуль, заблокировавший письмо, возвращает список строк, описывающих причины блокировки (их может быть больше одной). Например, модуль **Drweb**, в случае если был обнаружен вирус (или другая известная угроза) вернет его имя. Если блокировка была вызвана по другой причине, (например, по выполнению реакции на событие `SkipObject`), то он возвращает полное строки конфигурации `"<параметр>: <значение>"`, выполнение которой привело к блокировке письма. Например, модуль **HeadersFilter** может возвращать сообщение, что письмо было заблокировано по причине отсутствия заголовка `<header>`, указанного в параметре **MissingHeader**, в следующем виде: `"MissingHeader: <header>"`. **Примеры**: 1) `block:file:viruses.txt` Имя объекта или причины, вызвавшей блокировку письма некоторым подключаемым модулем, должно быть в списке, взятом из файла (подразумевается, что это имя некоторой найденной угрозы). 2) `"block:regex:.*skip.*"` Описание причины, вызвавшей блокировку письма некоторым подключаемым модулем, должно содержать подстроку `"skip"` (например, выполнение реакции на событие `SkipObject`).	Lookup
client-ip	IP-адрес отправителя письма (если получение информации об IP-адресе отправителя было задано в настройках компонента **Maild core**). Обратите внимание, что в качестве аргумента нельзя использовать формат CIDR. **Пример**: `client-ip:127.0.0.1` Письмо должно быть отправлено с локальной машины.	Lookup
client-port	Порт клиента, отправившего письмо.	Номер порта

Имя параметра	Описание	Тип значения
	Пример: `client-port:1234` Письмо должно быть отправлено с порта 1234.	
server-unix-socket	Абсолютный путь к файлу UNIX-сокета, на котором было принято соединение. **Пример**: `server-unix-socket:/var/drweb/ipc/sock1`	Путь к UNIX-сокету
server-ip	IP-адрес интерфейса, на котором **Receiver** принял письмо. Обратите внимание, что в качестве аргумента нельзя использовать формат CIDR.	Lookup
server-port	Порт сервера, на котором было принято соединение	Номер порта
id	Уникальный идентификатор компонета **Receiver**, принявшего письмо (если был задан в настройках компонента **Receiver**).	Строка-идентификатор
auth	Прошел ли авторизацию клиент, отправивший письмо. Обратите внимание, что параметр не имеет аргумента. **Пример**: `auth:`	Отсутствует
size	Размер сообщения. Перед размером можно указывать оператор сравнения: • `!=` – Не равно. • `==` – Равно (совпадает). • `<` – Меньше, чем. • `>` – Больше, чем. • `<=` – Меньше или равно. • `>=` – Больше или равно. Если отношение не указано, то по умолчанию предполагается, что это `<=`. Размер указывается в килобайтах, мегабайтах или гигабайтах, для чего после числа указывается соответствующий суффикс (`k`, `m`, `g`). **Пример**: `"size:>=10m"`	Размер или оператор сравнения
score	Счет сообщения. Перед размером можно указывать оператор сравнения. Если оператор не указан, то по умолчанию предполагается, что это `<=`. **Пример**: `"score:!=1000"`	Число или оператор сравнения

Обратите внимание, что:

- Если имя параметра в логическом терме не указано, то по умолчанию используется параметр **any** (например, запись `user@domain.com stop <some_settings>` эквивалентна Правилу `any:user@domain.com stop <some_settings>`).
- Если в значении параметра содержатся пробелы или символы '`|`', '`&`', '`)`', '`(`' '`!`' '`=`' '`,`' то его надо заключать в кавычки. Чтобы задать символ '`"`' (кавычка) внутри кавычек, его надо предварять (экранировать) знаком '`\`'.
- Если в Правиле нужно использовать пустое значение адреса, то следует использовать запись вида "", а использовать угловые скобки (<>) в данном случае нельзя (например `from:"" stop`

scan=no).

В качестве логических операторов используются AND ('&&'), OR ('||'), NOT ('!'). При помощи них можно составлять сложные условия из простых логических термов. Также можно использовать скобки для управления старшинством логических операций. Минимально условие правила состоит из одного логического терма.

Пример 1:

```
true stop <some_settings>
```

Настройки <some_settings> будут применяться всегда, если до этого Правила дошла очередь при проверке (условие тождественно истинно для любого письма). Более никакие нижележащие Правила применены не будут, т.к. будет выполнена директива **stop**.

Пример 2:

```
sender:test && "size:>=10k" cont scan=no
```

Данное условие будет истинным, если отправителем письма является "test" и размер письма больше 10 килобайт. В этом случае сообщение будет пропущено без проверки. Обратите внимание на использование кавычек: они тут необходимы.

Пример 3:

```
!("rcpt:ldap:///??sub?(mail=$s)" OR auth:) stop
```

Данное Правило сработает, если хотя бы один получатель письма, взятый из TO: не найден в LDAP по полю mail и отправитель не является авторизованным.

Настройки (SETTINGS) Правил

Часть SETTINGS правил имеет вид перечисления пар Параметр = Значение:

```
[plugin_name/]param = value,
[plugin_name/]param = value ...
```

где **plugin_name** (если указано) – название подключаемого модуля, к которому относится параметр, **param** – название параметра, а value – его значение.

Если имя модуля не указано, то это означает, что переопределяется значение параметра из основного конфигурационного файла **Dr.Web MailD** (имя секции не указывается). Например, указание в Правиле директивы **AdminMail**=root@domain означает переопределение для данного письма параметра **AdminMail** в секции [Notifier] конфигурационного файла **Dr.Web MailD.**

В перечислении должно присутствовать не менее одной пары Параметр = Значение. Элементы списка разделяются запятыми, поэтому, если в значении value содержится запятая, то перед ней следует поставить обратный слэш "\" (экранировать).

Пример 1:

```
sender:a@drweb.com cont headersfilter/Action = pass, vaderetro/max_size = 100k
```

При срабатывании этого Правила (для отправителя a@drweb.com) выбирается значение параметра **Action** = pass для подключаемого модуля **Dr.Web HeadersFilter** и максимальный размер проверяемого сообщения (**max_size**), равный 100k, для подключаемого модуля **Vaderetro**, после чего будет продолжен поиск подходящих Правил ниже (т.к. указана директива **cont**).

Пример 2:

```
to:a@drweb.com cont drweb/ProcessingErrors = pass\, redirect(err@drweb.com)
```

Обратите внимание, что задается один параметр **ProcessingErrors** для подключаемого модуля **Drweb**, имеющий в качестве значения два значения, разделенных запятой (`pass, redirect(err@drweb.com)`). Поэтому данном случае нельзя заключать его в кавычки, так как в этом случае парсер конфигурации будет воспринимать его как одно значение и не разбивать на подстроки при разборе параметра **ProcessingErrors**. Запятая в данном случае экранируется.

Вся обработка Правил происходит по порядку, сверху вниз и слева направо, поэтому:

- Если в одном и том же Правиле (в его части `SETTINGS`) значение одного и того же параметра задается несколько раз, то последующее значение параметра замещает значение предыдущее.
- Если сработало несколько Правил, в которых (в части `SETTINGS`) присваиваются разные значения одному и тому же параметру, то неизменным сохраняется первое присвоенное значение, а попытки последующих изменений будут проигнорированы.

Пример 3 (одно Правило с несколькими значениями одного и того же параметра):

```
true cont html=yes, html=no
```

Параметру **html** (о параметре **html** см. ниже) будет присвоено значение `no`.

Пример 4 (несколько сработавших Правил с разными значениями одного и того же параметра):

```
true cont html=yes
true cont html=no
```

Параметру **html** будет присвоено значение `yes`.

Описанное выше поведение относится ко всем параметрам, кроме параметров, которые обладают **накапливающей ("аддитивной") семантикой**. Каждый раз, когда встречается параметр с накапливающей семантикой, новое его значение, взятое из очередного сработавшего Правила, не конфликтует с предыдущим значением, присвоенным параметру, а добавляется к нему. В результате все найденные значения параметра объединяются в единый список-значение. При этом директива **stop**, встреченная в очередном сработавшем Правиле, для таких параметров работает как обычно – просмотр дальнейших Правил прекращается, параметр получает в качестве значения накопленный к этому моменту список значений. В данном документе параметры, обладающие аддитивной семантикой, отмечаются в их описании пиктограммой A.

Кроме того, имеется также особый тип параметров, называющихся **клонирующими**. Эти параметры обрабатываются для каждого получателя отдельно: т.е. если у двух получателей письма, в результате срабатывания разных Правил, для какого-либо параметра будут указаны разные значения, то для каждого из получателей будет создана отдельная копия письма, к которой будут применены свои настройки обработки. В данном документе параметры, поддерживающие клонирование, отмечаются в их описании пиктограммой C.

Параметры, используемые в настройках (SETTINGS) Правил

В части **SETTINGS** Правил могут использоваться параметры следующих видов:

- те, которые встречаются только в Правилах.
- те, значения которых задаются в конфигурационном файле **Dr.Web MailD** или в конфигурационных файлах других модулей или подключаемых модулей.

В данном документе параметры, которые могут использоваться в Правилах, отмечаются в их описании пиктограммой R.

1. Параметры, используемые только в Правилах:

html = {логический}	Использовать ли HTML-форматирование уведомлений MailD. В случае если указано Yes, уведомления MailD генерируются в формате HTML, в противном случае – в текстовом виде. Обратите внимание, что этот параметр не управляет видом DSN.
quarantine = {логический}	Разрешение перенаправления отвергнутых писем в **Карантин**. В случае если указано Yes, разрешается перемещать письма в **Карантин**, в случае если они не пройдет проверку подключаемыми модулями. В противном случае письмо не попадет в **Карантин**, даже в случае, если оно будет отвергнуто подключаемыми модулями.
scan = {текст}	Параметр определяет, какие подключаемые модули **Dr.Web MailD**, из заданных в секции [Filters], будут использованы для проверки письма. Если указано значение All, будут задействованы все модули. Если указано No, то никакие модули не будут использоваться. Также имеется возможность указать выборочный список используемых подключаемых модулей. Разделителем между именами модулей служит двоеточие ":". Чтобы исключить из использования какие-либо модули, следует перед их именем ставить знак минус "-" без пробела между минусом и именем модуля. Обратите внимание, что при указании значения All не разрешается использовать имена подключаемых модулей без знака минус, так как это является бессмысленным. **Примеры**: **scan** = All – использовать все модули; **scan** = no – не использовать ни одного модуля; **scan** = All:-plugin1 – использовать все подключаемые модули, кроме plugin1; **scan** = Plugin1:Plugin2 – использовать только подключаемые модули Plugin1 и Plugin2; **scan** = All:Plugin1 – неверно, т.к. нельзя использовать имена модулей без "-" после All; **scan** = -Plugin1:All – неверно, т.к. All может быть только на первой позиции; **scan** = -Plugin1 – неверно, т.к. в начале отсутствует All.

notify[.<тип уведомления>] = {allow | block} [(<типы адресов>)]

Данный параметр управляет отправкой уведомлений MailD разных типов.

Значение `allow` разрешает вывод соответствующего уведомления, а значение `block` – запрещает. Если тип уведомления не указан, то значение параметра применяется ко всем уведомлениям.

Возможные типы уведомлений зависят от того, какие виды уведомлений поддерживает компонент **Notifier**. Дополнительно установленные подключаемые модули могут добавлять свои собственные типы уведомлений.

По умолчанию поддерживаются следующие типы уведомлений:

- **notify.Virus** – уведомления об обнаруженных вирусах в почтовом сообщении;
- **notify.Cured** – уведомления об излеченных вирусах в сообщении;
- **notify.Skip** – уведомления о письмах, содержащих объекты, пропущенные при сканировании;
- **notify.Archive** – уведомления о сообщениях, не проверенных в связи с ограничениями на проверку архивов;
- **notify.Error** – уведомления об ошибках, возникших при проверке писем;
- **notify.Rule** – уведомления о блокировании письма каким-либо правилом;
- **notify.License** – уведомления о письмах, не проверенных в связи с лицензионными ограничениями;
- **notify.Malware** – уведомления об обнаруженных вредоносных программах.

Следом за значением параметра в скобках может идти необязательный модификатор, указывающий, адресаты каких типов подпадают под действие этого значения параметра. Можно указать несколько видов адресов, разделенных двоеточиями. Возможные значения модификатора (типы адресата):

- `sender` – отправитель письма;
- `rcpt` – получатель письма;
- `admin` – администратор;
- `any`, либо модификатор отсутствует – адресат любого типа.

Примеры:

notify=block или **notify**=block (any) – блокирование всех уведомлений для всех адресатов.

notify.Virus=block (sender:admin) – блокирование уведомлений о найденных вирусах для администратора и отправителя письма.

Если для уведомления некоторого типа `<type>` не найдено правило **notify.<type>**, а также не найдено общее правило **notify**, то предполагается, что соответствующее уведомление отключено.

Обратите внимание, что эта настройка не управляет видом DSN.

NotificationNamesMap = name1 file_name1, name2 file_name2, ...	Этот параметр позволяет отображать названия используемых шаблонов уведомлений в новые. Например, может использоваться для формирования уведомлений разного вида в зависимости от конверта. Здесь, соответственно: • nameN – имя запрашиваемого уведомления, для которого устанавливается новый файл шаблона. Список имен можно найти в описании параметра **notify**. Кроме того, здесь можно для шаблонов общих отчетов указывать слово report, а для DSN - dsn. • file_nameN – часть имени нового файла с шаблоном уведомления. Полное имя шаблона составляется по следующей схеме: в начало имени файла добавляется один из необходимых префиксов: sender_, rcpts_, admin_, report_ или dsn_, расширение файла меняется на .msg. В результате получается имя файла, например sender_file_nameN.msg, который будет искаться в каталоге, указанном в значении параметра **TemplatesBaseDir** секции настроек [Notifier]. **Пример**: **NotificationNamesMap** = virus my-virus, archive my-arch
SenderAddress = {address1|address2|...}	Адрес, передаваемый компоненту **Sender**, чтобы тот отправил на него письмо. Можно указывать несколько адресов, разделяя их знаком "|" (по аналогии с параметром **Router** из секции [Sender]). При использовании параметра **SenderAddress** в Правилах вида: <CONDITION> cont SenderAddress = address1|address2|address3 сообщение, удовлетворяющее условию <CONDITION>, будет послано на первый доступный адрес из этого списка. То есть, если address1 недоступен, то письмо будет отправлено на address2, а если и этот адрес не доступен, то на address3. Данный параметр может использовать специальные макросы: • CLIENT-IP – IP-адрес клиента, от которого было получено письмо • CLIENT-PORT – Порт клиента, с которого было получено письмо • SERVER-IP – IP-адрес, который обслуживается **Dr.Web MailD** • SERVER-PORT – Порт, через который **Receiver** принял письмо. Если **Sender** поддерживает данный параметр, то он передаст письмо именно на указанный адрес. Например, Правило вида true cont SenderAddress=inet:10025@CLIENT-IP будет перенаправлять все исходящие письма на хост, с которого они получены, на порт 10025. В текущей версии **Dr.Web MailD** параметр **SenderAddress** поддерживается только модулем **drweb-sender** с SMTP/LMTP методом отправки почты.

Значения «по умолчанию» для параметров, перечисленных в таблице выше, задаются в «безымянной» секции [Rule] конфигурационного файла.

Для корректной работы макросов CLIENT-IP и CLIENT-PORT, используемых в параметре **SenderAddress**, необходимо установить значения следующих параметров:

- В секции [Receiver]: **RealClients** = yes
- В секции [Maild]: **GetIpFromReceivedHeader** = yes

2. Параметры из секций основного конфигурационного файла, которые могут быть использованы в Правилах:

Секция	Параметры
[Maild]	**RedirectMail** **MaxScore** **MaxScoreAction** **LicenseLimit** **EmptyFrom** **ProcessingErrors** **UseCustomReply** **ReplyEmptyFrom** **ReplyProcessingError** **ReplyMaxScore**
[Notifier]	C **AdminMail** **FilterMail** C **NotifyLangs**
[Filters]	C **<plug-in>/use** - Параметр разрешает или запрещает использование указанного подключаемого модуля при проверке писем. Имеет логический тип {yes, no}. Обратите внимание, что этот параметр может быть использован только в Правилах и не задается в конфигурационном файле! C **<plug-in>/max_size** **<plug-in>/log_level** **<plug-in>/log_ipc_level** **<plug-in>/syslog_facility** **<plug-in>/path_to_lib**

3. Параметры подключаемых модулей, которые могут быть использованы в Правилах:

Модуль	Параметры
Drweb	**HeuristicAnalysis** **AddXHeaders** **Paranoid** A **RegexsForCheckedFilename** **LicenseLimit** **Infected** **Suspicious** **Incurable** **Adware** **Dialers** **Jokes** **Riskware** **Hacktools** **SkipObject** **ArchiveRestriction** **ScanningErrors**

Модуль	Параметры
	ProcessingErrors BlockByFilename UseCustomReply ReportMaxSize ReplyInfected ReplyMalware ReplySuspicious ReplySkipObject ReplyArchiveRestriction ReplyError ReplyBlockByFilename
Vaderetro	FullCheck NoHamFrom AddVersionHeader AddXDrwebSpamStateNumHeader AddXSpamLevel AddXHeaders CheckDelivery SubjectPrefix NotifySubjectPrefix UnconditionalSpamThreshold UnconditionalSubjectPrefix SpamThreshold UnconditionalAction Action NotifyAction SpamCustomReply A WhiteList A BlackList CheckForViruses AllowRussian AllowCJK UseCustomReply FromProtectedNetworkScoreAdd ProtectedNetworkReplyCacheLifeTime ReplyToProtectedNetworkScoreAdd
HeadersFilter	ScanEncodedHeaders A RejectCondition A AcceptCondition FilterParts A RejectPartCondition A AcceptPartCondition A MissingHeader Action UseCustomReply ReplyRuleFilter
Modifier	Encoding UseCustomReply ReplyRuleFilter A LocalRules – Локальные правила проверки писем модулем (аналогичен по

Модуль	Параметры
	формату параметру **GlobalRules** подключаемого модуля). Обратите внимание, что этот параметр модуля может быть использован только в Правилах и не задается в самом конфигурационном файле подключаемого модуля! О формате правил модуля см. в описании подключаемого модуля.

Обратите внимание, что логика работы параметра **<plug-in>/use** аналогична логике работы параметра **scan** (включение или исключение модуля в список используемых для проверки). Однако он может использоваться для дополнительного включения или выключения подключаемых модулей при срабатывании разных Правил, т.к. параметр **scan** не обладает накапливающей семантикой.

Пример:

Если в результате срабатывания двух Правил требуется отключить подключаемые модули **Drweb** и **Vaderetro**, то использование Правил вида

```
to:regex:test@.* cont scan=all:-vaderetro
to:regex:test@.* cont scan=all:-drweb
```

приведет к тому, что отключится только модуль **Vaderetro** (поскольку повторное изменение параметра **scan** будет отброшено). Однако, если переписать второе Правило следующим образом:

```
to:regex:test@.* cont scan=all:-vaderetro
to:regex:test@.* cont drweb/use=no
```

то после срабатывания первого Правила отключится **Vaderetro**, а после второго также и **Drweb**. Тот же эффект будет достигнут, если в **SETTINGS**-части первого правила заменить **scan**=all:-vaderetro на **vaderetro/use**=no.

Для использования в **SETTINGS**-части Правила значений параметров из некоторой именованной группы настроек (задается как секция с заданным названием [Rule:<название_секции>] в основном конфигурационном файле), используется директива **rule**=<имя группы>. В текущей версии **Dr.Web MailD** в каждом Правиле обработки писем может быть использовано не более одного параметра **rule** (примеры см. в описании секции [Rule]).

Особые случаи Правил

Обработка писем с несколькими получателями

В случае если письмо имеет несколько получателей, для которых одни и те же параметры в результате срабатывания разных Правил получают различные значения (например, для одного сработало Правило, содержащее настройку **html**=yes, а для другого – **html**=no), то возникающие противоречия разрешаются следующим образом:

1. Если параметры, получившие разное значение для различных получателей, указанных в письме, допускают клонирование, то письмо клонируется (создаются независимые копии письма, каждое из которых будет направлено конкретному получателю), к каждой копии применяются свое значение этого параметра.
2. Если параметр не допускает клонирования письма, то разные значения параметра для разных получателей заменяются на единственное значение этого параметра, извлекаемое из соответствующей секции настроек конфигурационного файла, или значение параметра по умолчанию, если он не задан в конфигурационном файле.
3. Если для всех получателей письма сработало одно и то же Правило, или во всех сработавших Правилах этот параметр имеет одно и то же значение, то для письма используется единое значение параметра, взятое из сработавших Правил.

Обратите внимание, что при обработке письма для всех параметров, не измененых в

сработавших Правилах, будут использованы значения, указанные в конфигурационных файлах. Для параметров, значения которых не указаны в конфигурационном файле, будут использованы значения по умолчанию. Письмо будет отправлено на проверку всем подключаемым модулям, указанным в секции `[Filters]`, за исключением тех, которые указаны в сработавших Правилах, как пропускаемые (**<plug-in>/use** = `no` или **scan**=`all:-<plug-in>`).

Пример 1:

a) Имеется 2 Правила:

```
[Rules]
to:user1@domain.ru cont drweb/Suspicious = pass\, quarantine\, notify
to:user2@domaun.ru cont drweb/Suspicious = discard\, quarantine\, notify
```

b) Обрабатывается письмо со следующими заголовками:

```
FROM: <another_user@externaldomain.com>
TO: <user1@domain.ru>, <user2@domain.ru>
```

В этом случае клонирования письма не происходит – при его обработке используется настройка для **Suspicious** подключаемого модуля **Drweb** по умолчанию из конфигурационного файла `plugin_drweb.conf`. Это связано с тем, что настройки для получателей различны, а различие значений параметра `drweb/`**Suspicious** клонирования не вызывает (он не относится к перечню «клонирующих» параметров).

Пример 2:

a) Имеется 3 Правила:

```
[Rules]
to:user1@domain.ru cont drweb/Suspicious = pass\, quarantine\, notify
to:user2@domain.ru cont drweb/Suspicious = discard\, quarantine\, notify
from:another_user@externaldomain.com cont drweb/Suspicious = reject\, add-
header (BLA:BLA)
```

b) Обрабатывается то же самое письмо, что и в предыдущем примере:

```
FROM: <another_user@externaldomain.com>
TO: <user1@domain.ru>, <user2@domain.ru>
```

К письму будет применено только последнее Правило, т.к. параметр `drweb/`**Suspicious** не относится к параметрам с накапливающей семантикой и не является клонирующим, и потому по первым двум правилам из-за различных его значений для обоих получателей он сбросится в значение по умолчанию, но третье правило также сработает для этого письма, и присвоит параметру `drweb/`**Suspicious** значение `reject,add-header(BLA:BLA)`.

Пример 3:

a) Имеется набор Правил:

```
[Rules]
to:user1@domain.ru cont NotifyLangs=ja, AdminMail=admin2@domain.ru
to:user2@domain.ru cont NotifyLangs=ja, AdminMail=admin2@domain.ru
to:user3@domain.ru cont NotifyLangs=ja, AdminMail=admin2@domain.ru
from:root@domain.ru cont NotifyLangs=ru
to:admin@domain.ru cont NotifyLangs=ru\, ja
```

b) Настройки отправки уведомлений MailD по умолчанию следующие:

```
...
[Notifier]
...
AdminMail = admin@domain.ru
FilterMail = drweb@domain.ru
NotifyLangs = en
...
```

c) Обрабатывается зараженное письмо:

```
FROM: <root@domain.ru>
TO: <user1@domain.ru>, <user2@domain.ru>, <user3@domain.ru>
```

Поскольку все уведомления отправляются от лица адреса, указанного в **FilterMail**, то будет отправлено пять уведомлений:

- От <drweb@domain.ru> администратору <admin@domain.ru>;
- От <drweb@domain.ru> отправителю <root@domain.ru>;
- От <drweb@domain.ru> получателю <user1@domain.ru>;
- От <drweb@domain.ru> получателю <user2@domain.ru>;
- От <drweb@domain.ru> получателю <user3@domain.ru>.

При этом администратор <admin@domain.ru> получит уведомления на русском и японском, отправитель <root@domain.ru> получит уведомление на английском, а все получатели (<user1@domain.ru>, <user2@domain.ru>, <user3@domain.ru>) получат уведомления на японском.

Обратите внимание, что Правило `from:root@domain.ru cont NotifyLangs=ru` в данном случае не применимо, так как при посылке уведомления отправителю адрес <root@domain.ru> выступает в качестве получателя уведомления (поэтому для этого уведомления будет использован языковой файл, указанный по умолчанию – `en`). Поэтому если нужно, чтобы ему приходили уведомления на русском, верным будет написать Правило

```
to:root@domain.ru cont NotifyLangs=ru
```

Правила с неявной частью SETTINGS

Как уже упоминалось выше, секцию `SETTINGS` в Правилах можно не указывать. В этом случае предполагается, что параметры можно запросить непосредственно с сервера при помощи Lookup, заданной в части `CONDITION`. Это полезно, например, при работе с LDAP или с базами данных.

Пример:

```
to:regex:.*@drweb.com && "ldap:///?drwebRules?sub?(mail=$s)" cont
```

В этом Правиле, если получатель находится в домене `drweb.com`, и отправитель или все получатели удовлетворяют LDAP-условию `mail=$s`, то параметры в `SETTINGS`-часть Правила будут подставлены из поля `drwebRules` LDAP-запроса. Загрузка происходит для каждого нового письма, и затем кешируется на время его проверки – таким образом пользователь может менять свои настройки в "горячем" режиме, не перезапуская сервер. Пожалуйста, обратите внимание, что `Lookup` к LDAP заключен в кавычки: это связано с присутствием в нем скобок. Возвращаемые полем `drwebRules` поля должны представлять собой корректные строки `SETTINGS` (т.е. `<parameter>=<value>[, <parameter>=<value>, ...]`).

Например, если они хранятся в некоторой таблице базы данных, то они должны иметь следующий вид:

Пример:

Address	Rules
test1@drweb.com	VadeRetro/SubjectPrefix = \"spam\",modifier/localrules=select message\,append_text "Some

Address	Rules
	Text"
test2@drweb.com	headersfilter/MissingHeader = Date,headersfilter/MissingHeader = From, headersfilter/MissingHeader = To

Также обратите внимание, что если часть SETTINGS в Правиле присутствует, а в условии CONDITIONS стоит выражение Lookup, извлекающее некоторые настройки из источника данных, то извлеченные при срабатывании Правила настройки будут присоединены (конкатенированы) к уже имеющейся части SETTINGS **слева** (сначала пойдут настройки, извлеченные из источника, а потом те, которые записаны непосредственно в SETTINGS-части Правила). Это важно иметь в виду в случае возможного конфликта разных значений одного и того же параметра, вставленных в Правило из разных мест.

Пример:

a) Допустим, таблица **table** в базе данных, доступ к которой настроен через ODBC, имеет следующий вид:

Address	Rules
test1@drweb.com	modifier/LocalRules = select message\, append_text "text from DB"

b) В конфигурационном файле задано следующее Правило:

```
"to:odbc:select Rules from table where Address='$s'" cont modifier/LocalRules =
select message\, append_text "text from rule", modifier/LocalRules = quarantine
```

В этом случае, при срабатывании Правила, с учетом всех настроек, если письмо идет получателю test1@drweb.com, то для него будут использоваться следующие настройки (SETTINGS):

```
modifier/LocalRules = select message, append_text "text from DB", select message,
append_text "text from rule", quarantine
```

Обратите внимание, что поскольку параметр **modifier/LocalRules** имеет "накапливающий" характер, то новые значения этого параметра в Правиле не затирают предыдущие, а добавляются в список через запятую.

Если в письме кроме test1@drweb.com есть еще получатели, и для них в базе данных указаны другие значения **modifier/LocalRules** (или они вообще не указаны), то все значения **modifier/LocalRules** из базы данных будут проигнорированы для всех получателей. Будет использоваться только та часть настроек, которая задана в Правиле (значение совпадает для всех получателей):

```
select message, append_text "text from rule", quarantine
```

При необходимости в используемых Lookup можно локально переопределять настройки **OnError** и **SkipDomains** используемого источника данных.

Пример:

```
to:ldap:onerror=exception|skipdomains=file:/etc/drweb/skipdomains.list|
///?description?sub?(cn=$u) cont
```

В соответствии с указанным условием, требуется, чтобы имена всех получателей письма (части username из записи адреса username@domain, согласно используемому макросу $u) были найдены в LDAP. Однако при этом в Lookup-выражении также определена настройка **SkipDomains**, требующая пропускать без запроса домены (часть domain адреса), которые будут найдены в файле /etc/drweb/skipdomains.list. Поэтому правило будет обработано следующим образом:

1. Из письма извлекается перечень получателей (поле To:).

2. Берется адрес очередного получателя письма.
3. Определяется, принадлежит ли домен этого адреса списку доменов, пропускаемых без запроса. Если да, то переход к следующему адресу из списка (пункт 2). Если нет, то выполняется LDAP-запрос. Обратите внимание, что если в письме всего один получатель, и его домен принадлежит к пропускаемым, то запрос к LDAP выполнен не будет, и фактически данное правило не сформирует никаких настроек для данного письма.
4. Если результат запроса положительный, содержимое поля `description` используется как настройка (конкатенируется слева к ранее извлеченным настройкам), затем – переход к следующему адресу из списка (пункт 2). В противном случае, если результат запроса отрицательный (имя пользователя не найдено в источнике данных), то условие Правила считается ложным для письма, Правило отвергается и все извлеченные из LDAP настройки отбрасываются. Оставшиеся запросы к LDAP не выполняются.
5. Если выполнить LDAP-запрос не удается, то эта ситуация является ошибкой обработки письма, обработка Правил для этого письма будет остановлена, и, в соответствии с настройкой **`OnError`**=`exception`, будет выполнено действие, указанное в параметре **`ProcessingErrors`** (секция `[Maild]`). Подробнее об обработке ошибок в Правилах см. ниже.

Поскольку в Правиле указана директива продолжения поиска `cont`, после просмотра этого Правила, если не было ошибок, будет продолжен просмотр других правил. Обратите внимание, что если в рассмотренном примере в письме указано несколько получателей, и из LDAP для них будут извлечены настройки с разными значениями параметров, то реально примененные настройки будут определены в результате клонирования письма или замены на значения параметров из конфигурационного файла (как описано выше, в пункте Обработка писем с несколькими получателями).

Соединение настроек Правил из разных источников

Как указано в начале раздела, при обработке письма значения требуемых параметров извлекаются из Правил, расположенных во внутренней базы данных, затем – из Правил в секции `[Rules]`, а потом из значений параметров, указанных в конфигурационном файле. Рассмотрим пример соединения для письма настроек, извлекаемых из разных мест в соответствии с описанным в начале главы алгоритмом.

Пример:

а) Пусть во внутренней базе данных для пользователя `test@drweb.com` имеются следующие Правила:

```
> email-info test@drweb.com
test@drweb.com A=1 S=1
  name:
  aliases: alias_test@drweb.com
  groups: divine good evil
  rules:
    1: true cont modifier/LocalRules = select message\, append_text
"Scanned! (1)",\
        modifier/LocalRules = quarantine
    2: true cont modifier/LocalRules = select message\, append_text
"Scanned! (2)"
    3: "rcpt:odbc:select rules from maild where addr='$s'" cont
  custom:
```

Т.е. с почтовым адресом `test@drweb.com` связано три Правила обработки, причем третье – с неявной частью `SETTINGS`, проверяющее условие через `Lookup` и импортирующее результат выборки (значение поля **`rules`**) как `SETTINGS`-часть Правила. Правило использует данные из таблицы **`maild`** базы данных, доступ к этой таблице также настроен через ODBC.

b) Пусть запись в таблице, связанная с адресом `test@drweb.com`, имеет следующий вид:

addr	rules
test@drweb.com	modifier/LocalRules = select message\, append_text "Scanned! (3)"

c) И пусть, кроме того, в секции `[Rules]` основного конфигурационного файла задано следующее Правило:

```
[Rules]
true cont modifier/LocalRules = select message\, append_text \
"Scanned! (4)", modifier/LocalRules = quarantine
```

В этом случае для письма, в котором в качестве получателя указан адрес `test@drweb.com`, выполняется следующий просмотр Правил:

1) Сначала выполняются Правила из внутренней базы данных. Поскольку все они тождественно истинны (в качестве условия задано `true`), то их значения используются. Кроме того, параметр **`modifier/LocalRules`** обладает накапливающей семантикой, а значит, значения из всех Правил для него будут последовательно конкатенированы.
2) Далее, в соответствии с третьим Правилом, будет произведен запрос к базе данных и результат запроса (значение поля **`rules`** для адреса `test@drweb.com`) также будет присоединен к текущему накопленному значению.
3) После этого (поскольку не встретилось **`stop`** и параметр **`modifier/LocalRules`** обладает накапливающей семантикой) просматриваются истинные Правила из секции `[Rules]` конфигурационного файла. Найденные значения параметра **`modifier/LocalRules`** также будут конкатенированы.

Итого, после просмотра всех Правил, для письма, в котором в качестве получателя указан адрес `test@drweb.com`, параметр **`modifier/LocalRules`** будет иметь следующее совокупное значение:

```
modifier/LocalRules = select message, append_text "Scanned! (1)", quarantine, select
message, append_text "Scanned! (2)", select message, append_text "Scanned! (3)",
select message, append_text "Scanned! (4)", quarantine
```

Обработка ошибок и проверка корректности Правил

Обработка ошибок

Если в строке с Правилом при обработке будет найдена ошибка, то информацию о ней выводится в журнал, а само Правило игнорируется. Кроме того, случае `Lookup` производится так же обработка возникшей ошибки в соответствии со значением параметра **`OnError`** (заданного для источника данных или переопределенного непосредственно в `Lookup`).

Обратите внимание, что значения `Lookup` и значения конкретных переменных не обрабатываются сразу – их разбор происходит только тогда, когда возникает реальная необходимость в их использовании. Поэтому при обычной загрузке конфигурационного файла ошибки в этих элементах не видны, и их можно будет заметить только при обработке писем (когда Правило с ошибкой будет проигнорировано).

Кроме того, если при анализе Правила `Lookup`-выражение вернет некорректные результаты, то все результаты, возвращаемые этим `Lookup`-выражением, будут отброшены и проигнорированы (даже если часть из них корректна).

Проверка Правил

Для проверки корректности Правил имеется возможность запустить модуль **drweb-maild** с указанием специальных параметров командной строки. С помощью этих параметров задаются различные свойства предполагаемого письма, и модуль выводит на экран консоли все настройки из Правил, которые будут применимы к данному письму. Перечень доступных параметров проверки перечислен в разделе Параметры командной строки (подраздел **Специфические параметры модуля drweb-maild**).

Пример команды проверки Правил:

```
$ ./drweb-maild --auth
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  notify* :
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  all : block
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  archive :
from=allow; admin=allow;
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  cured : from=allow;
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  error : from=allow;
admin=allow;
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  license :
admin=allow;
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  malware :
from=allow; to=allow; admin=allow;
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  rule : admin=allow;
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  skip : from=allow;
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  virus : from=allow;
to=allow; admin=allow;
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  scan : all
Thu May 29 16:03:44 2009 [3081324208] maild.rules DEBUG  html : 1
```

В данном случае эмулируется поступление на обработку письма с установленной отметкой об успешной авторизации отправителя (`auth`). Другие параметры письма не указаны. В выводе на экран указаны все параметры, которые установятся для такого письма после срабатывания всех Правил (в данном случае – настройка генерации уведомлений для разных результатов проверки, указание на что, что письмо будет проверено всеми используемыми подключаемыми модулями, а также, что отчеты будут генерироваться в формате HTML).

Технология Unified Score

Технология **Unified Score** позволяет через единый счет, присвоенный каждому письму, определять нежелательную корреспонденцию. Счет представляет собой целое знаковое число. Чем больше это число, тем больше вероятность того, что письмо является нежелательным, и наоборот, чем меньше число – тем меньше вероятность, что письмо нежелательное. По умолчанию считается, что письма со счетом меньше значения `SpamThreshold` (т.е. `99` и меньше) являются чистыми. Если счет письма больше значения `SpamThreshold`, но меньше `UnconditionalSpamThreshold` (т.е. от `100` до `999` по умолчанию), такое письмо считается спамом. В случае, когда счет письма больше или равен значению параметра `UnconditionalSpamThreshold` (`1000` по умолчанию), оно считается безусловным спамом.

Изменения счета письма происходит различными способами:

- в параметрах с типом `Action` можно использовать необязательное действие `score(SCORE)`, где `SCORE` – целое число, которое будет добавлено к текущему счету письма (или вычтено из него, если эта величина отрицательная);

- счет, выставляемый антиспам-модулем **Vaderetro**, прибавляется к общему счету письма, и уже это итоговое значение потом сравнивается с порогами спама.
- можно изменять значение счета (это возможно в правилах некоторых подключаемых модулей, а также с помощью параметров в некоторых ограничениях, задаваемых в секции `[Receiver]`).
- с помощью Reputation IP Filter можно изменять значение счета всех сообщений в данной сессии.

Использовать значение счета письма можно следующим образом:

- в подключаемом модуле **Vaderetro** он сравнивается с пороговыми значениями для спама;
- в Правилах обработки писем можно использовать значение счета письма в условиях (префикс `score`);
- в подключаемом модуле **Dr.Web Modifier** счет также можно использовать в условиях и изменять его (с помощью команд `add_score` и `set_score`);
- если счет письма превысит значение, указанное в параметре **MaxScore** секции `[MailD]` конфигурационного файла **Dr.Web MailD**, то проверка письма прерывается и выполняется действие, указанное в значении параметра **MaxScoreAction**.
- в некоторых ограничениях можно выполнять те или иные действия в зависимости от текущего счета письма;
- можно блокировать сессии в **drweb-receiver**, если общий накопленный счет письма превысит порог, указанный в значении параметра **MaxSessionScore** секции `[Receiver]`.
- с помощью **score_filter** из Reputation IP Filter можно фильтровать IP-адреса, чей общий счет слишком велик.

Технология Reputation IP Filter

Репутационный IP-фильтр (Reputation IP Filter) – технология, позволяющая вести историю по каждому из IP-адресов, соединяющихся с **Dr.Web для почтовых серверов UNIX**, и на основе данной истории либо временно блокировать данный IP-адрес, либо предпринимать другие действия. Данная технология позволяет эффективно распознавать распространителей спама, а также бороться с DHA-атаками.

Блок **Reputation IP Filter** включается, если задан хоть один фильтр в настройке **ReputationIPFilter** (одноименный параметр в секции `[Receiver]`), либо если значение параметра **MaxConcurrentConnection** в этой же секции не равно 0. По умолчанию значение параметра **ReputationIPFilter** установлено в `score_filter`, т.е. IP-фильтр включен и IP-адреса будут отфильтровываться на основании среднего значения счета, выставленного всем сообщениям и сессиям с этих IP-адресов (см. ниже).

Вся информация по IP-адресам хранится в оперативной памяти и сбрасывается в файлы. Сохранение в файлы происходит при завершении работы процесса **drweb-receiver**. Чтение файлов происходит только при запуске модуля **drweb-receiver**.

Файлы сохраняются и загружаются, только если в **ReputationIPFilter** есть хотя бы один фильтр. Также сохранения не происходит, если нет никакой информации (не было ни одного IP-соединения). Сохранение происходит в каталог, указанный в параметре **BaseDir** секции `[General]` в файлы `ipv4.bin` (для адресов IPv4) и `ipv6.bin` (для адресов IPv6). Если при сохранении или чтении возникает ошибка, то это будет отражено в журналах.

Информация, сохраняемая в эти файлы, имеет бинарный формат и зависит от системы, на которой работает продукт, поэтому в общем случае их нельзя переносить для использования на другие системы.

Проверка IP-адреса в **Reputation IP Filter** происходит сразу после проверки SMTP-ограничений **SessionRestrictions** в том случае, если IP-адрес не помечен флагом `trusted` (подробнее о SMTP-ограничениях и флаге `trusted` можно узнать в главе Секция [Receiver]).

Таким образом, если необходимо обезопасить некоторые IP-адреса от блокировки в **Reputation IP Filter**, то их необходимо пометить флагом `trusted` в SMTP-ограничениях, заданных в **SessionRestrictions**. Аналогично, если в **Reputation IP Filter** был по ошибке временно заблокирован IP-адрес, то его необходимо пометить как `trusted` в **SessionRestrictions**, и тогда все следующие соединения с этого IP-адреса будут игнорироваться **Reputation IP Filter**.

Репутационный IP-фильтр позволяет выставлять счет IP-адресу на основе набираемой по данному адресу статистики и временно блокировать IP-адрес в случае, если его итоговый счет превышает некоторое пороговое значение.

Доступны следующие фильтры: `anti_dha`, `errors_filter`, `score_filter`.

Reputation IP Filter проверяет IP-адрес сразу после прохождения им проверки **SessionRestriction**, если адрес этот не отмечен как `trusted` (т.е. если в результате проверок SMTP-ограничений, указанных в **SessionRestriction** адрес отмечается как `trusted`, то он не будет проходить через **Reputation IP Filter**).

Фильтры перечисляются через запятую и проверяются в порядке задания. Для каждого фильтра в начале указывается его название, затем перечисляются параметры, разделяемые пробелами (все эти параметры не являются обязательными).

Параметры представляют собой пары **NAME**=`VAL` (между знаком равенства и значением не должно быть пробелов).

Общие параметры для каждого из фильтров (здесь `U` – положительное целое число, `I` – целое число, `D` – положительное число с плавающей точкой):

- **min_msgs**=`U` – минимальное число сообщений, переданных на проверку в **MailD core**, после которого срабатывает фильтр. Если значение равно `0`, то параметр игнорируется.
- **min_errors**=`U` – минимальное число ошибок, зарегистрированных на этапе SMTP-сессии, после которого срабатывает фильтр. Если значение равно `0`, то параметр игнорируется.
- **min_wrong_rcpts**=`U` – минимальное число ошибочных получателей письма (которые были отклонены после SMTP-команды `RCPT TO`), переданных SMTP-клиентом, после которого срабатывает фильтр. Если значение равно `0`, то параметр игнорируется.
- **min_conn**=`U` – минимальное число соединений с этого IP-адреса, после которого срабатывает фильтр. Если значение равно `0`, то параметр игнорируется.
- **block_period**=`T` – задает время блокировки IP-адреса, если он подпадает под ограничения данного фильтра. `T` – имеет тип `{time}`. Если значение установлено в `0`, то блокировки не происходит, даже если IP-адрес подпадает под ограничения фильтра.
- **score**=`I` – счет, который будет выставлен всем сообщениям в данной сессии. Также он будет добавлен к общему счету IP-адреса. Если это значение установлено не в `0`, то при срабатывании фильтра вместо блокировки IP-адреса на время, указанное в значении параметра **block_period**, будет производиться выставление счета, на основе которого можно будет в дальнейшем осуществлять фильтрацию писем и адресов.

Для каждого из имеющихся фильтров имеются свои собственные параметры и значения по умолчанию для общих параметров:

- `anti_dha` – противодействие DHA-атакам (directory harvest attack). Для использования этого фильтра необходимо задать весь список защищаемых адресов (в параметре **ProtectedEmails** в секции `[Receiver]`).

  Специфические параметры:

  - **wrong_per_valid_rcpts**=`D` – отношение ошибочных получателей письма (которые

были отклонены после SMTP-команды `RCPT TO`) к корректным получателям. Основной параметр, который определяет работу фильтра. Если не было найдено ни одного корректного получателя, то это число принимается равным единице. Если значение установлено в `0`, фильтр полностью игнорируется. Значение по умолчанию: `10.0`

Значения по умолчанию для общих параметров:

- **min_msgs**=0
- **min_errors**=0
- **min_wrong_rcpts**=20
- **min_conn**=0
- **block_period**=2h
- **score**=0

- `errors_filter` – позволяет отфильтровывать IP-адреса на основании количества ошибок в SMTP-сессии, которые происходят при общении с данным IP-адресом.

  Специфические параметры:

  - **errors_per_msg**=D – отношение числа ошибок на этапе SMTP-сессии к количеству сообщений, переданных в **MailD core**. Если не было передано ни одного сообщения, то это число принимается равным единице. Если параметр установлен в `0`, то проверка игнорируется. Значение по умолчанию: `0`
  - **errors_per_conn**=D – отношение числа ошибок на этапе SMTP-сессии к числу соединений с этого IP-адреса. Проверка срабатывает только в том случае, если значение параметра установлено не в `0` и было хотя бы одно соединение с данного IP-адреса. Значение по умолчанию: `2.0`

  Если заданы оба параметра, то в начале проверяется **errors_per_msg**, а затем - **errors_per_conn**. Если оба параметра установлены в `0`, то фильтр игнорируется.

  Значения по умолчанию для общих параметров фильтров:

  - **min_msgs**=0
  - **min_errors**=100
  - **min_wrong_rcpts**=0
  - **min_conn**=50
  - **block_period**=2h
  - **score**=0

- `score_filter` – позволяет отфильтровывать IP-адреса на основании среднего значения счета, выставленного всем сообщениям и сессиям с этого IP-адреса. Входит в общую систему **Unified Score** и позволяет, к примеру, блокировать злостных распространителей спама уже на этапе SMTP-соединения.

  Специфические параметры:

  - **score_per_msg**=D – отношение общего счета для данного IP-адреса (сумма всех счетов сообщений, отправленных с IP-адреса, и счетов, выставленных сессиям (например, другими репутационными IP-фильтрами или ограничениями)) к сообщениям, переданным в **MailD core**. Если не было передано ни одного сообщения, то это число принимается равным единице. Если параметр установлен в `0`, то проверка игнорируется. Значение по умолчанию: `0`
  - **score_per_conn**=D – отношение общего счета для данного IP-адреса к числу соединений с этого IP-адреса. Проверка срабатывает только в том случае, если значение параметра установлено не в `0` и было хотя бы одно соединение с этого IP-адреса. Значение по умолчанию: `100.0`

  Если заданы оба параметра, то в начале проверяется **score_per_msg**, а затем – **score_per_conn**. Если оба параметра установлены в `0`, то фильтр игнорируется.

  Значения по умолчанию для общих параметров фильтров:

  - **min_msgs**=0
  - **min_errors**=0
  - **min_wrong_rcpts**=0
  - **min_conn**=100

- **block_period**=2h
- **score**=0

Пример:

```
ReputationIPFilter = errors_filter score=20, score_filter
```

Первый фильтр будет ставить для всех сессий и сообщений в них счет 20 для тех IP-адресов, число ошибок которых на этапе SMTP-сессии слишком велико, второй же фильтр блокирует все IP-адреса, у которых слишком большой средний счет относительно числа соединений с него.

Пример:

```
ReputationIPFilter = errors_filter errors_per_msg=0.05 errors_per_conn=1
min_msgs=0 min_errors=10 min_wrong_rcpts=3 min_conn=50, score_filter
score_per_msg=20 score_per_conn=30 min_wrong_rcpts=3, anti_dha
wrong_per_valid_rcpts=0.02 min_wrong_rcpts=20
```

В данном примере, фильтр `errors_filter` будет срабатывать, если выполняется одно из следующих условий:

- отношение числа ошибок на этапе SMTP-сессии к количеству переданных в **MailD core** сообщений будет равно 0.05 `(errors_per_msg=0.05);`
- отношение числа ошибок на этапе SMTP-сессии к числу соединений с этого IP-адреса будет равно 1 `(errors_per_conn=1);`
- было зарегистрировано более 10 ошибок на этапе SMTP-сессии `(min_errors=10);`
- число ошибочных получателей письма (которые были отклонены после SMTP-команды `RCPT TO`), переданных SMTP-клиентом, равно 3 `(min_wrong_rcpts=3);`
- при 50 и более соединений с данного IP-адреса `(min_conn=50)`.

`score_filter` будет срабатывать если:

- отношение общего счета для данного IP-адреса к количеству сообщений, переданным в **MailD core**, будет равно 20 `(score_per_msg=20);`
- отношение общего счета для данного IP-адреса к числу соединений с этого IP-адреса равно 30 `(score_per_conn=30);`
- число ошибочных получателей письма (которые были отклонены после SMTP-команды `RCPT TO`), переданных SMTP-клиентом равно 3 `(min_wrong_rcpts=3)`.

Фильтр `anti_dha` будет срабатывать если:

- отношение ошибочных получателей письма (которые были отклонены после SMTP-команды `RCPT TO`) к корректным получателям равно 0.02 `(wrong_per_valid_rcpts=0.02);`
- число ошибочных получателей письма (которые были отклонены после SMTP-команды `RCPT TO`), переданных SMTP-клиентом, равно 20 `(min_wrong_rcpts=20)`.

Обратите внимание, что при прохождении IP-адресом проверки **SessionRestriction** на стадии подключения будут учитываться только те значения счетчиков, которые были набраны IP-адресом за предыдущие сессии и за стадию подключения текущей сессии, поэтому счетчик **min_conn** всегда сработает раньше всех. Если IP-адрес прошел проверку SMTP-ограничений, указанных в **SessionRestriction**, но на следующих стадиях сессии значения счетчиков превысили те, что указаны в настройках фильтра **Reputation IP Filter**, то блокировки этого IP-адреса уже не будет до следующей попытки соединения с ним.

Одновременное подключение нескольких компонентов Receiver/Sender

Существует возможность подключать к **drweb-maild** одновременно несколько компонентов **Receiver** и/или **Sender**. Это может понадобиться в следующих случаях:

- для одновременной работы с несколькими MTA или **SMTP/LMTP-прокси**;
- для организации разной настройки каждой пары **Receiver**/**Sender** (что позволит, к примеру, прослушивать разные интерфейсы);
- для получения возможности перенаправлять сообщения из одних MTA в другие (т.е. для маршрутизации).

Обратите внимание, что роль компонентов **Sender** и **Receiver** могут выполнять различные исполняемые модули (например, в качестве **Receiver** может выступать не только модуль **drweb-receiver**, но также **drweb-milter**, **drweb-cgp-receiver**, в зависимости от того, какой способ интеграции с MTA реализован при установке и настройке **Dr.Web MailD**).

Полный перечень модулей, и роли (**Sender**, **Receiver**), которые они исполняют с точки зрения **Dr.Web MailD**, перечислен в разделе Используемые модули.

В данном разделе предполагается, что в роли компонента **Sender** запускается модуль **drweb-sender**, а в роли компонента **Receiver** – **drweb-receiver**.

Возможность запуска одновременно нескольких компонентов осуществляется следующим образом:

- каждому компоненту необходимо присвоить уникальный идентификатор (т.е. у каждого элемента из группы компонентов **Receiver** и группы компонентов **Sender** будет уникальный ID, при этом ID какого-либо **Receiver** должен совпадать с ID какого-либо **Sender**);
- затем каждому компоненту необходимо сообщить, откуда ему получать свои настройки;
- потом каждому письму, принятому компонентом **Receiver**, присваивается в качестве тега уникальный идентификатор данного компонента;
- после обработки сообщения **drweb-maild** ищет доступный компонент **Sender** с тем же идентификатором. Если он не найден, то письмо будет отправлено в **Sender**, заданный по умолчанию (это компонент **Sender**, у которого уникальный идентификатор не задан – такой **Sender** может быть только один), который должен быть всегда доступен.
- список доступных компонентов **Sender** инициализируется при старте и обновляется при получении сигнала `SIGHUP`;
- проблема маршрутизации писем, сгенерированных в **drweb-notifier**, решается при помощи настройки параметра **MsgIdMap** секции `[Notifier]` конфигурационного файла **Dr.Web MailD**. Этот параметр позволяет определять, в какой **Sender** требуется отправлять отчеты в ответ на письма от заданных компонентов **Receiver**.

Уникальный идентификатор задается для **Receiver** и **Sender** через параметр командной строки `--unique-id`. При запуске с заданным параметром компоненты создают в каталоге `%var_dir/msgs/{in|out}` набор подкаталогов для своей очереди писем, а в каталоге `%var_dir/ipc/` для **Sender** создается специальный UNIX-сокет (название каталога и UNIX-сокета определяются на основе заданного уникального идентификатора).

Когда запускается второй экземпляр одного и того же компонента (например, второй экземпляр **Receiver**), то необходимо выполнить дополнительную настройку компонента, т.е. определить, как вторая копия будет получать свою уникальную конфигурацию. Для получения конфигурации можно использовать два способа:

- создавать новую копию конфигурационного файла;
- модифицировать существующий конфигурационный файл.

Последний вариант более прост, но менее гибок.

Чтобы модифицировать существующий *.conf файл, необходимо:

- создать у **Dr.Web Agent** новый `.amc` файл для **Dr.Web MailD** и добавить в него информацию о новой копии компонента. Имя файла может быть произвольным.

  Пример:

```
Application "MAILD"
id                    40
ConfFile              "/etc/drweb/maild_smtp.conf"
Components
drweb-sender2         General, Logging, Sender2
  drweb-receiver2      General, Logging, /Maild/ProtectedNetworks, /
                       Maild/ProtectedDomains,\
                       /Maild/IncludeSubdomains, SASL, Receiver2
drweb-sender3         General, Logging, Sender3
  drweb-receiver3      General, Logging, /Maild/ProtectedNetworks, /
                       Maild/ProtectedDomains,\
                       /Maild/IncludeSubdomains, SASL, Receiver3
drweb-sender4         General, Logging, Sender4
  drweb-receiver4      General, Logging, /Maild/ProtectedNetworks, /
                       Maild/ProtectedDomains,\
                       /Maild/IncludeSubdomains, SASL, Receiver4
drweb-sender5         General, Logging, Sender5
  drweb-receiver5      General, Logging, /Maild/ProtectedNetworks, /
                       Maild/ProtectedDomains,\
                       /Maild/IncludeSubdomains, SASL, Receiver5
```

  Здесь **drweb-receiver*** и **drweb-sender*** – новые названия компонентов, под которым они будут известны **Dr.Web Agent**, а `Receiver*` и `Sender*` – новые названия уникальных копий соответствующих секций в конфигурационном файле.

  Остальной список параметров надо скопировать из настройки оригинального компонента. Подробнее о синтаксисе `*.amc` файлов можно почитать в описании компонента **Dr.Web Agent**.

- сделать копию основной секции настроек компонента в `*.conf` файл, переименовать ее, указав название, которое было задано на предыдущем этапе, и изменить остальные настройки в новой секции для второго компонента по своему усмотрению;
- запустить/перезапустить модуль компонента **Dr.Web Agent**, чтобы он прочитал новую информацию;
- запустить модуль новой копии компонента, указав ему дополнительно параметры командной строки `--unique-id`, `--component`, `--section`.

  Пример:

```
drweb-receiver --unique-id id2 --component drweb-receiver2 --section Receiver2
drweb-sender --unique-id id2 --component drweb-sender2 --section Sender2
```

Создание новой копии конфигурационного файла

При использовании способа, который требует создания новой копии конфигурационного файла, необходимо приложить больше усилий, но при этом появится возможность настраивать произвольно не только параметры основной секции компонента.

Для его реализации потребуется:

- создать копию оригинального основного файла конфигурации **Dr.Web MailD** и настроить в нем параметры по своему усмотрению (обратите внимание, что при этом никаких секций

переименовывать не требуется!);

- создать новый файл `.amc`, в который надо включить только информацию по новому компоненту(-ам). Также в нем надо указать путь к конфигурационному файлу **Dr.Web MailD**, созданному на предыдущем шаге;
- запустить/перезапустить модуль компонента **Dr.Web Agent**, чтобы он прочитал новую информацию;
- запустить модуль новой копии компонента, указав ему дополнительно параметры командной строки `--unique-id`, `--component`.

Пример:

```
drweb-receiver --unique-id id2 --component drweb-receiver2
drweb-sender --unique-id id2 --component drweb-sender2
```

Для обоих способов запуска можно настроить **Dr.Web Monitor** на использование новых компонентов. Для этого в `*.mmc` файл компонента **Dr.Web Monitor** для **Dr.Web MailD** необходимо добавить строки по запуску новых компонентов.

Подробнее о синтаксисе файлов `.mmc` можно почитать в описании компонента **Dr.Web Monitor**.

Оптимизация работы и использования системных ресурсов

Контроль пулов потоков:

Поскольку модули **Dr.Web MailD** используют многопоточную модель при получении, обработке и доставке сообщений, то каждый из модулей по мере необходимости создает определенное количество потоков обработки. Чем больше объем обрабатываемого почтового трафика и чем больше проверок сообщений надо выполнить компонентам (например, проверка для письма большого набора Правил обработки, сканирование содержимого писем антивирусным модулем **Drweb** в режиме «paranoid»), тем большее количество потоков будет создавать каждый компонент. Все потоки, создаваемые компонентами, организованы в пулы потоков. Поведение потоков в пулах регулируется параметрами типа **`PoolOptions`**. Эти параметры также указывают для каждого пула количество потоков в нем (как минимальное `t_min`, так и максимальное `t_max`). По умолчанию для всех пулов потоков всех модулей **Dr.Web MailD** задано значение `auto`.

Значение `auto`, заданное для пула, определяет следующие значения `t_min` и `t_max`:

- Для модулей **`drweb-receiver`** и **`drweb-sender`**: `t_min=2`, `t_max=500`;
- Для других модулей (**`drweb-maild`**, **`drweb-milter`**, **`drweb-notifier`** и т.д.): `t_min=2`, `t_max=1000`.

Ограничение количества потоков в пуле модуля **`drweb-receiver`** не только не позволяет ему создавать новых активных потоков сверх разрешенного количества, но и также влияет на его поведение в ходе SMTP-сессий. Если число подключений со стороны клиентов больше, чем разрешено иметь потоков в пуле, то модуль **`drweb-receiver`** создает максимально разрешенное число активных потоков, а остальные соединения, для которых поток-обработчик уже нельзя создать, помещаются в очередь ожидания. Как только любой из активных потоков-обработчиков освобождается, то он сразу начинает обрабатывать очередное соединение из этой очереди. Поскольку обработка соединений идет асинхронно, то один и тот же активный поток может обрабатывать по нескольку соединений одновременно. Длина очереди клиентских соединений также всегда ограничена максимально разрешенным числом потоков в пуле **`drweb-receiver`**. Таким образом, одновременно **`drweb-receiver`** может «удержать» не более, чем `2*t_max` соединений, при этом часть из них может находиться в очереди ожидания. Как только очередь ожидания будет заполнена, то **`drweb-receiver`** прекращает прием новых соединений и отвечает клиентам ошибкой вида:

```
Server error: 421 3.8 Too many concurrent SMTP connections; please try
again later
```

При больших нагрузках некоторая часть новых соединений в этом случае все же будет обрабатываться, так как освобождающиеся активные потоки компонента сразу будут забирать соединения из очереди ожидания и освобождать место для новых, но в целом в этом случае следует попробовать увеличивать верхнюю границу `t_max` количества потоков в пуле модуля **`drweb-receiver`**, чтобы избежать отказа в обслуживании новых подключений. Для остальных модулей **Dr.Web MailD** увеличение числа потоков в пулах на функциональном поведении не сказывается.

Чтобы контролировать состояние компонентов (количество активных потоков в пулах и длины очередей), рекомендуется периодически отправлять всем процессам **Dr.Web MailD** сигнал **`SIGUSR1`**.

Монитор **Dr.Web Monitor** и агент **Dr.Web Agent**, управляющие работой компонентов **Dr.Web MailD**, сигнал `SIGUSR1` в текущей версии не обрабатывают, поэтому им его посылать нельзя, это приведет к завершению их работы!

Когда компоненты **Dr.Web MailD** получают сигнал **`SIGUSR1`**, они производят сброс статистики по пулам потоков. Сброс статистики производится как в отдельные текстовые файлы, так и в виде записей в журнал (на уровне подробности `Debug`). Местоположение файлов статистики регулируется параметром **`BaseDir`** в секции `[General]`). Подробнее о форматах статистики см. в разделе Внутренняя статистика работы.

Статистика пулов потоков модулей **`drweb-sender`** и **`drweb-receiver`** сохраняется в файлах `sender_thr.txt` и `receiver_thr.txt` соответственно. Статистическая информация содержит данные о текущем размере пула, число активных потоков, и соединений, находящихся в очереди. Максимально разрешенное количество потоков в пуле `t_max` следует увеличивать, когда число соединений, находящихся в очереди `(pending)` приближается к числу активных потоков `(active)`.

Число реально созданных потоков (например, если посчитать их количество в процессах командой **`ps`** `aHx`) всегда будет больше, чем указано в настройках пулов. Это связано с тем, что в процессе работы создаются не только сами потоки-обработчики, но и вспомогательные потоки, а пулы потоков контролируют только потоки-обработчики.

Увеличивать числа потоков в пулах надо с осторожностью. Для этого требуется предварительно оценить:

- Объем потребляемой памяти;
- Количество открываемых файлов и сокетов (т.е. файловых дескрипторов);
- Мощность CPU.

Чем больше для пулов значение `t_min` (минимальное количество потоков в пуле), тем больше времени требуется компонентам **Dr.Web MailD** для запуска и установки начальных соединений. Например, если используется антивирусный модуль **Drweb**, то потоки из его пула потоков при запуске создают соединения с антивирусным сканером **Dr.Web Daemon**, в следствие чего удлиняется время запуска компонента **MailD core**. Если минимальное число потоков в пуле потоков некоторого компонента слишком велико, то при старте он может не успеть запуститься за период тайм-аута **`StartTimeout`**, указанного в его настройках запуска у управляющего компонента **Dr.Web Monitor**. В этом случае **Dr.Web Monitor** аварийно завершит работу как этого компонента, так и всего комплекса **Dr.Web MailD** при старте.

Аналогично, если для пулов некоторого компонента указано большое число `t_max` (максимальное количество потоков в пуле), то могут возникать уже проблемы при завершении работы комплекса **Dr.Web MailD**, когда его компоненты не будут успевать завершать свою

работу за отведенный на это тайм-аут, и в этом случае работа всего комплекса также будет завершаться управляющим компонентом **Dr.Web Monitor** аварийно.

Поэтому не следует увеличивать количество создаваемых потоков "про запас", так как, когда это число достаточно велико (порядка `1000` для модулей **drweb-receiver** и **drweb-sender** и порядка `2000` – для всех остальных модулей), возможно возникновение задержек в при создании новых потоков и появление ошибок, связанных с истечением тайм-аутов для потоков непосредственно при обработке писем, что приводит к ошибкам обработки и даже может повлечь потерю писем. В случае если это происходит, число потоков следует уменьшать. Если все же невозможно уменьшить количество используемых потоков, то необходимо:

1) Увеличить величину тайм-аута подсистемы IPC (регулируется параметром **IpcTimeout** в секции `[General]`), например до 10 минут;
2) Увеличить максимально разрешенное время ожидания закрытия одного потока, которое используется при перезапуске и при завершении работы **Dr.Web MailD** (регулируется параметром **MaxTimeoutForThreadActivity** в секции `[General]`), например до 3 минут;
3) Увеличить тайм-ауты ожидания запуска (**StartTimeout**) и завершения работы (**StopTimeout**) компонентов **Dr.Web MailD** в управляющем файле `maild_<mta>.mmc` монитора **Dr.Web Monitor** (так как большему количеству потоков требуется большее количество времени на остановку).

Также в этом случае крайне желательно провести более тонкую настройку всего комплекса в целом (см. ниже).

Возможные симптомы, указывающие на исчерпание системных ресурсов:

1) Может возникнуть ситуация, когда очередной поток в некотором пуле не может быть создан, что приведет к фиксации в журнале (логе) соответствующего компонента ошибки следующего вида:

   ```
   ERROR <some description>: boost::thread_resource_error
   ```

   В этом случае следует уменьшить количество активных потоков для соответствующего пула потоков, а в случае если количество потоков в пуле выбрано автоматическим `(auto)`, то задать явные количества потоков.

   Если заданного количества потоков недостаточно, а при увеличении количества потоков происходит указанная ошибка, то следует увеличить призводительность сервера, т.е. увеличить объем оперативной памяти и процессоров (ядер), доступных **Dr.Web MailD**.

2) При большой нагрузке могут не приниматься к обработке сообщения, при этом **Dr.Web MailD** будет фиксировать в журнале сообщения вида

   ```
   Too many open files
   ```

   Возникновение ошибки вызвано исчерпанием количества файловых дескрипторов, доступных **Dr.Web MailD** (в том числе – дескрипторов сокетов).

   Для ее решения необходимо (в ОС **Solaris** версии 10) перед запуском модуля **drweb-receiver** выставить переменную окружения `LD_PRELOAD_32` в значение `/usr/lib/extendedFILE.so.1`. Это действие можно выполнить:

   - Непосредственно в консоли, если **drweb-receiver** запускается не через стартовый скрипт, а из консоли;
   - "Завернув" запуск **drweb-receiver** в скрипт-обертку, выполняющий перед запуском модуля задание нужного значения этой переменной окружения;
   - Изменив стартовый скрипт для **drweb-monitor** `(/etc/init.d/drweb-monitor)`, добавив в него соответствующие строки установки значения системной переменной окружения.

Обратите внимание, что в последнем случае переменная окружения будет выставлена не только для **drweb-receiver**, но и для всех процессов **Dr.Web**, запускаемых через **Dr.Web Monitor**.

Если проблема все равно осталась, то, оставив все отписанные выше изменения, необходимо:

- увеличить значения **ulimit** -n;
- добавить (или исправить, если они имеются) следующие строки в файле /etc/system:

```
set rlim_fd_max = 65335
set rlim_fd_cur = 65335
```

При возникновении этой ошибки в других ОС (**FreeBSD** и **Linux**) следует увеличить лимит на число файловых дескрипторов для процесса/пользователя и увеличить значения **ulimit** -n.

Общие рекомендации по повышению производительности и быстродействия:

Для повышения производительности и быстродействия в случае больших нагрузок желательно:

- Использовать асинхронный режим обработки писем, то есть использовать примерно следующую расстановку подключаемых модулей по очередям:

```
BeforeQueueFilters = headersfilter, vaderetro
AfterQueueFilters = drweb, modifer
```

Помещение подключаемых модулей в очередь **BeforeQueueFilters** позволит им синхронно взаимодействовать с модулем **drweb-receiver** и обрабатывать сообщения до помещения их в базу. Но если основное количество писем является "тяжелыми" письмами (с большими по объему вложениями, или с большим количеством малых по объему вложений), то их проверка модулями будет осуществляться долго. В этом случае перемещение модулей в очередь **BeforeQueueFilters** не рекомендуется, поскольку это замедляет взаимодействие с внешними MTA при передаче писем.

Кроме того в этом случае возможно возникновение проблем при проверке писем из-за некорректно заданной (малой) величины тайм-аута **IpcTimeout**, что может привести к их потере (недоставке получателю без соответствующего уведомления об этом отправителя).

Если **Dr.Web MailD** интегрован с каким-либо MTA, то для повышения производительности рекомендуется синхронный режим работы, с указанием подключаемых модулей в списке **BeforeQueueFilters**, а в случае работы в конфигурации SMTP/LMTP-прокси рекомендуется использовать асинхронный режим, т.е. поместить подключаемые модули в список **AfterQueueFilters** (параметры находятся в секции [Filters]).

- Увеличить тайм-ауты:
  - Подсистемы IPC (регулируется параметром **IpcTimeout** в секции [General]);
  - Максимально разрешенное время ожидания закрытия одного потока, которое используется при перезапуске и при завершении работы **Dr.Web MailD** (регулируется параметром **MaxTimeoutForThreadActivity** в секции [General]);
  - Время ожидания запуска и завершения работы компонентов **Dr.Web MailD** в управляющем файле maild_<mta>.mmc монитора **Dr.Web Monitor**.
- Увеличить величины **ulimit** -n.
- Проанализировать загрузку пулов потоков путем сбора статистики (см. выше), если это необходимо, изменить граничные значения для соответствующих пулов потоков.
- Выполнить монтирование каталогов %var_dir/msgs и %var_dir/infected в файловую систему **tmpfs** (командой **mount** -t tmpfs tmpfs <directory>, где <directory> – монтируемый каталог).

Монтирование каталогов в файловую систему **tmpfs** следует выполнять с осторожностью, имея в виду следующие обстоятельства:

- В системе должен быть в наличии достаточный объем ОП;
- При отключении питания в этом случае будут потеряны и внешние очереди сообщений и содержимое **Карантина.**

- Во всех параметрах, использующих Lookup, использовать `Lookup` к файлам (`file:`, `rfile:`), регулярным выражениям (`regex:`) или простые списки (так как обращение к внешним СУБД и LDAP при обработке каждого письма резко снижает скорость обработки, кроме того, успешность обработки ставится в зависимость от стабильности подключения к СУБД или серверу LDAP).
- Установить значение параметра **MoveAll** в секции `[Quarantine]` в значение `No` (особенно если `%var_dir/msgs` и `%var_dir/infected` не смонтированы в **tmpfs**).
- Установить значение параметра **SyncMode** в секции `[MailBase]` в значение `No`.
- Увеличить объем памяти, используемой внутренней БД, для чего увеличить значение параметра **MaxPoolSize** в секции `[MailBase]`, что уменьшит количество обращений к диску.
- Отключить использование статистики и отчетов (установив значения параметров **Detail**=`off` и **Send**=`no` в секции `[Stat]` и секции `[Reports]` соответственно).
- Настроить вывод всех журналов в файлы вместо **syslog.**
- В параметрах **ProtectedNetworks** и **ProtectedDomains** в секции `[Maild]` защищаемые сети и домены перечислить списком (см. выше замечание о рекомендуемом использовании `Lookup`).
- Если нет Правил обработки писем, использующих условия, содержащие поле client-ip, то установить значение параметра **GetIpFromReceivedHeader** в секции `[Maild]` в значение `No`.
- Установить значение параметров: **SkipDSNOnBlock** = `Yes` (в секции `[Maild]`), **SendSDN** = `No` (в секции `[Sender]`), а в настройках действий подключаемых модулей стараться избегать использования опциональных действий `notify` и `redirect`.
- Отключить использование **Карантина** (убрать действие `quarantine` для подключаемых модулей, если оно задано как одно из действий).
- Ограничить максимальный размер сообщений, проверяемых подключаемыми модулями (задав значения параметров **MaxSizeBeforeQueueFilters** и **MaxSizeAfterQueueFilters** в секции `[Filters]`).
- Значения параметров **StalledProcessingInterval** в секции `[Sender]` и секции `[Receiver]` не должны быть меньше значений, заданных по умолчанию (`10m`).

Если в процессе работы **Dr.Web MailD** наблюдаются задержки при отправке сообщений и растет число соединений в очереди пула потоков модуля **drweb-sender**, то следует, в зависимости от настроенного метода доставки (указано в значении параметра **Method** в секции `[Sender]`), выполнить следующую регулировку:

- Для метода доставки `SMTP`:
  - Уменьшить величину тайм-аута **OtherCmdsTimeout** в секции `[Sender]`.
  - Если используется настройка **Router** в секции `[Sender]`, постараться не использовать в этом параметре `Lookup`, обращающихся ко внешним СУБД и LDAP (см. выше замечание о рекомендуемом использовании `Lookup`).
  - Проверить работу MTA, принимающих исходящие от **Dr.Web MailD** письма – как быстро от них приходит ответ при попытке модуля **drweb-sender** соединиться, нет ли ошибок при доставке сообщений.

- Для метода доставки `Pipe`:
  - Проверить работу локального MTA, принимающего исходящие от **Dr.Web MailD** письма. Его демон, отвечающий за локальную доставку сообщений, должен быть настроен корректно.

Если в процессе работы **Dr.Web MailD** растет очередь сообщений, ожидающих доставки (располагается в каталоге `%var/msgs/out`), то рекомендуется попробовать отправить модулю **drweb-sender** сигнал `SIGUSR2` в часы непиковых нагрузок.

Кроме того, вы можете построить кластерное решение, воспользовавшись внутренним проксированием запросов от компонентов **Sender** и **Receiver** к нескольким экземплярам компонента **MailD core**.

Рекомендации по настройке Dr.Web MailD в случае преобладания в обрабатываемом трафике писем большого размера:

1. Рекомендуется воздержаться от использования подключаемого модуля **Dr.Web Modifier** и вообще от любой фильтрации на основе анализа контента (т.е. это и использование настроек **RejectPartCondition**, **AcceptPartCondition**, **MissingHeader** у подключаемого модуля **Dr.Web HeadersFilter**; настройки **RegexForChechecked** у подключаемого модуля **Drweb** и т.д.), т.к. если поиск будет вестись именно внутри вложенных MIME-объектов, это сильно замедлит обработку писем. Подключаемые модули **Vaderetro**, **Dr.Web Modifier** и **Drweb** хранят и тело письма и заголовки при обработке в оперативной памяти, поэтому желательно поместить их в очередь **AfterQueueFilters** (использовать асинхронный режим).
2. Увеличить тай-маут IPC (**IpcTimeout** в секции `[General]`) до 5 минут, если используется подключаемый модуль **Drweb**.
3. Увеличить таймаут на сканирование файлов в настройках **Dr.Web Daemon**, а также **Timeout** в настройках подключаемого модуля **Drweb** (максимум – 10 минут).
4. Смонтировать каталоги писем и **Карантина** (`%var_dir/msgs` и `%var_dir/infected`) в файловую систему **tmpfs**, но только в том случае, если имеется достаточный объем оперативной памяти, т.к. письма имеют большой объем.
5. Указать максимальный размер тела письма, сохраняемого во внутренней БД, в 1 Кб (установить **MaxBodySizeInDB** = `1k` в секции `[MailBase]`).
6. Снять ограничения по объему писем в **Карантине** и объему диска, выделенному под **Карантин** (установить в 0 значения параметров **MaxSize** и **MaxNumber** в секции `[Quarantine]`). Обратите внимание, что если используется DBI, и **Карантин** с большими письмами будет сохраняться в СУБД, то это также вызовет дополнительную нагрузку на сервер, что напрямую не отразится на работе **Dr.Web MailD**, но отразится на величине средней загрузки сервера.
7. Время хранения писем в **Карантине**, если особых условий не требуется, лучше сократить (регулируется параметром **StoredTime** в секции `[Quarantine]`), чтобы они не накапливались и не занимали места.
8. Контролировать объем каталогов `%var_dir/msgs/out` и `%var_dir/msgs/out/failed`, чтобы сразу отслеживать проблемы с доставкой писем для доменов и не забивать жесткий диск.

Для стабильной работы **Dr.Web MailD** необходимо иметь запас оперативной памяти, больший, чем суммарный объем писем за секунду, умноженный среднее время обработки письма в секундах. При этом ограничение числа потоков в настройках пулов потоков компонентов ограничивает суммарный объем писем, обрабатываемых за секунду. Среднее время обработки письма в секундах зависит только от мощности сервера (при фиксированных настройках для подключаемых модулей, Правил обработки и т.п.), поэтому для его необходимо измерять по журналам **Dr.Web MailD** конкретно для текущей архитектуры. Для стабильности число потоков в пулах потоков модулей **drweb-receiver** (**drweb-milter**), **drweb-maild** и **drweb-sender**

должны быть одинаковыми. Однако, если **drweb-sender** использует маршрутизацию, заданную параметром **Router**, у него должно быть больше потоков в пуле, чем у других компонентов.

В случае если будут наблюдаться нехватки времени на обработку писем при установленном в 5 минут **IpcTimeout**, следует выполнить балансировку нагрузки и уменьшать число потоков в пулах компонентов (см. выше).

Использование прокси

Использование прокси, который входит в состав **Dr.Web MailD**, позволяет достичь нескольких целей:

1. Компоненты обработки почтового трафика (**Receiver** и **Sender** для входящего и исходящего трафика соответственно) и центральный компонент проверки почты **MailD core** распределяются по разным хостам, и между этими хостами с помощью компонентов проксирования настраивается взаимодействие. Это позволяет в ряде случаев добиться существенного повышения производительности программного комплекса.
2. Прокси поддерживает соединения по схеме `N:M` с балансировкой нагрузки, что позволяет оптимальным образом распределить ресурсы между различными узлами сети (здесь `N` – кол-во хостов, обрабатывающих почтовый трафик, `M` – кол-во хостов, проверяющих корреспонденцию на наличие вирусов и спама).

Следует учесть, что компонент **MailD core** (модуль **drweb-maild**) в настоящий момент не поддерживает кластерную реализацию и поэтому разные экземпляры этого модуля не могут обмениваться внутренней информацией друг с другом (статистикой, **Карантином**, настройками в базе данных и т.п.).

В результате для каждого из `M` компонентов **MailD core** будет своя статистика, свой **Карантин**, своя внутренняя база данных и свои настройки.

Прокси состоит из двух компонентов: **Proxy client** (модуль **drweb-proxy-client**) и **Proxy server** (модуль **drweb-proxy-server**).

- **Proxy client** работает на узле, где запущены компоненты **Receiver** и **Sender**, и запускается вместо компонента **MailD core**. Остальные компоненты воспринимают **Proxy-client** как **MailD core**, не имея никакого представления о существовании прокси.
- **Proxy server** работает на узле, где запущен основной компонент **MailD core**, и выполняет для него роль компонентов **Receiver** и **Sender**.

Оба компонента прокси взаимодействуют друг с другом, обеспечивая передачу оригинальных сообщений и их модификаций с целью обработки их другими компонентами **Dr.Web MailD**, расположенными на разных хостах.

Обратите внимание, что роль компонентов **Sender** и **Receiver** могут выполнять различные исполняемые модули (например, в качестве **Receiver** может выступать не только модуль **drweb-receiver**, но также **drweb-milter**, **drweb-cgp-receiver**, в зависимости от того, какой способ интеграции с MTA реализован при установке и настройке **Dr.Web MailD**).

Полный перечень модулей, и роли (**Sender**, **Receiver**), которые они исполняют с точки зрения **Dr.Web MailD**, перечислен в разделе Используемые модули.

Компоненты **Dr.Web Notifier**, **Dr.Web Monitor** и **Dr.Web Agent** работают на каждом из хостов.

Общая схема работы с использованием прокси выглядит следующим образом (N=2, M=3):

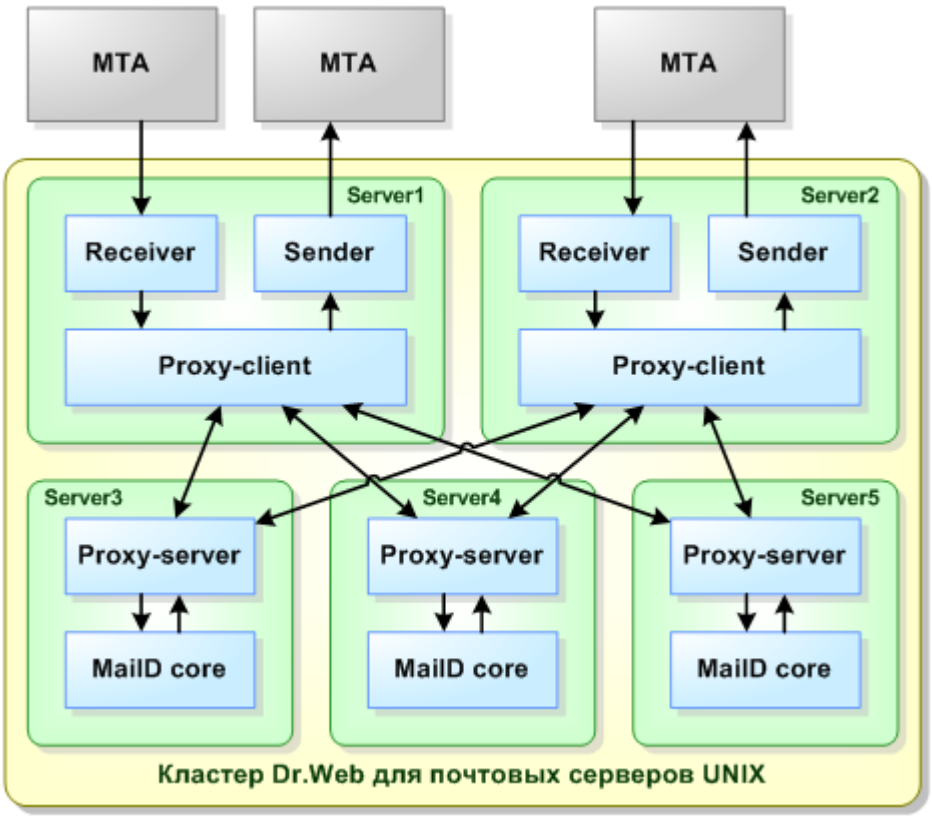


Рис. 18. Схема распределенной работы Dr.Web MailD с использованием прокси

Из данной схемы видно, что как **Proxy client**, так и **Proxy server** способны работать с произвольным числом экземпляров каждого. Обеспечивается это с помощью балансировки соединений через систему весов.

Каждому адресу сокета, указанному в значении параметров **ProxyServersAddresses** из секции [ProxyClient] и **ProxyClientsAddresses** из секции [ProxyServer] присваивается определенный вес. Соответственно, адреса задаются в следующем формате:

ADDRESS1 [WEIGHT1], ADDRESS2 [WEIGHT2] .. – где ADDRESS имеет стандартный тип адреса, а WEIGHT представляет собой необязательный вес этого адреса. Вес может принимать значения от 0 до 100 включительно. Он определяет относительную нагрузку на данный узел по сравнению с остальными узлами: чем больше вес, тем больше будет нагрузка на конкретный сервер.

В файлах конфигурации на каждом из серверов **Server1, Server2** параметр **ProxyServersAddresses** из секции [ProxyClient] задает адреса серверов **Server3, Server4, Server5** (см. схему), на которых работают компоненты **Proxy server**. В файлах конфигурации на каждом из серверов **Server3, Server4, Server5** параметр **ProxyClientsAddresses** из секции [ProxyServer] задает адреса серверов **Server1, Server2** (см. схему), на которых работают компоненты **Proxy client**.

Пример:

```
ProxyServersAddresses = inet:8066@10.3.0.73 10, inet:8066@10.3.0.72 5
```

В этом случае на адрес 10.3.0.73 будет отправляться, в среднем, в два раза больше писем, чем на адрес 10.3.0.72 (по 5 и по 10 из 15 соответственно).

Если вес адреса не указан, то он принимается по умолчанию равным 1. Если для адресов указан одинаковый вес, то они считаются полностью равноправными и получают одинаковый объем

запросов. Если указан вес, равный `0`, то адреса с этим весом считаются запасными (backup-адреса) и на них почта передается, только если не удалось отправить сообщение ни на один адрес с весом, большим или равным `1`.

Общий алгоритм выбора адреса, на который будет осуществляться запрос, следующий:

1) Случайно выбирается адрес в соответствии с весами (чем больше вес адреса, тем вероятнее его выбор).
2) Производится попытка отправки сообщения на выбранный адрес.
3) Если сообщение было успешно отправлено, то конец. Если не удалось передать сообщение на выбранный адрес, то выбирается:
   - либо другой адрес с тем же весом (если такие есть),
   - либо следующий по весу адрес (его вес должен быть не менее 1),

   и переход к пункту 2. Если доступных адресов с весом не менее 1 больше не осталось, то переход к пункту 4.
4) Производятся попытки отправить сообщение на backup-адреса (backup-адреса проверяются последовательно, в порядке их задания в списке). Если и backup-адреса недоступны, то возвращается ошибка.

Выбор веса следует осуществлять на основе имеющихся ресурсов на каждом из узлов, т.е. указывать большие веса тем узлам, которые мощнее.

Когда письма отправляются на проверку модулю **MailD core** и обрабатываются подключаемыми модулями из очереди **`BeforeQueue`**, то обработанная почта отправляется тому клиенту, от которого она была получена. Если письма обрабатываются подключаемыми модулями из очереди **`AfterQueue`**, то адрес клиента для отправки обработанной почты выбирается из **`ProxyClientsAddresses`** согласно заданным для адресов весам. Также клиенту из списка **`ProxyClientsAddresses`** отправляются клонированные письма (см. описание Правил обработки писем), и письма, сгенерированные самим **MailD core** (уведомления, отчеты) – независимо от того, в какой из очередей **`(BeforeQueueFilters`** или **`AfterQueueFilters)`** находятся подключаемые модули. При отправке почты клиентам из списка **`ProxyClientsAddresses`** будут учитываться настройки из Правил обработки писем (настройка **`SenderAddress)`**.

Обратите внимание, что при использовании прокси вместе с почтовыми системами **Qmail, Courier** (и любыми почтовыми системами, использующими протокол `Milter`) нельзя помещать подключаемые модули в очередь **`AfterQueue`**`Filters`, т.к. в настоящий момент прокси не поддерживают callback-соединения с **`drweb-milter`**. Поэтому если ответ не возвращается сразу (а если подключаемый модуль помещен в **`AfterQueueFilters`**, то ответ и не может быть возвращен сразу, т.к. это асинхронный режим работы), то модуль **`drweb-milter`** завершит SMTP-сессию только по истечении периода времени, указанного в значении параметра **`ProcessingTimeout`**.

Ниже описана предпочтительная схема подключения прокси для случая, когда `M=N=1`. Предложенный порядок действий не является единственно возможным, но он позволяет с наибольшей вероятностью избежать различных ошибок настройки.

1. Установить и полностью настроить **Dr.Web для почтовых серверов UNIX** на **Server1** (т.е. на том хосте, через который будет проходить проверяемый трафик и где будет располагаться компонент **Proxy client**). Проверить с помощью команды:

   ```
   /etc/init.d/drweb-monitor check  (для Linux и Solaris)
   /usr/local/etc/rc.d/00.drweb-monitor.sh check (для      FreeBSD)
   ```

   что конфигурация корректна.
2. Запустить **Dr.Web для почтовых серверов UNIX** на **Server1** и проверить, что почта

корректно обрабатывается.

3. Установить **Dr.Web для почтовых серверов UNIX** на **Server2** (т.е. на хост, где будет осуществляться фактическая проверка почты и располагаться компонент **Proxy server**). При установке можно не настраивать запуск модулей, исполняющих роли компонентов **Receiver** и **Sender**, поскольку они тут не понадобятся.
4. Настроить конфигурацию **Server2** аналогично **Server1**.
5. На **Server2** в файле `mmc` (из каталога `%etc_dir/monitor`) **Dr.Web для почтовых серверов UNIX** необходимо закомментировать запуск модулей, исполняющих роли компонентов **Receiver** и **Sender** и раскомментировать запуск модуля компонента **Proxy server** (**`drweb-proxy-server`**).
6. На **Server2** в конфигурационном файле **Dr.Web для почтовых серверов UNIX** надо задать значение параметра **`ProxyClientsAddresse`** из секции `[ProxyServer]`, указав в нем IP-адрес **Server1**, на который будет отправляться почта (тот же, что и в значении параметра **`Address`** секции `[ProxyClient]`).
7. Проверить корректность настройки на хосте **Server2** с помощью команды:

```
/etc/init.d/drweb-monitor check  (для Linux и Solaris)
/usr/local/etc/rc.d/00.drweb-monitor.sh check (для      FreeBSD)
```

   Если все нормально, то можно запускать комплекс. Теперь **Server2** полностью настроен и готов к работе.

8. На **Server1** в конфигурационном файле **Dr.Web для почтовых серверов UNIX** надо задать значение параметра **`ProxyServersAddresses`** из секции `[ProxyClient]`, указав в нем IP-адрес **Server2,** на который будут отправляться запросы на проверку сообщений (тот же, что и в значении параметра **`Address`** секции `[ProxyServer]`).
9. На **Server1** в файле `mmc` **Dr.Web для почтовых серверов UNIX** необходимо закомментировать запуск компонента **MailD core** (модуль **`drweb-maild`**). Там же надо раскомментировать запуск компонента **Proxy client** (модуль **`drweb-proxy-client`**).

   Пожалуйста, обратите внимание, что при попытке одновременно запустить на одном хосте модули **`drweb-proxy-client`** и **`drweb-maild`**, то компонент **Dr.Web Monitor** завершит свою работу и никакие компоненты не будут запущены. Информация об ошибке будет выведена в журнал компонента **Dr.Web Monitor**.

10. Проверить корректность настройки на хосте **Server1** с помощью команды:

```
/etc/init.d/drweb-monitor check  (для Linux и Solaris)
/usr/local/etc/rc.d/00.drweb-monitor.sh check  (для      FreeBSD)
```

    Если все нормально, то можно перезапустить комплекс и, в результате, вся почта для проверки будет переправляться на **Server2.**

11. Опционально на **Server1** теперь можно отключить запуск модуля компонента **Dr.Web Daemon** (**`drwebd`**) и расписание запуска компонента обновления **Dr.Web Updater** (если на данном хосте нет других продуктов **Dr.Web**), так как они больше там не нужны.

Масштабирование для случаев, когда `M` и/или `N` больше `1`, выполняется аналогично: достаточно подключить дополнительные узлы, как было описано выше, и отредактировать соответствующие значения параметров **`ProxyClientsAddresses`** секции `[ProxyServer]` и **`ProxyServersAddresses`** секции `[ProxyClient]` на уже настроенных узлах, установив каждому адресу веса в соответствии с ресурсами узлов.

Обратите внимание на особенность функционирования прокси в случае если прокси-клиентов (т.е. узлов, на которых функционирует компонент **Proxy client**) более одного, и на каком-либо из прокси-клиентов компонент **Receiver** имеет настройку **ReturnReject** = No.

В этом случае если прокси-сервер отвергнет письмо, полученное от этого прокси-клиента, то он сформирует DSN, которое отправит на доставку на какой-либо из прокси-клиентов, выбранный произвольно. В случае если выбранный для отправки DSN прокси-клиент обслуживает сегмент сети, отличный от того, из которого поступило письмо, то отправленное DSN может быть не доставлено отправителю письма по причине недоступности его адреса из этого сегмента сети.

В связи с этим следует избегать использования настройки **ReturnReject** = No при конфигурациях проксирования, имеющих более одного прокси-клиента, если прокси-клиенты обслуживают разные сегменты сети и возможна потенциальная недоставка DSN отправителям писем.

Интеграция с Cyrus SASL

Чтобы реализовать механизм аутентификации SASL клиентов через службу авторизации **Cyrus SASL (saslauthd)**, необходимо выполнить следующие действия:

1. Настроить и запустить службу аутентификации **Cyrus SASL (saslauthd)**.
2. Выполнить настройку **Dr.Web MailD** на использование **Cyrus SASL** (все настройки задаются в секции [SASL] и секции [Cyrus-SASL]):
   - Включить использование аутентификации SASL и использование драйвера **Cyrus SASL**:

```
Use = yes
Driver = cyrus
```

   - В качестве значения параметра **Lib** указать корректный путь к библиотеке **libsasl**;
   - В качестве значения параметра **Path** указать путь к файлу конфигурации аутентификации клиентов сервисом **saslauthd**. Например: /etc/sasl2/maild (обратите внимание, что расширение файла .conf указывать в параметре не нужно). Этот файл должен быть помещен в тот каталог, из которого **saslauthd** читает конфигурацию аутентификации.

Каталог, в котором должен быть расположен конфигурационный фал, зависит от версии **Cyrus SASL** и используемого дистрибутива ОС.

- **Cyrus SASL** версий 2.x пытается найти файл в каталоге /usr/lib/sasl2/
- **Cyrus SASL** версий 2.1.22 и новее также ищет файл в каталоге /etc/sasl2/

Cyrus SASL в первую очередь ищет файл конфигурации в каталоге /usr/lib/sasl2/. Если она обнаружит файл по этому пути, то другие файлы не будут использоваться.

3. Создать и заполнить файл конфигурации аутентификации (в этом примере – /etc/sasl2/maild.conf). Этот файл состоит из строк вида <параметр>: <значение>. Если параметр допускает список значений, то они должны указываться в одну строку, разделенные пробелом. Обязательны следующие параметры:
   - pwcheck_method – Метод проверки пароля (способ аутентификации). Здесь указывается имя модуля, используемого для аутентификации. Возможные значения:

Значение	Используемый источник аутентификации
saslauthd	Непосредственно демон **saslauthd**
auxprop	Модуль, извлекающий атентификационные данные из внешних хранилищ (базы данных, LDAP)

- `mech_list` – Список используемых механизмов аутентификации. Возможные значения: `plain`, `login`, `cram-md5`, `digest-md5`, `ntlm`.

  Обратите внимание, что если в качестве источника аутентификации выбран неспосредственно **`saslauthd`**, то вы можете использовать только механизмы `plain` и `login`.

  Демон **`saslauthd`** может использовать для аутентификации данные, извлекаемые из системного файла `/etc/shadow`, а также механизм PAM или данные IMAP-сервера. О настройке **`saslauthd`** на работу с конкретным источником см. в документации **Cyrus SASL.**

- Если требуется использовать для аутентификации данные, хранящиеся в базах данных или в LDAP, следует использовать `authprop`. В этом случае источник данных указывается в дополнительном параметре `auxprop_plugin`. Возможные значения параметра:

Подключаемый модуль	Назначение
`sasldb`	Использование базы данных **sasldb** (Berkeley DB для Cyrus SASL)
`sql`	Использование баз данных MySQL, PostgreSQL, SQLite
`ldapdb`	Использование LDAP

В случае указания источника `sasldb`, путь к базе данных указывается в параметре `sasldb_path`. По умолчанию, если параметр не указан, используется путь `/etc/sasldb2`.

Если указан источник `sql`, то следующий набор дополнительных параметров определяет использование базы данных:

Параметр	Назначение
`sql_engine`	Определяет тип используемой СУБД: • `mysql` для подключения к MySQL; • `pgsql` для подключения к PostgreSQL; • `sqlite` для подключения к SQLite.
`sql_hostnames`	Определяет хост для подключения к серверу СУБД (имя хоста или пару имя:порт). В случае использования нескольких серверов, адреса можно перечислить через запятую. Примечание: Для СУБД MySQL укажите localhost для подключения через UNIX-сокет или IP-адрес 127.0.0.1 для подключения через TCP-сокет
`sql_user`	Имя пользователя для подключения к базе данных
`sql_passwd`	Пароль пользователя
`sql_database`	Имя используемой базы данных
`sql_select`	SQL-оператор SELECT, который должен извлекать из базы данных пароль пользователя в виде plain text. Важное примечание: Не заключайте текст SQL-выражения в кавычки, а для указания макросов (см. ниже) используйте апострофы. Макросы, предоставляемые для SQL-выражений, представлены ниже. Они будут заменены на значения соответствующих данных, посланных аутентифицируемым клиентом. Доступны следующие макросы: • `%u` – Имя пользователя.

Параметр	Назначение
	• `%r` – Realm (домен), к которому принадлежит пользователь. Это может быть KERBEROS realm, FQDN хоста, на котором запущен SASL, или почтовый домен (часть почтового адреса, указываемая после символа '@')

Если указан источник `ldapdb`, то следующий набор дополнительных параметров определяет использование `LDAP`:

Параметр	Назначение
`ldapdb_uri`	Используемый URI LDAP. Используйте следующие префиксы: • `ldapi://` для подключения через UNIX-сокет • `ldap://` для подключения через незащищенное TCP-соединение • `ldaps://` для использования защищенного TCP-соединения (TLS)
`ldapdb_id`	Логин для аутентификации на LDAP-сервере (прокси-авторизация)
`ldapdb_pw`	Пароль (plain text) для аутентификации на LDAP-сервере (прокси-авторизация)
`ldapdb_mech`	Механизм аутентификации, используемый сервером LDAP
`ldapdb_rc` `(опционально)`	Путь к файлу, содержащему индивидуальные настройки клиента LDAP (libldap). Например, в нем можно определить клиентский сертификат TLS, используемый для защищенного подключения.
`ldapdb_starttls` `(опционально)`	Политика TLS, используемая для подключения к серверу LDAP. Определено два значения – `try` и `demand`. Если указано `try`, то модуль попытается установить защищенное соединение, а если эта попытка не увенчается успехом, перейдет в режим незащищенного соединения . Если указано `demand`, и защищенное соединение установить не удается, соединение с сервером LDAP будет разорвано.

Примеры:

1. Простейшая конфигурация (используется **`saslauthd`**):

```
pwcheck_method: saslauthd
mech_list: plain login
```

2. Использование **sasldb**:

```
pwcheck_method: authprop
auxprop_plugin: sasldb
mech_list: plain login cram-md5
sasldb_path: /etc/sasldb2
```

3. Использование **PosIgreSQL**:

```
pwcheck_method: auxprop
auxprop_plugin: sql
mech_list: PLAIN LOGIN CRAM-MD5 DIGEST-MD5 NTLM
sql_engine: pgsql
sql_hostnames: 127.0.0.1, 192.0.2.1
sql_user: username
sql_passwd: secret
sql_database: dbname
sql_select: SELECT password FROM users WHERE user = '%u@%r'
```

Обратите внимание, что файл конфигурации аутентификации для **Cyrus SASL** можно не создавать, тогда будут использоваться настройки аутентификации и источник аутентификационных данных по умолчанию.

Шаблоны уведомлений

Шаблоны уведомлений представляют собой файлы с расширением `.msg`, хранящие структуру сообщения электронной почты, соответствующую `RFC 822`, и могут содержать различные заголовки. Файлы шаблонов используются компонентом **Notifier** для генерации служебных сообщений – уведомлений MailD, отчетов со статистикой и уведомлений DSN.

В теле шаблона помимо обычного текста могут встречаться макросы (ограниченные знаком $), которые заменяются реальными данными в момент генерации уведомления на основе данного шаблона.

Обработка уведомлений

При обработке письма **Dr.Web MailD** любой из используемых подключаемых модулей может запросить отправление уведомления о каком-либо событии (обнаружении вируса, ошибке обработки, блокировке письма по определенному критерию и т.п.). Данные уведомления формируются компонентом **Notifier** (модуль **drweb-notifier**), который генерирует письмо и затем отправляет его с использованием компонента **Sender**. Кроме того, сам компонент **Sender** может запросить компонент **Notifier** сформировать (для отправки отправителю сообщения) специальное служебное письмо DSN о невозможности доставки письма получателю.

Все уведомления и отчеты, включая DSN, генерируются на основе файлов шаблонов, которые компонент **Notifier** ищет в каталоге, указанном в значении параметра **TemplatesBaseDir**.

Существует три типа уведомлений:

- **Уведомления MailD, отправляемые для конкретного письма**

  Для этих уведомлений **Notifier** проверяет необходимость их отправки с использованием Правил обработки писем (подробнее о них – в описании Правил обработки писем) каждому участнику из перечисленных:

  - Отправителю письма.
  - Получателям письма (при этом, если настройки уведомлений отличаются для разных получателей, то уведомлений будет отправлено больше, чтобы каждый получатель получил уведомление именно в таком виде, в котором задано для него);
  - Администратору **Dr.Web MailD**.

  Имя файла шаблона, используемого для генерации уведомления, образуется прибавлением префиксов `sender_`, `rcpts_` и `admin_` соответственно к названию события, о котором генерируется уведомления, и расширения `.msg`, т.е. соответствует следующему регулярному выражению:

  ```
  (admin|rcpts|sender)_(.*?)\.msg.
  ```

  Например, `sender_virus.msg` – шаблон уведомления отправителю письма о том, что в его письме был обнаружен вирус. Если такой файл шаблона не будет найден, то возникает ошибка. В таблице ниже представлен перечень событий, для которых могут быть сгенерированы уведомления MailD, с указанием суффикса, который следует использовать в именах шаблонов этих уведомлений.

Суффикс	Назначение уведомления
`archive`	Уведомление о том, что в сообщении имеется вложение-архив, не проверенное в связи с нарушением ограничений на проверку архивов, заданных для сканирующего демона **Dr.Web Daemon**

Суффикс	Назначение уведомления
cured	Уведомление о том, что в сообщении имеется вложение, содержавшее вирус, который был успешно излечен
error	Уведомление о том, что при проверке сообщения возникла какая-либо ошибка
license	Уведомление о том, что сообщение не было проверено в связи с нарушением установленных лицензионных ограничений
malware	Уведомление о том, что в сообщении имеется вложение, содержащее вредоносную программу
rule	Уведомление о том, что сообщение было заблокировано из-за срабатывания какого-либо правила (как Правила обработки писем **MailD core**, так и правила проверки какого-либо из подключаемых модулей **Dr.Web Modifier** или **Dr.Web HeadersFilter**)
skip	Уведомление о том, что в сообщении имеется некоторое вложение, которое было пропущено при сканировании (например, запароленный архив или зашифрованный файл)
virus	Уведомление о том, что в сообщении имеется вложение, содержащее вирус

Обратите внимание, что:

- Для некоторых типов событий могут отсутствовать шаблоны уведомлений для некоторых адресатов. Например, по умолчанию в поставку **Dr.Web для почтовых серверов UNIX** включается отчет о событии **Skip** только для отправителя (`sender_skip.msg`). В случае необходимости другие отчеты об этом событии можно получить, скопировав этот шаблон и соответствующим образом его переименовав (`rcpts_skip.msg` и `admin_skip.msg` для получателей и администратора соответственно). Однако дополнительно рекомендуется выполнить модификацию этих шаблонов, так, чтобы они содержали информацию, требуемую соответствующему адресату уведомления. Перечень шаблонов, доступных по умолчанию, перечислен ниже.
- Если по результатам проверки одного письма возникло сразу несколько событий, то компонент **Notifier** отправит по одному уведомлению на каждое событие каждому типу адресата, для которого это уведомление разрешено.
- Имеется возможность запретить отправку уведомлений разных типов разным типам адресатов в зависимости от результатов проверки условий, используя настройку **notify** в Правилах обработки писем **MailD core.**

- **Периодические уведомления MailD об общей работе комплекса (отчеты администратору)**

  Уведомления этого типа **Notifier** отправляет администратору. Эти уведомления содержат общую статистику по работе комплекса. Шаблон для отчета содержится в файле `report.msg`.

- **Служебные уведомления о невозможности доставки письма (DSN)**

  Эти уведомления являются почтовыми сообщениями специального формата о невозможности доставки письма получателю. Они всегда отправляются только отправителю письма и имеют пустой заголовок `FROM:`. Их шаблон содержится в файле `dsn.msg`.

Во всех случаях компонент комплекса **Dr.Web для почтовых серверов UNIX**, затребовавший отправку уведомления, отправляет модулю **drweb-notifier** тип уведомления, которое требуется отправить. Все шаблоны, кроме DSN-шаблонов, по умолчанию поддерживают письма двух видов: в HTML и в виде текста (plain text). Выбор типа письма уведомления происходит на основе значения параметра **html**, задаваемого в Правилах обработки писем.

Обратите внимание, что уведомления MailD и периодические отчеты будут отправляться получателям с адреса, указанного в параметре **FilterMail**, при этом они, также, как и DSN, будут перед отправкой проходить проверку на срабатывание Правил.

Обратите внимание, что уведомления MailD получателям и отправителям писем, а также администратору, которые отсылаются компонентом **Notifier**, отправляются всегда как письма с адреса, заданного в параметре **FilterMail**. При этом служебные уведомления DSN всегда имеют пустое `From:`.

Существует возможность менять имя файла, из которого берется шаблон уведомления, в зависимости от различных критериев. Для этого в Правилах обработки писем введен параметр настройки **NotificationNamesMap**, который определяет схему отображения имени уведомления, переданного в **Notifier**, в новое значение, из которого затем будет сформировано новое имя файла с шаблоном по вышеописанной схеме. Отображение имеет смысл производить только в имя, известное компоненту **Notifier**, так как в противном случае требуемый файл не будет найден. Эта ситуация будет являться ошибочной и будет обрабатываться в соответствии со значением параметра **ProcessingError**.

Обратите внимание, что с помощью механизма **NotificationNamesMap** в Правилах обработки писем можно настраивать выбор разных пользовательских файлов шаблонов только для уведомлений второго и третьего типов, т.е. только для периодических отчетов и DSN.

Пример:

```
[Rule:buh]
...
NotificationNamesMap = report r1, dsn d1
...
[Rules]
to:regex:*@buh.domain.org cont rule=buh
```

Срабатывание этого Правила обработки писем приведет к тому, что в случае если письма будут поступать получателям из домена `buh.domain.org`, в качестве уведомления второго типа (периодического отчета администратору) будет использоваться файл `report_r1.msg`, а для DSN – файл `dsn_d1.msg`.

Пожалуйста, обратите внимание, что в состав поставки **Dr.Web MailD** кроме стандартного файла шаблона DSN-уведомлений `dsn.msg` также включен дополнительный файл `dsn_for_exchange.msg`. Это специальная версия DSN-уведомления, которая должна использоваться, только если в качестве целевого MTA используется почтовый сервер **MS Exchange** (это связано с особенностями реализации **MS Exchange**, которая не полностью соответствует требованиям `RFC 3464`).

Это DSN-уведомление **нельзя использовать с другими MTA**. Штатный режим его использования, если оно требуется – это замена стандартного шаблона `dsn.msg` на модифицированный, например, при помощи команды:

```
cp dsn_for_exchange.msg dsn.msg
```

Этот способ позволит избежать перенастройки компонента генерации уведомлений **Notifier**.

Однако, обратите внимание, что на случай, если в дальнейшем Вам потребуется использовать стандартный шаблон `dsn.msg`, то рекомендуется сохранить его копию перед заменой.

Кроме этого, Вы можете воспользоваться рассмотренным выше механизмом замены имени шаблона для DSN, используемого компонентом **Notifier**, при помощи создания Правила, меняющего значение параметра **NotificationNamesMap**. Этот способ позволит не выполнять замену стандартного файла шаблона `dsn.msg` на нестандартный.

Перечень шаблонов, доступных по умолчанию

По умолчанию в поставку **Dr.Web для почтовых серверов UNIX** включены следующие файлы шаблонов:

Имя шаблона	Описание
Шаблоны уведомлений и отчетов, отправляемых администратору:	
ADMIN_ARCHIVE.msg	Шаблон уведомления об ошибке сканирования письма с архивом во вложении, если на этот архив распространяются ограничения, заданные в главном конфигурационном файле drweb32.ini
ADMIN_CURED.msg	Шаблон уведомления об успешном излечении зараженного письма
ADMIN_ERROR.msg	Шаблон уведомления об ошибке в работе **Dr.Web Daemon** или подключаемого модуля
ADMIN_LICENSE.msg	Шаблон уведомления об ошибке сканирования письма, возникшей в связи с ограничениями, налагаемыми лицензией
ADMIN_MALWARE.msg	Шаблон уведомления об обнаружении в письме вредоносных программ
ADMIN_RULE.msg	Шаблон уведомления о блокировке письма в соответствии с заданным Правилом обработки
ADMIN_VIRUS.msg	Шаблон уведомления об обнаружении в письме вирусов
Шаблоны уведомлений, отправляемых получателю письма:	
RCPTS_MALWARE.msg	Шаблон уведомления об обнаружении в письме вредоносных программ
RCPTS_VIRUS.msg	Шаблон уведомления об обнаружении в письме вирусов
Шаблоны уведомлений, отправляемых отправителю письма:	
SENDER_ARCHIVE.msg	Шаблон уведомления об ошибке сканирования письма с архивом во вложении, если на этот архив распространяются ограничения, заданные в главном конфигурационном файле drweb32.ini
SENDER_CURED.msg	Шаблон уведомления об успешном излечении зараженного письма
SENDER_ERROR.msg	Шаблон уведомления об ошибке в работе **Dr.Web Daemon** или подключаемого модуля
SENDER_MALWARE.msg	Шаблон уведомления об обнаружении в письме вредоносных программ
SENDER_VIRUS.msg	Шаблон уведомления об обнаружении в письме вирусов
SENDER_SKIP.msg	Шаблон уведомления об ошибке сканирования письма. Успешному сканированию могут препятствовать защищенные паролем архивы или файлы нестандартных форматов во вложении. Также сканирование может быть прервано по истечении максимального времени ожидания ответа от **Dr.Web Daemon** или подключаемых модулей
Прочие шаблоны:	
DSN.msg	Шаблон уведомления о доставке письма (DSN)
REPORT.msg	Шаблон для регулярных отчетов **Dr.Web Daemon**

Макросы, используемые в шаблонах

Во всех шаблонах могут быть использованы следующие макросы:

Макрос	Назначение
$LC*$	Заменяется на строку текста из языкового файла с указанным номером (* – десятичный номер строки, например – $LC150$). Используемый языковой файл определяется значением макроса $LANG$. Полученный текст преобразуется в нужную кодировку согласно значениям макросов $CHARSET$ и $CONTENT_TRANSFER_ENCODING$
$POSTMASTER$	Содержит адрес, на который отсылаются уведомления (используется значение параметра конфигурации AdminMail из секции [Notifier])
$FILTER_MAIL$	Содержит адрес, используемый **Dr.Web MailD** (используется

Макрос	Назначение
	значение параметра конфигурации FilterMail из секции [Notifier])
$HOSTNAME$	Содержит имя хоста, на котором установлен **Dr.Web MailD** (используется значение параметра конфигурации Hostname из секции [General]). Данный макрос не может использоваться в циклах, см. Управляющие конструкции
$LANGS$	Содержит список языков, для которых будут формироваться уведомления (задается перечнем значений параметра конфигурации NotifyLangs из секции [Notifier]). Данный макрос имеет списковый тип и может использоваться в циклах, см. Управляющие конструкции
$LANG$	Содержит имя языка, который используется для формирования данной части уведомления. От значения этого макроса зависит интерпретация некоторых других макросов (например, $CHARSET$)
$CHARSET$	Содержит набор символов текущего используемого языка. Набор символов для конкретного языка задается в языковом файле. Имя текущего языка берется из макроса $LANG$
$CONTENT_TRANSFER_ENCODING$	Содержит Content-Transfer-Encoding для текущего языка. Значение для конкретного языка задается в языковом файле. Имя текущего языка берется из макроса $LANG$
$TYPE$	Содержит тип содержимого уведомления (HTML или PLAIN). Определяется значением параметра html в Правилах обработки писем.
$FULLHEADERS$	Содержит полный набор заголовков сообщения электронной почты.
$MSGID$	Содержит внутренний идентификатор сообщения в MTA, из которой пришло письмо
$SUBJECT$	Содержит тему сообщения (если тема не указана – пустой). При вставке в генерируемое сообщение, при необходимости, вставляемое значение преобразуется в зависимости от текущих значений макросов $CHARSET$ и $CONTENT_TRANSFER_ENCODING$
$DIRECT_SUBJECT$	Содержит тему сообщения (если тема не указана – пустой). Не подвергается преобразованию кодировки и CTE при вставке
$SENDER$	Содержит адрес оригинального отправителя сообщения
$RCPTS$	Содержит список адресов всех получателей оригинального сообщения. Данный макрос имеет списковый тип и может использоваться в циклах, см. Управляющие конструкции
$SECURE_RCPTS$	Идентичен $RCPTS$, если получатель один, или устанавливается в значение "Recipients of original message" <#@[]>, если получателей более одного
LOG_REPORT	Содержит записи из журнала **Dr.Web MailD**, касающиеся обработки сообщения, относительно которого генерируется уведомление
$STOP_REASON$	Содержит запись из журнала **Dr.Web MailD** с основной причиной, по которой было отправлено данное уведомление
$REPORT$	Содержит отчет от подключаемого модуля об анализе сообщения, относительно которого генерируется уведомление

Макрос	Назначение
$MESSAGE_STATUS$	Состояние исходного сообщения, указываемое фильтрами POP3 и IMAP, согласно результатам его обработки. Может принимать значения reject, discard, tempfail и error.
$BLOCK_LIST$	Содержит список строк, описывающих причины блокировки исходного сообщения подключаемым модулем (их может быть больше одной). Например, модуль **Drweb**, в случае если был обнаружен вирус (или другая известная угроза), вернет его имя. Если блокировка была вызвана по другой причине (например, по выполнению реакции на событие **SkipObject**), то он возвращает полное значение строки конфигурации <параметр> = <значение>, выполнение которой привело к блокировке письма
$SCAN_STAT$	Содержит статистику проверки сообщения от подключаемого модуля
$ARCHIVE_RECORD$	Содержит имя файла в **Карантине**
$ORIGINAL_MESSAGE$	Содержит тело оригинального сообщения, для которого формируется уведомление. Будьте внимательны, вставляя его в уведомление, т.к. если, например, в нем содержался вирус, то сформированное уведомление может быть заблокировано другой антивирусной системой!
R_MAILS	Содержит перечень адресов, на которые надо отправлять уведомления. Значение макроса берется из параметра конфигурации Mail в секции [Reports]. Данный макрос имеет списковый тип и может использоваться в циклах, см. Управляющие конструкции
CO_CLIENT_IP	Содержит IP-адрес клиента, передавшего сообщение (если известен)
CO_CLIENT_PORT	Содержит номер порта, использованного клиентом, передавшим сообщение (если компонент **Receiver** предоставил эту информацию)
CO_AUTH	Содержит yes, если клиент, отправивший исходное письмо, успешно прошел авторизацию (если компонент **Receiver** предоставил эту информацию)
$CO_SERVER_UNIX_SOCK$	Содержит имя UNIX-сокета, который был использован компонентом **Receiver** для приема исходного письма (если компонент **Receiver** предоставил эту информацию)
CO_SERVER_IP	Содержит IP-адрес прослушивающего сокета, который был использован компонентом **Receiver** для приема исходного письма (если компонент **Receiver** предоставил эту информацию)
CO_SERVER_PORT	Содержит порт прослушивающего сокета, который был использован компонентом **Receiver** для приема исходного письма (если компонент **Receiver** предоставил эту информацию)
CO_RS_ID	Содержит идентификатор компонента **Receiver**, который принял исходное сообщение (если этот экземпляр компонента **Receiver** был запущен с непустым идентификатором)
$CO_SENDER_ADDRESS$	Содержит адрес, указанный в значении параметра SenderAddress из Правил для данного сообщения
$Q_CONTROL_BY_EMAILS$	Содержит yes, если разрешены управляющие сообщения для управления **Карантином**
$TEMPLATES_DIR$	Содержит название каталога, содержащего шаблоны уведомлений. Этот же путь используется для поиска всех файлов, указанных в шаблоне в директиве include, см. Управляющие конструкции
Q_REMOVE_TIME	Содержит время удаления сообщения из **Карантина** (пустая строка, если время хранения сообщения не ограничено)

Макрос	Назначение
$PRODUCT$	Содержит строку "MailD"
$EXT_PRODUCT$	Содержит строку "for Unix mail servers"

Макросы, используемые в отчетах со статистикой

Макрос	Назначение
$R_PLUGINS$	Содержит перечень названий подключаемых модулей, для которых формируются отчеты о статистике. Значение макроса берется из параметра конфигурации Names в секции [Reports]. Данный макрос имеет списковый тип и может использоваться в циклах, см. Управляющие конструкции
R_PERIOD	Содержит период времени, за который формируется отчет со статистикой
RP_NAME	Содержит имя подключаемого модуля, для которого формируется отчет со статистикой
$RP_BLOCKED_OBJECTS_WITH_NUM_AND_PERCENTS$	Содержит вычисленную статистику блокирующих объектов для подключаемого модуля, указанного в макросе RP_NAME
$RP_SENDERS_ENVELOPE_WITH_NUM_AND_PERCENTS$	Содержит статистику отправителей блокирующих объектов для подключаемого модуля, указанного в макросе RP_NAME
$RP_CLIENT_IP_WITH_NUM_AND_PERCENTS$	Содержит статистику IP-адресов, заблокированных подключаемым модулем, указанным в макросе RP_NAME
$RP_BLOCKED_OBJECTS_NUM$	Содержит число объектов, которые будут выведены в статистике блокирующих объектов. Если значение равно 0,то не будет выведено ни одного объекта, при –1 будут выведены все блокирующие объекты, иначе выводится заданное число блокирующих объектов. Зависит от значения макроса RP_NAME
$RP_SENDERS_ENVELOPE_NUM$	Содержит число отправителей, которые будут выведены в статистике отправителей блокирующих объектов. Если значение равно 0,то не будет выведено ни одного отправителя, при –1 будут выведены все отправители блокирующих объектов, иначе выводится заданное число отправителей блокирующих объектов. Зависит от значения макроса RP_NAME
$RP_CLIENT_IP_NUM$	Содержит число IP-адресов, которые будут выведены в статистике заблокированных IP-адресов. Если значение равно 0, то не будет выведено ни одного IP-адреса, при -1 выводятся все IP-адреса, иначе выводится заданное число IP-адресов. Зависит от значения макроса RP_NAME
$RP_TEMPFAIL_SIZE$	Содержит общий размер сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие tempfail
RP_PASS	Содержит общее число сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие pass
RP_REJECT	Содержит общее число сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было

Макрос	Назначение
	выполнено действие reject
$RP_DISCARD$	Содержит общее число сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие discard
$RP_TEMPFAIL$	Содержит общее число сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие tempfail
$RP_REJECT_PLUS_TEMPFAIL$	Содержит общее число сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, были выполнены действия reject или tempfail
$RP_QUARANTINE$	Содержит общее число сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие quarantine
$RP_REDIRECT$	Содержит общее число сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие redirect
RP_NOTIFY	Содержит общее число сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие notify
RP_PASS_SIZE	Содержит общий размер сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие pass
RP_REJECT_SIZE	Содержит общий размер сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие reject
$RP_DISCARD_SIZE$	Содержит общий размер сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие discard
$RP_REJECT_PLUS_TEMPFAIL_SIZE$	Содержит общий размер сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие reject или tempfail
$RP_QUARANTINE_SIZE$	Содержит общий размер сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие quarantine
$RP_REDIRECT_SIZE$	Содержит общий размер сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие redirect
RP_NOTIFY_SIZE	Содержит общий размер сообщений, для которых подключаемым модулем, имя которого указано в макросе RP_NAME, было выполнено действие notify
RP_BLOCK_PERC	Содержит общий процент сообщений, заблокированных подключаемым модулем, имя которого указано в макросе RP_NAME
RP_BLOCK_SIZE	Содержит общий размер сообщений, заблокированных подключаемым модулем, имя которого указано в макросе RP_NAME
$RP_CHECK_TIME_SUM$	Содержит общее время проверки сообщений подключаемым модулем, имя которого указано в макросе RP_NAME
$RP_CHECK_TIME_AVR$	Содержит среднее время проверки сообщений подключаемым модулем, имя которого указано в макросе RP_NAME
$RP_CHECKED_MSGS$	Содержит общее число сообщений, проверенных подключаемым модулем, имя которого указано в макросе RP_NAME

Макрос	Назначение
$RP_CHECKED_SIZE$	Содержит общий размер всех сообщений проверенных подключаемым модулем, имя которого указано в макросе RP_NAME
$RP_AGENT_STAT_UUID$	Содержит UUID ключа, который **Dr.Web Agent** использует для отправки статистики на сервера статистики **Dr.Web** или серверу централизованной защиты (содержит пустую строку, если данная функция отключена)

Управляющие конструкции

В шаблонах уведомлений могут использоваться следующие управляющие конструкции:

1. Объявление собственных макросов

В случае необходимости можно определять собственные макросы прямо в шаблоне. Для этого используется конструкция

```
<def NAME=DATA>
```

где NAME – имя макроса (без символа$), DATA – новое значение макроса.

Имя макроса может состоять из символов диапазона [a-zA-Z_-]. Новое значение макроса можно заключать в кавычки. Символ, следующий сразу за косой чертой "\", воспринимается непосредственно (т.е. таким образом можно добавить, к примеру, обратную угловую скобку).

Пример:

```
<def MY_MACROS="\<my macros\>">
```

В данном случае определяется макрос MY_MACROS, содержащий '<my macros>'.

2. Включение содержимого внешних файлов

В шаблон может быть включено содержимое внешнего файла, для чего используется директива

```
<include FILENAME>
```

где FILENAME – имя и путь к файлу относительно значения, заданного в макросе $TEMPLATES_DIR$. Имя и путь к файлу можно заключать в кавычки. Символ, следующий сразу за косой чертой "\", воспринимается непосредственно (т.е. таким образом можно добавить, к примеру, обратную угловую скобку).

Пример:

```
<include "style.css">
```

В данном случае при генерации письма вместо директивы будет вставлен текст из файла style.css.

3. Условные операторы

В шаблонах можно использовать условные операторы. Они определяются следующей конструкцией:

```
<if NAME [ ( '==' | '!=' ) DATA ] >
TEXT
</if>
```

где NAME – имя макроса, DATA – регулярное выражение для проверки его значения (используются регулярные выражения **Perl),** а TEXT – текст, который будет вставлен в письмо в случае истинности условия. В условиях используются два типа операторов сравнения:

- == – Истина, если значение макроса соответствует регулярному выражению;
- != – Истина, если значение макроса не соответствует регулярному выражению.

Если часть конструкции, приведенная в квадратных скобках, не указана (т.е. указывается условие <if NAME>), то это аналогично записи: <if NAME != "" >.

Допускается использование вложенных условных операторов и циклов (см. ниже). Если внутри оператора находится директива определения нового макроса def, то новый макрос будет определяться, только если условный оператор выполняется. При этом новое значение макроса сохранится и после выхода из условного оператора и будет доступно для использования ниже.

Пример:

```
<def N="n123">
<if N>N is not empty!</if>
<if N == "n.*">N stars with n!</if>
<if N != n123>WRONG!<\if>
```

В письме, сгенерированном на основе такого шаблона, окажутся строки:

```
N is not empty!
N starts with n!
```

3. Циклы

В шаблонах также могут быть использованы циклы. Цикл описывается конструкцией вида:

```
<for NAME;LIST [ ( '==' | '!=' ) DATA ; [DELIM] ] >
TEXT
</for>,
```

Где

- NAME – имя макроса, который будет использован в роли локальной переменной в теле цикла. На каждой итерации цикла ему будет присвоено очередное значение, извлеченное из LIST.
- LIST – Макрос, используемый в качестве списка значений для локальной переменной цикла. Макрос спискового типа. В данном случае может быть указан любой макрос, при этом, если он имеет не списковый тип, то будет преобразован к нему посредством разбивки строки по запятым.
- DATA – Регулярное выражение для проверки и выбора значений из списка на каждой итерации цикла.
- DELIM – разделитель, вставляемый в тело генерируемого письма между фрагментами текста, сгенерированными на каждой итерации цикла.

Если значения, приведенные в квадратных скобках, не указаны, это аналогично записи <for NAME;LIST == ".*" >, т.е. из LIST берутся все значения.

В противном случае каждое значение из LIST сравнивается с регулярным выражением,

указанным в DATA, и если результат сравнения истинный (== – соответствует, а != – не соответствует), то очередное значение присваивается макросу NAME и происходит итерация цикла. Если указан разделитель DELIM, то между каждой итерацией в генерируемое письмо вставляется значение, указанное как разделитель.

Обработка циклов происходит следующим образом:

1) Для каждого выбранного NAME генерируется текст

```
<def NAME="LIST_VAL">
TEXT DELIM,
```

   где LIST_VAL – следующее значение, выбранное из LIST, а TEXT – текст, указанный в теле цикла.

2) Для этого текста вызывается синтаксический анализатор, результат работы которого помещается в генерируемое письмо сразу после метки <\for>.
3) Данная операция выполняется для каждого выбранного значения из LIST.
4) Конструкция цикла удаляется из генерируемого письма и синтаксический анализатор начинает повторно разбирать сформированный циклом текст.

Пример:

```
<def RCPTS="root@localhost, test@mydoamin.com">
<for RCPT;RCPTS==".*";", "><a href="mailto:$RCPT$">$RCPT$</a></for>
```

В результате обработки этой конструкции будет получен следующий текст:

```
<a href="mailto:root@localhost">root@localhost</a>,
<a href="mailto:test@mydoamin.com">test@mydoamin.com</a>
```

Все ключевые слова def, if и for являются регистронезависимыми.

Пример шаблона

Пример шаблона для общего отчета о работе подключаемых модулей, поддерживающего форматы как HTML, так и PLAIN:

```
From: "DrWeb-$PRODUCT$" <$FILTER_MAIL$>
To: $R_MAILS$
Subject: Report from Dr.Web $PRODUCT$ per period of $R_PERIOD$
Content-Type: multipart/mixed;
 boundary="001-DrWeb-MailFilter-Notification"
MIME-Version: 1.0

<if TYPE==HTML>
<for LANG;LANGS>
--001-DrWeb-MailFilter-Notification
Content-Type: text/html; charset=$CHARSET$
Content-Transfer-Encoding: $CONTENT_TRANSFER_ENCODING$

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://
www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=$CHARSET$" />
<title>$LC510$</title>
<include "style.css">
</head>

<body>
  <div align="center">
    <table width="600" border="0" cellspacing="0" cellpadding="0">
      <tr>
        <td align="center" valign="top"><a name="top" id="top_$LANG$"></a>
          <include "head.templ">
          <table width="100%">
            <tr>
              <td align="right" ><for RP_NAME;R_PLUGINS==".*";" "><a
href="#$RP_NAME$_$LANG$" class="ancherlinks">$LC543$ $RP_NAME$</a></for></
td>
            </tr>
          </table>
          <p class="titletext">$PRODUCT$: $LC542$ $R_PERIOD$</p>

          <for RP_NAME;R_PLUGINS>
          <table width="100%">
            <tbody>
              <tr>
                <th colspan="2" class="header"><a name="$RP_NAME$"
id="$RP_NAME$_$LANG$"></a>$LC543$ $RP_NAME$</th>
              </tr>
              <tr>
                <td colspan="2" ><table width="100%" >
                    <tr>
                      <if RP_BLOCKED_OBJECTS_WITH_NUM_AND_PERCENTS><if
RP_BLOCKED_OBJECTS_NUM!="0">
                        <td valign="top"><table width="300"
cellpadding="5" cellspacing="0"  class="statistic" id="statistic2">
                          <tbody>
                            <tr>
                              <th colspan="2" class="statisticheader" >
                                <if RP_BLOCKED_OBJECTS_NUM=="-1">
                                  $LC544$:
                                </if><if RP_BLOCKED_OBJECTS_NUM!="-1">
                                  $LC545$ $RP_BLOCKED_OBJECTS_NUM$
$LC546$:
                                </if>
                              </th>
                            </tr>
                            <tr>
                              <td>$RP_BLOCKED_OBJECTS_WITH_NUM_AND_PERC
ENTS$</td>
```

```
                                </tr>
                              </tbody>
                            </table></td>
                        </if></if>
                        <if RP_SENDERS_ENVELOPE_WITH_NUM_AND_PERCENTS><if
RP_SENDERS_ENVELOPE_NUM!="0">
                            <td valign="top"><table width="300"
cellpadding="5" cellspacing="0"  class="statistic" id="statistic">
                              <tbody>
                                <tr>
                                  <th colspan="2" class="statisticheader" >
                                    <if RP_SENDERS_ENVELOPE_NUM=="-1">
                                      $LC547$:
                                    </if><if RP_SENDERS_ENVELOPE_NUM!="-1">
                                      $LC545$ $RP_SENDERS_ENVELOPE_NUM$
$LC548$:
                                    </if>
                                    </th>
                                </tr>
                                <tr>
                                  <td
class="regulartext">$RP_SENDERS_ENVELOPE_WITH_NUM_AND_PERCENTS$</td>
                                </tr>
                              </tbody>
                            </table></td>
                        </if></if>
                        <if RP_CLIENT_IP_WITH_NUM_AND_PERCENTS><if
RP_CLIENT_IP_NUM!="0">
                            <td valign="top"><table width="300"
cellpadding="5" cellspacing="0"  class="statistic" id="statistic">
                              <tbody>
                                <tr>
                                  <th colspan="2" class="statisticheader" >
                                    <if RP_CLIENT_IP_NUM=="-1">
                                      $LC566$:
                                    </if><if RP_CLIENT_IP_NUM!="-1">
                                      $LC545$ $RP_CLIENT_IP_NUM$ $LC565$:
                                    </if>
                                    </th>
                                </tr>
                                <tr>
                                  <td
class="regulartext">$RP_CLIENT_IP_WITH_NUM_AND_PERCENTS$</td>
                                </tr>
                              </tbody>
                            </table></td>
                        </if></if>
                      </tr>
                  </table></td>
              </tr>
              <tr>
                <td class="body">$LC550$:</td>
                <td align="right" class="body">$RP_PASS$ ($RP_PASS_SIZE$)</
td>
              </tr>
              <tr>
                <td class="body">$LC551$:</td>
                <td align="right" class="body">$RP_REJECT$
($RP_REJECT_SIZE$)</td>
              </tr>
              <tr>
                <td class="body">$LC552$:</td>
                <td align="right" class="body">$RP_DISCARD$
($RP_DISCARD_SIZE$)</td>
              </tr>
```

```
              <tr>
                <td class="body">$LC553$:</td>
                <td align="right" class="body">$RP_TEMPFAIL$
($RP_TEMPFAIL_SIZE$)</td>
              </tr>
              <tr>
                <td class="body">$LC554$:</td>
                <td align="right" class="body">$RP_QUARANTINE$
($RP_QUARANTINE_SIZE$)</td>
              </tr>
              <tr>
                <td class="body">$LC555$:</td>
                <td align="right" class="body">$RP_REDIRECT$
($RP_REDIRECT_SIZE$)</td>
              </tr>
              <tr>
                <td class="body">$LC556$:</td>
                <td align="right" class="body">$RP_NOTIFY$
($RP_NOTIFY_SIZE$)</td>
              </tr>
              <tr>
                <td class="subtitle">$LC557$:</td>
                <td align="right" class="subtitle">$RP_CHECKED_MSGS$
($RP_CHECKED_SIZE$)</td>
              </tr>
              <tr>
                <td class="subtitle">$LC571$:</td>
                <td align="right" class="subtitle">$RP_BLOCK_PERC$
($RP_BLOCK_SIZE$)</td>
              </tr>
              <tr>
                <td class="subtitle">$LC570$:</td>
                <td align="right" class="subtitle">$RP_CHECK_TIME_SUM$
(~$RP_CHECK_TIME_AVR$ $LC558$)</td>
              </tr>
            </tbody>
            </table>
            <if RP_NAME == "drweb" ><if RP_AGENT_STAT_UUID>
             <p align="right" class="regulartext"> $LC567$ <a href="http://
stat.drweb.com/view/$RP_AGENT_STAT_UUID$">$LC568$</a>. </p>
            </if></if>
            <p> <a href="#top_$LANG$" class="ancherlinks">$LC561$</a> </p>
          </for>
          </td>
      </tr>
    </table>
  </div>
  <br />
</body>
</html>
</for>
</if><if TYPE==PLAIN>
<for LANG;LANGS>
--001-DrWeb-MailFilter-Notification
Content-Type: text/plain; charset=$CHARSET$
Content-Transfer-Encoding: $CONTENT_TRANSFER_ENCODING$

              $LC542$ $R_PERIOD$

<for RP_NAME;R_PLUGINS==".*";"">
    *** $LC543$ $RP_NAME$ ***
<if RP_BLOCKED_OBJECTS_WITH_NUM_AND_PERCENTS>
<if RP_BLOCKED_OBJECTS_NUM!="0">
<if RP_BLOCKED_OBJECTS_NUM=="-1">
$LC544$:
```

```
</if><if RP_BLOCKED_OBJECTS_NUM!="-1">
$LC545$ $RP_BLOCKED_OBJECTS_NUM$ $LC546$:
</if>
$RP_BLOCKED_OBJECTS_WITH_NUM_AND_PERCENTS$
</if>
</if>
<if RP_SENDERS_ENVELOPE_WITH_NUM_AND_PERCENTS>
<if RP_SENDERS_ENVELOPE_NUM!="0">
<if RP_SENDERS_ENVELOPE_NUM=="-1">
$LC547$:
</if><if RP_SENDERS_ENVELOPE_NUM!="-1">
$LC545$ $RP_SENDERS_ENVELOPE_NUM$ $LC548$:
</if>
$RP_SENDERS_ENVELOPE_WITH_NUM_AND_PERCENTS$
</if>
</if>
<if RP_CLIENT_IP_WITH_NUM_AND_PERCENTS>
<if RP_CLIENT_IP_NUM!="0">
<if RP_CLIENT_IP_NUM=="-1">
$LC566$:
</if><if RP_CLIENT_IP_NUM!="-1">
$LC545$ $RP_CLIENT_IP_NUM$ $LC565$:
</if>
$RP_CLIENT_IP_WITH_NUM_AND_PERCENTS$
</if>
</if>
  $LC549$:
 $LC550$:       $RP_PASS$ ($RP_PASS_SIZE$)
 $LC551$:       $RP_REJECT$ ($RP_REJECT_SIZE$)
 $LC552$:       $RP_DISCARD$ ($RP_DISCARD_SIZE$)
 $LC553$:       $RP_TEMPFAIL$ ($RP_TEMPFAIL_SIZE$)
 $LC554$:       $RP_QUARANTINE$ ($RP_QUARANTINE_SIZE$)
 $LC555$:       $RP_REDIRECT$ ($RP_REDIRECT_SIZE$)
 $LC556$:       $RP_NOTIFY$ ($RP_NOTIFY_SIZE$)

------------------------------
 $LC557$:       $RP_CHECKED_MSGS$ ($RP_CHECKED_SIZE$)
 $LC571$:       $RP_BLOCK_PERC$ ($RP_BLOCK_SIZE$)
 $LC570$:       $RP_CHECK_TIME_SUM$ (~$RP_CHECK_TIME_AVR$ $LC558$)
<if RP_NAME == "drweb" ><if RP_AGENT_STAT_UUID>

    $LC567$ $LC568$:
       http://stat.drweb.com/view/$RP_AGENT_STAT_UUID$
</if></if>
</for>
</for>
</if>
--001-DrWeb-MailFilter-Notification--
```

Языковые файлы

Языковые файлы используются как источники текстовых данных (строковых ресурсов) при формировании уведомлений, выводе текстовых сообщений или вставке текстовых фрагментов в обрабатываемые письма. Языковые файлы используются в основном компонентом **Notifier**, но также могут использоваться подключаемыми модулями (в частности, модуль **Drweb** использует текст из этого файла для вставки сообщения в письмо вместо удаленного из него вредоносного вложения). Путь к каталогу, в котором хранятся используемые языковые файлы, должен быть одним и тем же как для компонента **Notifier**, так и для подключаемых модулей. Путь к используемому каталогу с языковыми файлами указывается в параметре **LngBaseDir** в секции [Notifier] основного конфигурационного файла.

Языковые файлы имеют имя, образованное по принципу `[<plug-in>_]<language>.lng`, где:

- `<plug-in>` – имя подключаемого модуля, использующего этот языковой файл в качестве источника строк (`drweb`, `modifier` и т.п.)
- `<language>` – название языка, на котором составлены текстовые строки, хранящиеся в файле.

Языковые файлы, используемые непосредственно компонентом **Notifier**, не имеют в своем названии приставки `<plug-in>_`.

Языковой файл имеет следующую внутреннюю структуру:

- В первой строке указывается сокращенное название языка, на котором составлены все содержащиеся в файле тексты (`en`, `ru` и т.п.).
- Во второй строке – название используемой кодировки (например, `koi8-r`).
- В третьей строке – указание на используемую битность кодировки CTE (`7bit` или `8bit`).
- Оставшаяся часть файла состоит из строк вида `N="текст"`, где `N` – номер (идентификатор) текста, а `текст` – используемый текст.
- Также в языковом файле могут присутствовать строки-комментарии, начинающиеся с символа '#' или пустые строки. Такие строки игнорируются.

В параметре **NotifyLangs** в секции `[Notifier]` используются только обозначения языка, которые берутся из первой строки языковых файлов.

Пример языкового файла:

```
#language name = LANG
en
#coding system = CHARSET
UTF8
#Content-Transfer-Encoding: 7bit/8bit
8bit

1 = "OK"
2 = "password protected, skipped"
...
```

Все подключаемые модули используют в своей работе только тот языковой файл, который указан первым в списке **NotifyLangs**. При этом для поиска необходимой строки (в случае обработки вызова макроса вставки строки `$n`, например, при действии `add-header`) всегда выполняется следующий алгоритм:

1) Определяется каталог, содержащий языковые файлы (используется значение параметра **LngBaseDir**);
2) Определяется первый используемый язык из списка, указанного в значении параметра **NotifyLangs**;
3) Ищется файл, в имени которого присутствует префикс с именем подключаемого модуля, а первая строчка соответствует сокращенному названию используемого языка;
4) Если такой файл найден, то в нем ищется строка с требуемым номером `n` в левой части. При этом предполагается, что содержимое этой строки закодировано с использованием кодировки и CTE, указанных в заголовке этого языкового файла.

Найденная строка будет использована при обработке письма (добавлена как текст в письмо или в заголовок, в зависимости от совершаемого модулем действия), при этом будут использованы кодировка и CTE, указанные в заголовке языкового файла, из которого была извлечена строка.

Если требуемый файл не будет найден или в нем не будет найдена строка с требуемым номером, эта ситуация будет являться ошибочной и будет обрабатываться в соответствии со значением параметра **ProcessingErrors** в секции [Maild] основного конфигурационного файла.

При необходимости вы можете добавить в языковые файлы свои строки. При этом нужно следить, чтобы добавленные строки:

- не использовали уже занятые в этом файле номера, поскольку строки с этими номерами уже используются какими-то модулями **Dr.Web MailD** (или подключаемым модулем).
- использовали кодировку и битность СТЕ, указанные в заголовке языкового файла.

Подключаемые модули

На текущий момент реализованы следующие подключаемые модули компонента **Dr.Web MailD**:

- Антивирусный модуль **Drweb**, осуществляющий проверку содержимого сообщений на вирусы и вредоносное ПО;
- Модуль антиспам-проверки **Vaderetro**, проверяющий письма на наличие признаков спама;
- Модуль **Dr.Web HeadersFilter**, осуществляющий фильтрацию писем по значениям различных заголовков;
- Модуль **Dr.Web Modifier**, позволяющий произвольно изменять части писем.

Каждый подключаемый модуль представляет собой динамически подгружаемую библиотеку (файл с расширением .so). Библиотеки располагаются по умолчанию в каталоге %bin_dir/maild/plugins. Файл библиотеки каждого подключаемого модуля по умолчанию имеет имя вида **lib<name>.so**, где <name> – имя модуля. Например, файл библиотеки подключаемого модуля **Drweb** имеет имя **libdrweb.so**.

Каждый подключаемый модуль использует собственный конфигурационный файл. Конфигурационные файлы подключаемых модулей по умолчанию располагаются в каталоге %etc_dir. Конфигурационный файл конкретного модуля имеет по умолчанию имя plugin_<name>.conf, где <name> – имя модуля. Например, конфигурационный файл подключаемого модуля **Drweb** по умолчанию имеет имя plugin_drweb.conf.

При необходимости в секции [Filters] основного конфигурационного файла **Dr.Web MailD** можно настроить каждый подключаемый модуль на использование конфигурационных файлов и динамических библиотек, название которых не соответствует схеме, принятой по умолчанию.

Обратите внимание, что при запуске **Dr.Web MailD** временно переименовывает библиотеки используемых подключаемых модулей, добавляя к их имени дополнительное расширение .cache для избежания возможных конфликтов при обновлении модулей через компонент обновления **Dr.Web Updater**.

Например, библиотека подключаемого модуля **Drweb** при запуске получает имя **libdrweb.so.cache**.

Антивирусный модуль Drweb

Drweb – подключаемый модуль компонента **Dr.Web MailD**, осуществляющий антивирусную проверку почтовых сообщений.

Для работы этого модуля необходимы **Dr.Web Daemon** и **Антивирусное ядро Dr.Web**, которые осуществляют непосредственную антивирусную проверку сообщений. Компонент **Dr.Web Daemon** и **Антивирусное ядро Dr.Web** входят в базовый пакет программного комплекса **Dr.Web для почтовых серверов UNIX**, который должен быть установлен до установки подключаемого модуля **Drweb**.

Сообщения передаются сканирующему демону **Dr.Web Daemon** (**drwebd**) на проверку уже разобранными на части, поэтому поддержка MIME-разбора в **Антивирусном ядре** и модуле **Dr.Web Daemon** не требуется. Закончив анализ письма, подключаемый модуль **Drweb** передает модулю **Maild core** результаты проверки и (при значении Yes параметра **AddXHeaders** конфигурационного файла подключаемого модуля) может добавить в него следующие заголовки:

- X-Antivirus: Name, где Name – название и версия антивируса;
- X-Antivirus-Code, где Code – код завершения работы модуля **Dr.Web Daemon**.

Управление параметрами работы подключаемого модуля **Drweb** осуществляется (по умолчанию) в конфигурационном файле plugin_drweb.conf.

Подключение модуля

Чтобы подключить антивирусный модуль **Drweb** к программному комплексу **Dr.Web для почтовых серверов UNIX** достаточно в конфигурационном файле **Dr.Web MailD** добавить строку drweb в список подключаемых модулей, обрабатывающих письма.

Если письма должны обрабатываться антивирусным модулем **Drweb** до помещения в базу данных, его следует добавлять в список значений параметра **BeforeQueueFilters** секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
BeforeQueueFilters = drweb, vaderetro
```

Если же письма должны попадать в антивирусный модуль уже после помещения в базу данных, он добавляется в список значение параметра **AfterQueueFilters** секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
AfterQueueFilters = drweb
```

Настройка модуля

Все основные параметры работы модуля задаются (по умолчанию) в конфигурационном файле %etc_dir/plugin_drweb.conf. Устройство конфигурационного файла и краткое описание его параметров приведены в в разделе Конфигурационные файлы.

В секции [Antivirus] собраны общие настройки работы антивирусного модуля **Drweb**:

Секция [Antivirus]

Address = {адрес}	Сокет, через который антивирусный модуль взаимодействует с **Dr.Web Daemon**.

	Допускается указание нескольких сокетов для взаимодействия с экземплярами **Dr.Web Daemon**, находящимися на разных серверах, при этом взаимодействие осуществляется с использованием функции балансировки нагрузки на каждый из используемых серверов.
	Адреса сокетов задаются в виде:
	`ADDRESS1 [WEIGHT1], ADDRESS2 [WEIGHT2] ...`
	где `ADDRESS` указан в стандартном формате, а `WEIGHT` представляет собой необязательный вес этого адреса. `WEIGHT` определяет относительную нагрузку на данный узел сети и может принимать значения от 0 до 100 включительно.
	Среди указанных адресов должен присутствовать хотя бы один корректный адрес сервера.
	Кроме адресов стандартного формата, можно указывать путь к PID файлу **Dr.Web Daemon**, из которого впоследствии будет извлечена нужная информация о сокетах.
	Примеры:
	Задание адреса PID:
	`Address = pid:%var_dir/run/drwebd.pid`
	Задание нескольких адресов:
	`Address = pid:%var_dir/run/drwebd.pid 10, inet:3000@srv2.example.com 5`
	Значение по умолчанию:
	`Address = pid:%var_dir/run/drwebd.pid`

`Timeout = {время}`	Максимальное время ожидания исполнения команды **Dr.Web Daemon**. Если значение параметра равно `0`, время ожидания не ограничено. Значение по умолчанию: `Timeout = 30s`
`ScanType = {local | remote | auto}`	Тип взаимодействия с **Dr.Web Daemon** для сканирования содержимого писем. • `local` – Локальный режим. Передаются только имена файлов, но не их содержимое; • `remote` – Удаленный режим. Передается содержимое файлов; • `auto` – Автоматический режим. Передаются либо имена, либо содержимое файлов. Определение реально используемого режима будет зависеть от размера файла письма, от того, локально или удаленно расположен используемый для сканирования **Dr.Web Daemon**, а также, в каком из режимов (синхронном или асинхронном) модуль проверяет это письмо. Крайне рекомендуется использовать режим `auto`, заданный по умолчанию. Режим сканирования `local` может быть использован только в том случае, если используемый для сканирования **Dr.Web Daemon** расположен на локальной машине (определяется типом адреса, заданного в параметре **`Address`**). Если в списке адресов имеется хотя бы один удаленный **Dr.Web Daemon**, лучше не использовать режим `local`. **Важно!** В случае если параметр **`ScanType`** имеет значение

	`local` или `auto`, включение в настройках анализатора **Dr.Web Daemon** режима **`ScanFiles`** = `ByType` приведет к тому, что он будет пропускать письма без антивирусной проверки! Значение по умолчанию: **`ScanType`** = `auto`
`HeuristicAnalysis` = {логический}	Использование *Эвристического анализа* позволяет **Dr.Web Daemon** обнаруживать неизвестные вирусы. При отключении *Эвристического анализа* будут обнаружены только уже известные вирусы, информация о которых хранится в антивирусных базах. При включении анализатора **Dr.Web Daemon** может посылать ложные сообщения об обнаружении вирусов, поскольку работа полезных программ иногда бывает похожа на вирусную активность. Использование *Эвристического анализа* может привести к увеличению времени сканирования. Значение по умолчанию: **`HeuristicAnalysis`** = `Yes`
`TCP_NODELAY` = {логический}	При значении `Yes` сокет будет создан с параметром **`TCP_NODELAY`**. Если вы не испытываете проблем с сетью, не изменяйте заданное по умолчанию значение `No`. Значение по умолчанию: **`TCP_NODELAY`** = `No`
`ReportMaxSize` = {размер}	Максимальный размер файла отчета **Dr.Web Daemon**. Когда значение параметра **`ReportMaxSize`** равно `0`, размер файла отчета не ограничен. Не рекомендуется устанавливать значение равным `0`, так как в противном случае размер файла отчета может превысить несколько мегабайт после обнаружения в сообщениях вредоносных программ или "почтовых бомб". Значение по умолчанию: **`ReportMaxSize`** = `50k`
`AddXHeaders` = {логический}	Если указано значение `Yes`, к каждому проверенному **Dr.Web Daemon** сообщению добавляются заголовки `X-Antivirus` и `X-Antivirus-Code`. Значение по умолчанию: **`AddXHeaders`** = `Yes`
`Paranoid` = {логический}	Если у параметра указано значение `Yes`, все сообщения будут сканироваться в "параноидальном" режиме. В этом случае **Dr.Web Daemon** будет обрабатывать каждое сообщение дважды: целиком и по частям. Такой подход позволяет повысить надежность обнаружения вирусов, но одновременно приводит к увеличению времени сканирования. Обратите внимание, что если в письме находится объект, для которого выполняется действие `pass`, то возможно удвоение статистической информации по этому объекту (если вирус найден и при отправке вложения, и при отправке всего письма), а также могут по два раза выполняться

	дополнительные действия (notify, redirect). Значение по умолчанию: **Paranoid** = No
R A **RegexsForCheckedFilename** = {список регулярных выражений}	Список регулярных выражений, используемых подключаемым модулем при проверке имен файлов в отчете, присылаемом **Dr.Web Daemon** после сканирования сообщения. Имена файлов в архивах будут начинаться с символа ">" (количество символов ">" перед именем файла будет зависеть от степени вложенности архива). При совпадении части имени файла с каким-либо из элементов списка, выполняется действие, заданное в настройках параметра **BlockByFilename**. Данная проверка будет производиться только для файлов, в которых не найдено вирусов. Значение по умолчанию: **RegexsForCheckedFilename** =
R **LicenseLimit** = {список действий}	Действия, применяемые к сообщениям, которые не были проверены **Dr.Web Daemon** по причине окончания срока действия лицензии. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: pass, tempfail, discard, reject. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **LicenseLimit** = pass
R **Infected** = {список действий}	Действия, совершаемые с сообщениями, зараженными известными вирусами. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: cure, remove, discard, reject. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify. Значение по умолчанию: **Infected** = cure, quarantine
R **Suspicious** = {список действий}	Действия, совершаемые с сообщениями, которые могут быть заражены неизвестным вирусом (если сработал *Эвристический анализ*). Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: pass, remove, discard, reject. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score.

	Значение по умолчанию: **Suspicious** = reject, quarantine, notify
Incurable = {список действий} R	Действия, совершаемые с сообщениями, зараженными неизлечимо. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: remove, discard, reject. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Incurable** = reject, quarantine, notify
Adware = {список действий} R	Действия, совершаемые с сообщениями, которые содержат программы для показа рекламы. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: pass, remove, discard, reject. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Adware** = reject, quarantine, notify
Dialers = {список действий} R	Действия, совершаемые с сообщениями, содержащими программы дозвона. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: pass, remove, discard, reject. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Dialers** = reject, quarantine, notify
Jokes = {список действий} R	Действия, совершаемые с сообщениями, содержащими программы-шутки. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: pass, remove, discard, reject. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Jokes** = reject, quarantine, notify
Riskware = {список действий} R	Действия, совершаемые с сообщениями, содержащими потенциально опасные программы. Должно быть задано одно основное действие (обязательно), и,

	возможно, несколько дополнительных. К основным действиям относятся: `pass`, `remove`, `discard`, `reject`. Дополнительно могут быть заданы следующие действия: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **`Riskware`** = `reject, quarantine, notify`
`Hacktools` = `{список действий}` R	Действия, совершаемые с сообщениями, содержащими программы, предназначенные для получения несанкционированного доступа к компьютерным системам. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: `pass`, `remove`, `discard`, `reject`. Дополнительно могут быть заданы следующие действия: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **`Hacktools`** = `reject, quarantine, notify`
`SkipObject` = `{список действий}` R	Действия, совершаемые с сообщениями, содержащими объекты, которые не могут быть проверены **Dr.Web Daemon**. Возможны следующие причины: • Во вложении находятся защищённые паролем или испорченные архивы, символические ссылки, файлы нестандартных форматов или зашифрованные файлы. • Достигнуто максимальное время ожидания проверки сообщения. (Для получения более подробной информации обратитесь к описанию параметров **`SocketTimeout`** и **`FileTimeout`** в главном конфигурационном файле `drweb32.ini`). Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: `pass`, `remove`, `discard`, `reject`. Дополнительно могут быть заданы следующие действия: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **`SkipObject`** = `pass`
`ArchiveRestriction` = `{список действий}` R	Действия, совершаемые с сообщениями, содержащими архивы, которые не могут быть проверены **Dr.Web Daemon** вследствие нарушения любого из следующих ограничений: • Степень сжатия архива превышает значение параметра **`MaxCompressionRatio`**. • Размер запакованного объекта превышает значение параметра **`MaxFileSizeToExtract`**. • Степень вложенности архива превышает значение параметра **`MaxArchiveLevel`**. Все эти ограничения определяются в настройках модуля **Dr.Web Daemon**. Должно быть задано одно основное действие (обязательно), и,

	возможно, несколько дополнительных. К основным действиям относятся: `pass`, `remove`, `discard`, `reject`. Дополнительно могут быть заданы следующие действия: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **`ArchiveRestriction`** = `reject, quarantine, notify`
`ScanningErrors` = {список действий}	R Действия, совершаемые с сообщениями, вызывающими у **Dr.Web Daemon** ошибки в процессе проверки. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: `pass`, `remove`, `discard`, `reject`, `tempfail`. Дополнительно могут быть заданы следующие действия: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **`ScanningErrors`** = `reject, quarantine`
`ProcessingErrors` = {список действий}	R Действия, совершаемые с сообщениями, вызывающими у сканирующего демона **Dr.Web Daemon** ошибки в процессе проверки. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: `pass`, `discard`, `reject`, `tempfail`. Дополнительно могут быть заданы следующие действия: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **`ProcessingErrors`** = `reject`
`BlockByFilename` = {список действий}	R Действия, выполняющиеся в случае совпадения одного из регулярных выражений, указанных в настройках параметра **`RegexsForCheckedFilename`**, с именем файла из отчета, присылаемого **Dr.Web Daemon** после сканирования сообщения. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: `pass`, `discard`, `reject`, `tempfail`. Дополнительно могут быть заданы следующие действия: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Обратите внимание, что в случаях, когда связь с **Dr.Web Daemon** осуществляется через TCP-сокет, в отчетах используется иной формат имени файлов. **Пример**: `127.0.0.1 [17078] >/var/drweb/msgs/db/6/00007976/.msg/1.part - Ok` Т.е. имя файла будет начинаться не с символа ">", а с IP-адреса и номера сканирующего процесса. Эту разницу необходимо учитывать при задании регулярных выражений в

	значении параметра **RegexsForCheckedFilename**. Значение по умолчанию: **BlockByFilename** = reject, quarantine, notify

В тех случаях, когда сообщение блокируется (reject) модулем в синхронном режиме обработки, ответ **Dr.Web MailD** клиенту состоит из кода SMTP (55* или 250, в зависимости от значения параметра **ReturnReject** в секции [Receiver]), а также текстового сообщения, содержание которого может задаваться идущими далее параметрами. Значения параметров должны быть заключены в кавычки.

UseCustomReply = {логический} R	Использование настраиваемых сообщений в SMTP-сессии для случаев, когда сообщения отклоняются. Значение по умолчанию: **UseCustomReply** = No
ReplyInfected = {текст} R	Настраиваемое сообщение в SMTP-сессии для случаев, когда выполняется действие **Infected** = reject или **Incurable** = reject, и если **UseCustomReply** = yes. Вы можете задать только текстовую часть сообщения. Текст, содержащий пробелы, должен быть заключен в кавычки. **Пример**: 550 5.7.0 "Text part of reply" Значение по умолчанию: **ReplyInfected** = "DrWEB Antivirus: Message is rejected because it contains a virus."
ReplyMalware = {текст} R	Настраиваемое сообщение в SMTP-сессии для случаев, когда выполняется какие-либо из действий **Adware**, **Dialers**, **Jokes**, **Riskware**, **Hacktools** = reject, и если **UseCustomReply** = Yes. Вы можете задать только текстовую часть сообщения. Текст, содержащий пробелы, должен быть заключен в кавычки. **Пример**: 550 5.7.0 "Text part of reply" Значение по умолчанию: **ReplyMalware** = "DrWEB Antivirus: Message is rejected because it contains a malware."
ReplySuspicious = {текст} R	Настраиваемое сообщение в SMTP-сессии для случаев, когда выполняется действие **Suspicious** = reject, и если **UseCustomReply** = Yes. Вы можете задать только текстовую часть сообщения. Текст, содержащий пробелы, должен быть заключен в кавычки. **Пример**: 550 5.7.0 "Text part of reply" Значение по умолчанию: **ReplySuspicious** = "DrWEB Antivirus: Message is rejected because it contains suspicious content."
ReplySkipObject = {текст} R	Настраиваемое сообщение в SMTP-сессии для случаев, когда выполняется действие **SkipObject** = reject, и если **UseCustomReply** = Yes. Вы можете задать только текстовую часть сообщения. Текст,

	содержащий пробелы, должен быть заключен в кавычки. **Пример**: `550 5.7.0 "Text part of reply"` Значение по умолчанию: **ReplySkipObject** = "DrWEB Antivirus: Message is rejected because it cannot be checked."
ReplyArchiveRestriction = {текст}	R Настраиваемое сообщение в SMTP-сессии для случаев, когда выполняется действие **ArchiveRestriction** = reject, а также если **UseCustomReply** = Yes. Вы можете задать только текстовую часть сообщения. Текст, содержащий пробелы, должен быть заключен в кавычки. **Пример**: `550 5.7.0 "Text part of reply"` Значение по умолчанию: **ReplyArchiveRestriction** = "DrWEB Antivirus: Message is rejected because it contains archive which violates restrictions."
ReplyError = {текст}	R Настраиваемое сообщение в SMTP-сессии для случаев, когда выполняется какое-либо из действий **ScanningErrors**, **ProcessingErrors** = reject, и если **UseCustomReply** = Yes. Вы можете задать только текстовую часть сообщения. Текст, содержащий пробелы, должен быть заключен в кавычки. **Пример**: `550 5.7.0 "Text part of reply"` Значение по умолчанию: **ReplyError** = "DrWEB Antivirus: Message is rejected due to software error."
ReplyBlockByFilename = {текст}	R Настраиваемое сообщение в SMTP-сессии для случаев, когда выполняется действие **BlockByFilename** = reject, а также если **UseCustomReply** = Yes. Вы можете задать только текстовую часть сообщения. Текст, содержащий пробелы, должен быть заключен в кавычки. **Пример**: `550 5.7.0 "Text part of reply"` Значение по умолчанию: **ReplyBlockByFilename** = "DrWEB MailD: Message is rejected due to filename pattern"

Если **UseCustomReply** = No или соответствующая строка текста не задана, будет выдано стандартное сообщение "The message has been rejected by the Dr.Web MailD".

Примеры

Примеры настройки действия антивирусного подключаемого модуля **Drweb**:

1. **Пример настройки действия `Suspicious`.** Если письмо заражено неизвестным объектом, то отклонить его, поместить в **Карантин** и добавить в письмо заголовок `X-DRWEB-PLUGIN-CHECK-STATUS` со значением, извлеченным из строки с идентификатором 120 из языкового файла по умолчанию:

```
Suspicious = reject, quarantine, notify, add-header (X-DRWEB-PLUGIN-CHECK-STATUS:$120)
```

2. **Пример настройки действия `ArchiveRestriction`.** Если архив, содержащийся в письме, нарушил при проверке ограничения на проверку архивов, то пропустить письмо к получателю (точнее, разрешить продолжить его дальнейшую проверку другими подключаемыми модулями, если они еще остались в очереди), сформировать уведомление об результатах проверки для администратора, и увеличить счет письма на 100 баллов:

```
ArchiveRestriction = pass, notify, score (100)
```

Модуль Dr.Web HeadersFilter

Dr.Web HeadersFilter – подключаемый модуль компонента **Dr.Web MailD**, осуществляющий фильтрацию писем по их заголовкам. При задании правил фильтрации можно использовать регулярные выражения (синтаксис регулярных выражений **Perl**).

Подключение модуля

Чтобы подключить модуль фильтрации **Dr.Web HeadersFilter** к программному комплексу **Dr.Web для почтовых серверов UNIX**, достаточно в конфигурационном файле **Dr.Web MailD** добавить строку `headersfilter` в список подключаемых модулей, обрабатывающих письмо.

Если письмо должно обрабатываться модулем **Dr.Web HeadersFilter** до помещения в базу данных, то его название следует добавлять в список значений параметра **`BeforeQueueFilters`** секции `[Filter]` конфигурационного файла **Dr.Web MailD**.

Пример:

```
BeforeQueueFilters = drweb, headersfilter
```

Если же письмо должно обрабатываться модулем уже после помещения в базу данных, то его название добавляется в список значений параметра **`AfterQueueFilters`** секции `[Filter]` конфигурационного файла **Dr.Web MailD**.

Пример:

```
AfterQueueFilters = headersfilter
```

Настройка модуля

Все основные параметры работы подключаемого модуля задаются (по умолчанию) в файле `%etc_dir/plugin_headersfilter.conf`. Устройство конфигурационного файла и краткое описание его параметров приведены в в разделе Конфигурационные файлы.

В секции `[Headersfilter]` собраны общие настройки работы модуля фильтрации по заголовкам **Dr.Web HeadersFilter**.

Параметры фильтрации писем задаются с помощью **правил фильтрации**, описанных ниже. Проверка правил фильтрации осуществляется в порядке их задания, т.е. правило, заданное первым, будет проверено первым. Поиск соответствия правилу фильтрации осуществляется до первого подходящего правила, после чего возвращается действие, заданное для этого правила фильтрации.

Если почтовое сообщение попало под действие правила фильтрации типа **`Reject*`**, то дальнейшая проверка сообщения не производится. Если почтовое сообщение попало под действие правила фильтрации типа **`Accept*`**, то оставшиеся правила фильтрации игнорируются и продолжается обработка сообщения остальными подключаемыми модулями **Dr.Web MailD** (если в очередях еще остались модули, не обработавшие письмо).

Секция `[Headersfilter]`

`ScanEncodedHeaders` = {логический}	R	Сканирование заголовков сообщений перед их перекодированием. К примеру, указание значения **Yes** для параметра **`ScanEncodedHeaders`** вместе с условием **`RejectCondition`** = `Subject = "iso-8859-5"` позволяет отфильтровать сообщения, поле `Subject` которых

	закодировано в `iso-8859-5`. Пожалуйста, обратите внимание, что все при установке этого параметра в Yes закодированные заголовки будут просканированы всеми правилами фильтрации дважды: до и после перекодирования (при этом сканирование останавливается, если срабатывает какое-либо правило). Значение по умолчанию: **ScanEncodedHeaders** = Yes
RejectCondition = {набор условий} R A	Отвергающие правила фильтрации сообщений. Если какой-либо из заголовков письма попадает под действие того или иного из указанных правил, письмо отфильтровывается. Действия, применяемые к такому почтовому сообщению, задаются параметром **Action** в той же секции конфигурационного файла. Отвергающие правила фильтрации могут быть заданы для любого из заголовков. Обычно каждое правило фильтрации состоит из имени заголовка и регулярного выражения. `HEADER_NAME = regular_expression` Допустимо объединять несколько правил с помощью скобок или логических операторов `OR` и `AND`. Оператор `"!="` (не равно) также может быть использован. Выражения, содержащие пробелы, обязательно должны быть заключены в кавычки. **Пример**: **RejectCondition** = Subject = "money" AND Content-Type = "text/html" Также существует два дополнительных типа фильтрации: • `No HEADER_NAME` - условие, позволяющее отфильтровать сообщения, у которых какой-либо из заголовков отсутствует. **Пример**: **RejectCondition** = No From - отфильтровывает все письма с отсутствующим заголовком `From`. • `HEADER_NAME = "8bit"` - условие, позволяющее отфильтровать сообщения, у которых в заголовках содержатся 8-битные символы. См. важную сноску под таблицей. Значение по умолчанию: **RejectCondition =**
AcceptCondition = {набор условий} R A	Принимающие правила фильтрации сообщений. Если какой-либо из заголовков письма попадает под действие того или иного из указанных принимающих правил, сканирование его заголовков прекращается, и письмо передается для дальнейшей обработки другим модулями. Принимающие правила фильтрации могут быть заданы для любого из заголовков. Все, сказанное про набор условий **RejectCondition**, справедливо и для условий **AcceptCondition**. Значение по умолчанию: **AcceptCondition =**

FilterParts = {логический}	R Значение Yes разрешает обработку правил фильтрации, заданных параметрами **RejectPartCondition** и **AcceptPartCondition**. Значение по умолчанию: **FilterParts** = Yes
RejectPartCondition = {набор условий} **AcceptPartCondition** = {набор условий}	R A Правила фильтрации, аналогичные **RejectCondition** и **AcceptCondition**, но работающие с заголовками вложенных объектов. Также может быть использовано следующее правило: **FileName** = mask где "mask" — регулярное выражение. Обработка сообщений в соответствии с этими правилами фильтрации возможна только если параметр **FilterParts** имеет значение Yes. Значение по умолчанию: **RejectPartCondition** = **AcceptPartCondition** =
MissingHeader = {текст}	R A Набор заголовков, отсутствие которых в письме становится условием его отфильтровывания. **Пример**: **MissingHeader** = "To", "From" Значение по умолчанию: **MissingHeader** =
Action = {список действий}	R Действия, совершаемые с отфильтрованными сообщениями. Должно быть задано одно основное действие (обязательно), и, возможно, несколько дополнительных. К основным действиям относятся: pass, tempfail, discard, reject. Дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Action** = reject, notify

В тех случаях, когда сообщение блокируется (reject) модулем в синхронном режиме обработки, ответ **Dr.Web MailD** клиенту состоит из кода SMTP (55* или 250, в зависимости от значения параметра **ReturnReject** в секции [Receiver]), а также текстового сообщения, содержание которого может задаваться идущими далее параметрами. Значения параметров должны быть заключены в кавычки.

UseCustomReply = {логический}	R Использование настраиваемых сообщений в SMTP-сессии для случаев, когда сообщения отклоняются. Значение по умолчанию: **UseCustomReply** = No
ReplyRuleFilter = {текст}	R Настраиваемое сообщение в SMTP-сессии для случаев, когда выполняется действие **Action** = reject, а также если **UseCustomReply** = Yes. Вы можете задать только текстовую часть сообщения. Текст, содержащий пробелы, должен быть заключен в кавычки.

	Пример: `550 5.7.0 "Text part of reply"` Значение по умолчанию: **ReplyRuleFilter** = "DrWEB HeadersFilter plugin: Message is rejected by headers rule filter."

Если **UseCustomReply** = No или соответствующая строка текста не задана, будет выдано стандартное сообщение "The message has been rejected by the Dr.Web MailD".

Пожалуйста, обратите внимание, что **во всех правилах фильтрации**, меняющих или ищущих значение некоторого заголовка или части письма, следует **учитывать кодировку**, в которой встречается искомый текст.

Правила работы со значениями заголовков в произвольных кодировках рассмотрены в разделе Работа со строковыми значениями

Примеры

Примеры настройки действия модуля фильтрации **Dr.Web HeadersFilter**:

1. Требуется отфильтровывать письма, не содержащие заголовка To, или письма, тема которых состоит из слова «тест» в кодировке CP1251. Отфильтрованные письма следует отклонять, помещать в **Карантин** и уведомлять об этом администратора:

```
MissingHeader = "To"
RejectCondition = Subject = "=\\?cp1251\\?B\\?8uXx8go=\\?="
Action = reject, quarantine, notify
```

2. Если письмо не содержит темы, пропустить его, но добавить тему «(none)» и увеличить счет письма на 500 баллов:

```
MissingHeader = "Subject"
Action = pass, score (500), add-header ("Subject: (none)")
```

Модуль антиспам-проверки Vaderetro

Vaderetro – подключаемый модуль компонента **Dr.Web MailD**, осуществляющий спам-фильтрацию почтовых сообщений с помощью библиотеки **VadeRetro**, разработанной компанией **Vade Retro Technology** (подразделение компании **GoTo Software**).

Библиотека **VadeRetro** проводит анализ почтовой корреспонденции автономно, без обращения к внешним источникам информации о спаме. Кроме того, библиотека **VadeRetro** обеспечивает высокую скорость обработки писем и постоянное улучшение качества анализа сообщений благодаря динамическому обновлению кода библиотеки (обновление производится посредством компонента обновления **Dr.Web Updater**).

Файл динамически подгружаемой библиотеки **VadeRetro**, используемой антиспам-модулем **Vaderetro**, находится в каталоге `%var_dir/lib` и называется **libvaderetro.so** (так же, как и файл динамической библиотеки этого подключаемого модуля, расположенной по умолчанию в каталоге `%bin_dir/maild/plugins`). Обратите внимание, что не смотря на то, что имя файла библиотеки **VadeRetro** и имя файла библиотеки подключаемого модуля совпадают, это разные библиотеки!

Обратите внимание, что при запуске **Dr.Web MailD** временно переименовывает файл библиотеки **VadeRetro**, добавляя к его имени дополнительное расширение `.cache` для избежания возможных конфликтов при обновлении библиотеки через компонент обновления **Dr.Web Updater**.

В зависимости от результатов анализа, каждому письму, проверенному библиотекой **VadeRetro**, дается оценка (*score*) – целое число в диапазоне от `-10000` до `+10000`. Чем больше эта величина, тем больше вероятность, что письмо является спамом. Пороговое значение оценки для отнесения письма к спаму самим антиспам-модулем **Vaderetro** задается в параметре **SpamThreshold** конфигурационного файла подключаемого модуля (если оценка, присвоенная сообщению, больше или равна значению параметра **SpamThreshold**, то письмо классифицируется как спам).

Закончив анализ письма, библиотека **VadeRetro** может добавить в него следующие заголовки:

- **X-Drweb-SpamScore**: `n`, где `n` – оценка, данная письму библиотекой **VadeRetro**. Этот заголовок добавляется только в том случае, если для параметра **AddXHeaders** задано значение `Yes`.
- **X-Drweb-SpamState**: `b`, где `b` – `Yes` для спама и писем с вирусами и `No` для "не-спама" и уведомлений DSN. Этот заголовок добавляется только в том случае, если для параметра **AddXHeaders** задано значение `Yes`.
- **X-Drweb-SpamState-Num**: `s`, где `s` – результаты классификации письма библиотекой **VadeRetro**. `s` может принимать четыре значения: `0`, `1`, `2` и `3`:
  - `s = 0` – письмо не является спамом;
  - `s = 1` – письмо является спамом;
  - `s = 2` – письмо содержит вирус;
  - `s = 3` – сообщение является уведомлением DSN.

  Этот заголовок добавляется только в том случае, если для параметра **AddXDrwebSpamStateNumHeader** задано значение `Yes`.
- **X-Drweb-SpamVersion**: `version`, где `version` – версия библиотеки **VadeRetro**. Этот заголовок добавляется только в том случае, если для параметра **AddVersionHeader** задано значение `Yes`.
- **X-Spam-Level**: `z`, где `z` – это набор `*`, каждая из которых соответствует 10 очкам, присвоенным письму. Добавляется только если для параметра **AddXSpamLevel** задано значение `Yes`.

- **X-DrWeb-SpamReason**: some_text, где some_text – зашифрованное диагностическое сообщение антиспам-модуля. Оно необходимо для улучшения качества распознавания спама. Этот заголовок добавляется только в том случае, если для параметра **AddXHeaders** установлено значение yes.

Кроме того, в начало поля Subject писем, классифицированных подключаемым модулем с использованием счетчика **SpamThreshold** как спам, модуль может добавлять значение параметра **SubjectPrefix** своего конфигурационного файла. Это происходит только в том случае, если для параметра **SubjectPrefix** установлено значение, отличное от пустой строки.

Аналогично для уведомлений может добавляться префикс заголовка из **NotifySubjectPrefix**, а для писем, помеченных как "спам" или "содержащее угрозу" с использованием счетчика **UnconditionalSpamThreshold**, может добавляться префикс заголовка из **UnconditionalSubjectPrefix**.

Также предусмотрена возможность изменения счета сообщения на основе информации об адресах отправителей и получателей сообщения:

1. Имеется возможность задать белые и черные списки адресов отправителей (параметры конфигурации модуля **WhiteList** и **BlackList** соответственно). При обнаружении одного из адресов отправителей в черном или белом списке его счет изменяется на 5000 баллов (увеличивается и уменьшается соответственно) для каждого адреса, найденного в списке. Более подробно см. в описании этих параметров.
2. Имеется возможность указать, на какое количество баллов следует изменить счет писем, следующих из защищаемых сетей (т.е. сетей, перечисленных в параметре **ProtectedNetworks** секции [Maild] основного конфигурационного файла **Dr.Web MailD**).
3. Также предусмотрено использование специального кэша reply_cache, хранящего информацию о письмах, следующих из защищаемых сетей (а именно – перечень адресов получателей), с тем, чтобы учитывать эту информацию при анализе писем, следующих в защищаемые сети, и являющихся ответами на эти письма. В случае если отправитель входящего письма уже находится в кэше, то к счет письма может быть изменен на указанное количество баллов.

Обратите внимание, что письмо проходит последовательно через все проверки, поэтому если для одного и того же адреса, указанного в письме, сработали несколько условий, то все получаемые изменения счета суммируются. Например, если отправитель письма оказался в черном списке и при этом он же присутствует в кэше ответов reply_cache, то к счету письма, присвоенному после анализа на наличие формальных признаков спама, будет прибавлен штраф за нахождение отправителя в черном списке, а потом – величина, указанная в параметре **ReplyToProtectedNetworkScoreAdd** конфигурации подключаемого модуля.

Письма с ложными срабатываниями спам-фильтра **VadeRetro** следует пересылать по адресу vrnonspam@drweb.com, а письма с пропущенным спамом – на адрес vrspam@drweb.com.

Подключение модуля

Чтобы подключить антискам-модуль **Vaderetro** к компоненту **Dr.Web MailD**, достаточно в конфигурационном файле **Dr.Web MailD** добавить строку vaderetro в список подключаемых модулей, обрабатывающих письма.

Если письма должны обрабатываться модулем **Vaderetro** до помещения в базу данных, то его название следует добавлять в список значений параметра **BeforeQueueFilters** секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
BeforeQueueFilters = drweb, vaderetro
```

Если же письма должны обрабатываться модулем уже после помещения в базу данных, то его название добавляется в список значений параметра **AfterQueueFilters** секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
AfterQueueFilters = vaderetro
```

Настройка модуля

Все основные параметры работы модуля задаются (по умолчанию) в файле %etc_dir/plugin_vaderetro.conf. Устройство конфигурационного файла и краткое описание его параметров приведены в разделе Конфигурационные файлы.

В секции [VadeRetro] собраны общие настройки работы антиспам-модуля **Vaderetro**:

Секция [VadeRetro]

PathToVadeRetro = {путь к файлу}	Путь к антиспам-библиотеке **VadeRetro**. Возможно ее динамическое обновление с помощью компонента обновления **Dr.Web Updater**. Скрипт скачает новую версию библиотеки, запишет ее на место предыдущей и пошлет сигнал SIGHUP модулю **drweb-maild**, чтобы произошла перезагрузка библиотеки модуля. Значение по умолчанию: **PathToVadeRetro** = %var_dir/lib/libvaderetro.so
FullCheck = {логический}	R Параметр определяет стратегию проверки писем на спам. В случае если параметр имеет значение No, то письмо проверяется на признаки наличия спама только в случае если общая величина "положительных" характеристик письма (например, баллы, полученные за нахождение отправителя письма в белом списке) ниже порогового значения, встроенного в библиотеку VadeRetro. В противном случае (если параметр имеет значение Yes) письмо будет проверяться на все признаки спама вне зависимости от величины его "положительных" характеристик. Обратите внимание, что при включении безусловной полной проверки (**Fullcheck** = Yes) общая скорость проверки писем может уменьшиться, а кроме того, в этом случае возможно возникновение ситуаций, когда наличие отправителя письма в белом списке (см. ниже), если таковое имеет место, может быть не учтено и письмо все равно будет классифицировано как спам на основании анализа его внутреннего содержимого. Значение по умолчанию: **FullCheck** = Yes
NoHamFrom = {логический}	R При значении Yes данного параметра не будет производиться проверка на спам писем, направленных на адреса, находящиеся во встроенном списке ham библиотеки **VadeRetro** (ящики типа nospam@<domain>). Обратите внимание, что этот список встроен в библиотеку и не имеется возможности управлять его составом. Значение по умолчанию: **NoHamFrom** = Yes

AddXHeaders = {логический}	R Добавление к сообщению заголовков X-Drweb-SpamState и X-Drweb-SpamScore, содержащих информацию о том, является ли сообщение спамом и каков его итоговый счет по результатам проверки. Значение по умолчанию: **AddXHeaders** = Yes
AddVersionHeader = {логический}	R Добавление к сообщению заголовка X-Drweb-SpamVersion, содержащего информацию о версии подключаемого модуля **Vaderetro.** Значение по умолчанию: **AddVersionHeader** = No
AddXDrwebSpamStateNumHeader = {логический}	R Добавление к сообщению заголовка X-Drweb-SpamState-Num, в который входит числовое значение, присвоенное сообщению библиотекой **VadeRetro** по результатам классификации писем (каждое письмо будет отнесено к одному из указанных классов): • 0 - письмо не является спамом; • 1 - письмо является спамом; • 2 - письмо содержит вирус; • 3 - письмо является уведомлением DSN. Значение по умолчанию: **AddXDrwebSpamStateNumHeader** = No
AddXSpamLevel = {логический}	R Добавление к сообщению заголовка X-Spam-Level, состоящего из символов *. Один символ * добавляется за каждые 10 очков, присвоенных письму. Например, при счете 110 к письму будет добавлено: X-Spam-Level: **********. Значение по умолчанию: **AddXSpamLevel** = No
CheckForViruses = {логический}	R Эвристическая проверка сообщений библиотекой **VadeRetro** на наличие вирусов. В случае обнаружения вируса библиотека **VadeRetro** отнесет сообщение к классу 2. Значение по умолчанию: **CheckForViruses** = Yes
CheckDelivery = {логический}	R Определяет, следует ли проверять на признаки спама письма, классифицированные библиотекой **VadeRetro** как уведомления DSN (сообщения класса 3). В случае обнаружения признаков спама библиотека **VadeRetro** отнесет сообщение к классу 1. Значение по умолчанию: **CheckDelivery** = No
AllowRussian = {логический}	R Определяет, добавлять или нет дополнительные баллы к счету письма, если оно содержит кириллический текст (если установлено в Yes, дополнительные баллы добавлены не будут).

	Значение по умолчанию: **AllowRussian** = Yes
AllowCJK = {логический} R	Определяет, добавлять или нет дополнительные баллы к счету письма, если оно содержит текст на китайском, японском или корейском языке (если установлено в Yes, дополнительные баллы добавлены не будут). Значение по умолчанию: **AllowCJK** = Yes
WhiteList = {LookupLite} R A	Белый список отправителей. Адресами отправителя считаются адреса из полей From и Return-Path в теле письма. Если тело письма не будет содержать полей From и Return-Path, либо если перед полем From в теле письма будет стоять одна или несколько пустых строк – то поиск отправителя в белом списке производится не будет. Если в теле содержатся два поля From, то адрес будет взят из первого найденного поля. Допустимо использование в списке шаблонов вида *@<domain> (все адреса, принадлежащие конкретному домену). Например, чтобы внести в белый список все почтовые адреса домена mycompany.com, достаточно указать *@mycompany.com. Указываемые домены должны быть FQDN. Если адрес отправителя был найден в белом списке, то от общего счета письма отнимается 5 000 баллов. Если в белом списке найдены адреса из обоих полей (From и Return-Path), то от счета письма отнимется 10 000 баллов. Обратите внимание на следующие особенности работы с белым списком: 1. Поскольку белый список не сортируется, один и тот же адрес может быть случайно указан в нем несколько раз. В таком случае 5 000 баллов будут отниматься от общего счета письма столько раз, сколько раз адрес отправителя (из поля From или поля Return-Path) встречается в списке (например, если встретился 3 раза, то отнимется 15 000 баллов). 2. Если в письме домен адреса отправителя (из поля From или из поля Return-Path) совпадает с доменом адреса получателя (из поля To), и этот же домен присутствует в белом списке в виде шаблона (*@<domain>), то проверка адреса отправителя письма по белому списку не производится. Аналогичное поведение, если в письме указан один и тот же адрес отправителя и получателя, и этот же адрес присутствует в белом списке. 3. Если домен адреса отправителя <domain1> не совпадает с доменом получателя <domain2> в поле To и оба домена представлены в белом списке в виде шаблонов (*@<domain1> и *@<domain2> соответственно), то проверка письма производится, причем от его счета будет отнято 10 000 баллов. Аналогичное поведение, если адреса отправителя и получателя не совпадают, и оба присутствуют в белом списке. Обратите внимание, что в случае если **FullCheck** = Yes, то этот параметр может не иметь никакого эффекта: если письмо будет признано спамом по анализу содержимого, то вычисленные здесь баллы будут пригнорированы и не будут

	вычтены из общего счета письма (см. выше).
	Обратите внимание, что значение параметра – LookupLite.
	Пример:
	`hello@myneighbourhood.co.uk`
	`*@mycompany.com`
	Значение по умолчанию:
	WhiteList =

BlackList = {LookupLite} R A	Черный список отправителей. Аналогично белому списку отправителей (см. параметр **WhiteList**). Указываемые домены должны быть FQDN. Если адрес отправителя был найден в черном списке, то к общему счету письма прибавится 5 000 баллов. Если в черном списке найдены адреса из обоих полей (From и Return-Path), то к счету письма прибавится 10 000 баллов. Обратите внимание на следующие особенности работы с черным списком: 1. Поскольку черный список не сортируется, один и тот же адрес может быть случайно указан в нем несколько раз. В таком случае 5 000 баллов будут прибавляться к общему счету письма столько раз, сколько раз адрес отправителя (из поля From или поля Return-Path) встречается в списке (например, если встретился 3 раза, то прибавится 15 000 баллов). 2. Если в письме домен адреса отправителя совпадает с доменом адреса получателя, и этот же домен присутствует в черном списке в виде шаблона (*@<domain>), то проверка адреса отправителя письма по черному списку **производится**. Аналогичное поведение, если в письме указан один и тот же адрес отправителя и получателя, и этот же адрес присутствует в черном списке. Обратите внимание, что значение параметра – LookupLite. Значение по умолчанию: **BlackList** =
SpamThreshold = {числовое значение} R	Пороговое значение суммы баллов письма для отнесения его модулем **Vaderetro** к спаму (выставление заголовка письма **X-Drweb-SpamState** в Yes). Если величина суммы баллов письма больше либо равна порогу **SpamThreshold**, но меньше величины порога **UnconditionalSpamThreshold** (см. ниже), то к письму применяется действие, заданное параметром **Action**, кроме того, в начало темы сообщения добавляется текст, указанный в параметре **SubjectPrefix** (см. ниже). Величина параметра **SpamThreshold** должна быть меньше либо равна значению параметра **UnconditionalSpamThreshold**. Пожалуйста, обратите внимание, что подключаемый модуль **Vaderetro** классифицирует письмо как "не спам" или "спам" только на основании соотношения суммы баллов письма, присвоенных ему библиотекой **VadeRetro**, и порогового значения **SpamThreshold**. При этом не принимается во внимание класс, к которому библиотека **VadeRetro** отнесла письмо. Например, письмо может быть классифицировано

	самой библиотекой как спам и отнесено к классу 1, но если количество баллов, присвоенных библиотекой, будет меньше указанного здесь порога, то письмо не будет признано подключаемым модулем **Vaderetro** как спам (заголовку письма **X-Drweb-SpamState** будет присвоено значение `No`), и следовательно, к нему не будет применено действие, заданное для спама. Значение по умолчанию: **SpamThreshold** = 100
UnconditionalSpamThreshold = {числовое значение}	R Пороговое значение суммы баллов письма для отнесения его подключаемым модулем **Vaderetro** к безусловному спаму. Если величина суммы баллов письма больше либо равна порогу **UnconditionalSpamThreshold**, то письмо помечается как спам (заголовку письма **X-Drweb-SpamState** будет присвоено значение `Yes`), но к письму применяется действие, заданное параметром **UnconditionalAction**, кроме того, в начало темы сообщения добавляется текст, указанный в параметре **UnconditionalSubjectPrefix** (см. ниже). Величина параметра **UnconditionalSpamThreshold** должна быть больше либо равна значению параметра **SpamThreshold**. Значение по умолчанию: **UnconditionalSpamThreshold** = 1000
Action = {список действий}	R Действия, совершаемые с письмом, если оно по сумме баллов определено подключаемым модулем **Vaderetro** как спам. Обязательно должно быть указано одно из основных действий: pass, reject, discard, tempfail. Также может быть указано одно или несколько дополнительных действий: quarantine, redirect, add-header, score. Значение по умолчанию: **Action** = pass
UnconditionalAction = {список действий}	R Действия, совершаемые письмом, если оно по сумме баллов определено подключаемым модулем **Vaderetro** как безусловный спам. Обязательно должно быть указано одно из основных действий: pass, reject, discard, tempfail. Также может быть указано одно или несколько дополнительных действий: quarantine, redirect, add-header, score. Значение по умолчанию: **UnconditionalAction** = pass
NotifyAction = {список действий}	R Действия, совершаемые с письмом, если оно по сумме баллов определено как спам или безусловный спам подключаемым модулем **Vaderetro**, а кроме того классифицировано библиотекой **VadeRetro** как уведомление DSN (сообщение класса 3). Обязательно должно быть указано одно из основных

	действий: `pass`, `reject`, `discard`, `tempfail`. Также может быть указано одно или несколько дополнительных действий: `quarantine`, `redirect`, `add-header`, `score`. Значение по умолчанию: **`NotifyAction`** = `pass`
`SubjectPrefix` = `{текст}`	R Префикс, добавляемый к теме сообщения, если оно классифицировано подключаемым модулем **Vaderetro** по сумме баллов как спам (сумма баллов письма больше либо равна значению параметра **`SpamThreshold`**). См. важную сноску под таблицей. Значение по умолчанию: **`SubjectPrefix`** =
`UnconditionalSubjectPrefix` = `{текст}`	R Префикс, добавляемый к теме сообщения, если оно классифицировано подключаемым модулем **Vaderetro** по сумме баллов как безусловный спам (сумма баллов письма больше либо равна значению параметра **`UnconditionalSpamThreshold`**). См. важную сноску под таблицей. Значение по умолчанию: **`UnconditionalSubjectPrefix`** =
`NotifySubjectPrefix` = `{текст}`	R Префикс, добавляемый к теме сообщения, если оно по сумме баллов определено как спам или безусловный спам подключаемым модулем **Vaderetro**, а кроме того классифицировано библиотекой **`VadeRetro`** как уведомление DSN (сообщение класса 3). См. важную сноску под таблицей. Значение по умолчанию: **`NotifySubjectPrefix`** =
`FromProtectedNetworkScoreAdd` = `{числовое значение}`	R Значение, которое всегда прибавляется к текущей сумме баллов письма, если оно идет из сети, указанной в перечне **`ProtectedNetworks`** (параметр в секции `[Maild]` основного конфигурационного файла **Dr.Web MailD**). Прибавляемое значение может быть отрицательным (для уменьшения счета). Для отключения этой функции можно указать значение `0`. Значение по умолчанию: **`FromProtectedNetworkScoreAdd`** =
`UseReplyCache` = `{Yes | No}`	Управляет (включает или отключает) использованием кэша `reply_cache`. Этот кэш используется для временного хранения данных о письме (адреса всех его получателей) для того, чтобы учесть их при антиспам-проверке писем, которые будут следовать в обратном направлении и являться ответами на проверенное письмо (Reply-to). Если указано значение `Yes`, кэш `reply_cache` включен и используется.

	Обратите внимание, что содержимое кэша `reply_cache` сбрасывается при каждой перезагрузке подключаемого модуля **Vaderetro**. Значение по умолчанию: **UseReplyCache** =
ProtectedNetworkReplyCacheLifeTime = {время}	Задает время хранения в кэше `reply_cache` записей о письмах, которые идут из сетей, указанных в перечне **ProtectedNetworks** (параметр в секции `[Maild]` основного конфигурационного файла **Dr.Web MailD**). Если добавляемый адрес уже имеется в `reply_cache`, запись для него обновляется. Значение по умолчанию: **ProtectedNetworkReplyCacheLifeTime** =
ReplyToProtectedNetworkScoreAdd = {числовое значение}	Значение, которое всегда прибавляется к текущей сумме баллов письма, если его отправитель находится в кэше `reply_cache`. Прибавляемое значение может быть отрицательным (для уменьшения счета). Для отключения этой функции можно указать значение `0`. Кроме того, если сам `reply_cache` отключен (т.е. **UseReplyCache** = No), то отправитель никогда не будет найден там, и потому любое указанное здесь значение фактически не будет использоваться. Значение по умолчанию: **ReplyToProtectedNetworkScoreAdd** =

В тех случаях, когда сообщение блокируется (`reject`) модулем в синхронном режиме обработки, ответ **Dr.Web MailD** клиенту состоит из кода SMTP (`55*` или `250`, в зависимости от значения параметра **ReturnReject** в секции `[Receiver]`), а также текстового сообщения, содержание которого может задаваться идущими далее параметрами. Значения параметров должны быть заключены в кавычки.

UseCustomReply = {логический}	Использование настраиваемых сообщений в SMTP-сессии для случаев, когда сообщения отклоняются. Значение по умолчанию: **UseCustomReply** = No
SpamCustomReply = {текст}	Настраиваемое сообщение в SMTP-сессии для случаев, когда подключаемым модулем **Vaderetro** выполняется действие `reject` для параметров **Action**, **UnconditionalAction**, **NotifyAction**, если **UseCustomReply** = yes. Вы можете задать только текстовую часть сообщения. Текст, содержащий пробелы, должен быть заключен в кавычки. **Пример**: `550 5.7.0 "Text part of reply"` Значение по умолчанию: **SpamCustomReply** = "Dr.Web vaderetro plugin: this is spam!"

Если **UseCustomReply** = No или соответствующая строка текста не задана, будет выдано стандартное сообщение "The message has been rejected by the Dr.Web MailD".

Пожалуйста, обратите внимание, что **во всех параметрах**, меняющих или ищущих значение некоторого заголовка или части письма, следует **учитывать кодировку**, в которой встречается искомый текст.

Правила работы со значениями заголовков в произвольных кодировках рассмотрены в разделе Работа со строковыми значениями.

Примеры

Примеры настройки действия антиспам-модуля **Vaderetro**:

1. Письмо, набравшее баллов больше, чем в **`Threshold`**:
   - Будет заблокировано (клиент получит в ответ код SMTP 550);
   - Помещено в **Карантин**;
   - Копия письма будет перенаправлена на адрес, заданный в параметре **`AdminMail`** секции `[Notifier]` конфигурационного файла **Dr.Web MailD**).

   ```
   Action = reject, quarantine, redirect
   ```

2. Аналогично 1, только копии письма будут разосланы по указанным адресам (на адрес, указанный в **`AdminMail`** секции `[Notifier]`, ничего не придет):

   ```
   Action = reject, quarantine, redirect (admin1@domain | admin2@domain
   | admin3@domain)
   ```

Модуль Dr.Web Modifier

Назначение модуля

Подключаемый модуль **Dr.Web Modifier** используется для:

- *Контентного анализа писем* – поиска в телах обрабатываемых писем вложенных объектов с определенными MIME-типами (графика, исполняемые файлы, медиа-файлы), а также MIME-объектов, удовлетворяющих определенным условиям;
- *Модификации тел писем* – удаления MIME-объектов, удовлетворяющих определенным условиям, модификации заголовков и/или содержимого выбранных MIME-объектов;
- *Управления обработкой писем* – блокировки, помещения в **Карантин**, перенаправления, добавление заголовков и счета в зависимости от найденных в телах обрабатываемых писем MIME-объектов.

Ниже представлена краткая информация о структуре письма, которая используется при его анализе и обработке подключаемым модулем **Dr.Web Modifier**.

Письмо представляет собой составной объект, который может быть выбран для совершения действия либо целиком, либо по частям, поскольку каждое письмо можно представить как иерархический набор элементов (MIME-объекты, их заголовки и содержимое). Части письма также можно выбирать для индивидуальной обработки. На рисунке ниже показана иерархическая структура письма, содержащего вложенные объекты.

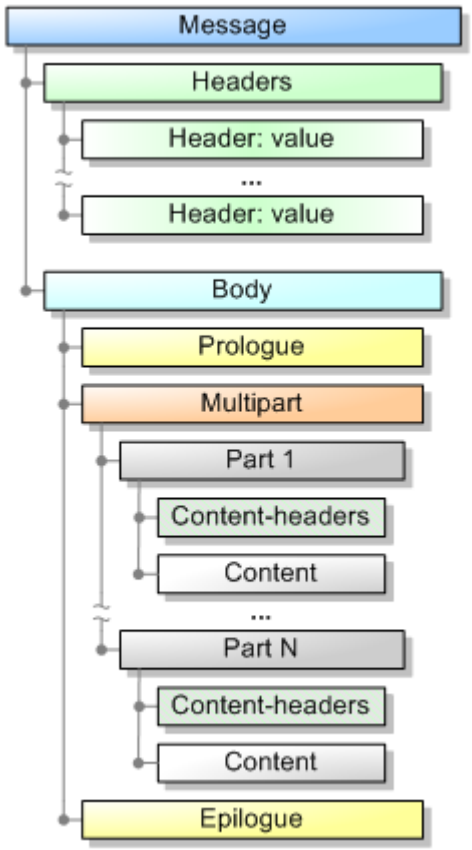


Рис. 19. Иерархическая структура письма, содержащего вложенные объекты

Письмо состоит из следующих элементов:

1. На верхнем уровне письмо разделяется на область заголовков **`Headers`** и тело письма **`Body`**.

2. Если MIME-тип содержимого в теле письма – `multipart/*`, то тело представляет собой составной объект, состоящий из следующих частей:
   - **`Prologue`** – пролог, некоторый текст, помещенный в тело до начала первого вложенного MIME-объекта. Этот текст не отображается MUA, поддерживающими MIME multipart, и как правило, отсутствует;
   - **`Epilogue`** – эпилог, некоторый текст, помещенный в тело после окончания последнего вложенного MIME-объекта. Этот текст не отображается MUA, поддерживающими MIME multipart, и как правило, отсутствует;
   - Если тело письма состоит из нескольких частей, то каждая из них рассматривается как отдельный MIME-объект, содержащий собственный набор заголовков (как правило, это контент-заголовки, такие, как `Content-type`, `Content-description`, `Content-disposition` и т.д.), содержимое (контент), и, возможно, пролог и эпилог. В свою очередь, MIME-объект тоже может оказаться составным (иметь тип `multipart/*`).

В противном случае тело письма рассматривается как монолитный объект, не содержащий вложенных частей, а также пролога и эпилога.

Поскольку структура любого вложенного объекта совпадает со структурой письма в целом (объект, как и все письмо, состоит из заголовка и тела, а также, возможно, имеет пролог и эпилог), то и само письмо в целом можно считать MIME-объектом, у которого **`Headers`** являются заголовками, а **`Body`** – содержимым (контентом). Этот MIME-объект называется **корневым**.

Для поиска объектов в письме **Dr.Web Modifier** поддерживает следующие версии регулярных выражений:

- Базовые (`basic regular expressions`);
- Расширенные (`extended regular expressions`);
- Perl-совместимые (`Perl-compatible regular expressions`).

С основами регулярных выражений вы можете ознакомиться, например, в **Wikipedia** (статья «Регулярные выражения»).

Порядок применения правил модификации при обработке писем

Подключаемый модуль обрабатывает письма, применяя **правила модификации**. Правила модификации делятся на локальные (local rules) и глобальные (global rules). Сначала при обработке письма применяются локальные правила, а затем – глобальные. Если для письма локальные правила модификации не заданы (это зависит от того, какие Правила обработки почты сработали для письма, см. ниже), то письмо проверяется сразу с использованием глобальных правил модификации. Если письмо было отклонено при использовании локальных правил модификации, то глобальные правила модификации к этому письму уже не применяются.

Локальные правила модификации **Dr.Web Modifier** задаются только в Правилах обработки почты, задаваемых в секции `[Rules]` конфигурационного файла **Dr.Web MailD** или во встроенной базе данных пользователей и групп. Для этого в Правилах используется параметр **`modifier/LocalRules`**. Глобальные правила модификации **Dr.Web Modifier** задаются только параметром **`GlobalRules`** в конфигурационном файле подключаемого модуля, в секции `[Modifier]`.

Обратите внимание, что параметр **`GlobalRules`** не может использоваться в Правилах обработки почты, а параметр **`LocalRules`** не может задаваться в конфигурационном файле подключаемого модуля.

Формат правил модификации

Любое правило модификации состоит из следующих частей:

```
<Оператор выборки элементов>, [<Оператор проверки условия>,] <Оператор действия>[,
<Оператор действия>, ...]
```

Обратите внимание, что операторы в правиле модификации разделяются запятыми. Операторов действия может быть более одного. Проверка условия может отсутствовать. Минимально допустимая форма правила модификации состоит из оператора выборки и оператора действия.

Правила модификации всегда записываются в одну строку. При необходимости разбиения правила модификации на несколько строк используйте символ '\', который следует указывать в конце строки непосредственно перед ее разрывом, например:

```
<Оператор выборки элементов>, \
<Оператор действия>, <Оператор действия>
```

Кроме того, в разделе "операторов действий" таже можно формировать новую выборку элементов и проверять условия и выполнять действия уже для нее.

1. Операторы формирования выборки

Выборка элементов из письма для последующей обработки осуществляется при помощи оператора **select** <element>, которому в качестве аргумента <element> указывается тип элементов, выбираемых из письма для анализа и обработки. Доступны следующие типы выделения:

- **message** – Выбор корневого MIME-объекта письма, т.е. всего письма целиком.
- **mime(**<ОБЛАСТЬ>**) | mime.**<ОБЛАСТЬ> [[<имя>] <регулярное_выражение>] – Выбор MIME-объектов или содержимого MIME-объектов, имеющих содержимое в указанной области.

  Различие между синтаксисом со скобками и с точкой заключается в том, что аргумент со скобками выбирает сами объекты, содержащие элементы в указанной области, а агрумент с точкой – только элементы этих объектов (конкретные заголовки, текст из пролога, содержимое тела, и т.д.), находящиеся в указанной области.

  В качестве <ОБЛАСТИ> могут быть использованы следующие значения:

  - headers – область заголовков MIME-объекта;
  - prologue – область пролога MIME-объекта;
  - body – область тела MIME-объекта (content);
  - epilogue – область поиск в эпилоге MIME-объекта.

  Дополнительно могут быть указаны следующие параметры:

  - <имя> – имя искомого заголовка. Задается только в том случае, если <ОБЛАСТЬ> – headers.
  - **<**регулярное_выражение**>** – шаблон поиска искомого элемента (например, шаблон, которому должно удовлетворять значение заголовка, или текст, содержащийся в элементе).

  Пример:

  Команда, выбирающая содержимое всех видео-фрагментов из письма:

  ```
  select mime(headers) Content-type "x-video"
  ```

  Команда, выбирающая информацию о типе данных из всех видео-фрагментов (значение из заголовка):

  ```
  select mime.headers Content-type "x-video"
  ```

  Также для заголовков (в целях совместимости с подключаемым модулем **Vaderetro**) можно использовать команды сравнения с целым числом ">" и "<" (знак экранирования '\' при этом не используется!). Сравниваемый заголовок считается удовлетворяющим условию отбора, если он содержит целое число (например: X-Drweb-SpamScore: "30"), которое удовлетворяет заданному условию.

Пример:

```
select mime(headers) X-Drweb-SpamScore "<50"
```

В результате выбираются элементы, в которых имеется заголовок `X-Drweb-SpamScore`, значение которого – целое число, меньшее 50. Обратите внимание, что в данном случае обратный слэш перед знаком "<" не нужен. Если бы правило модификации было задано как:

```
select mime(headers) X-Drweb-SpamScore "\<50"
```

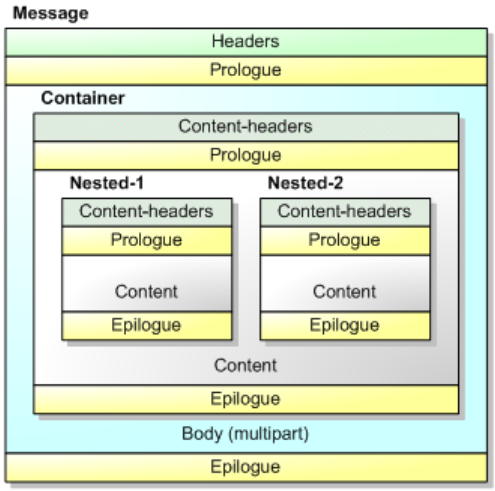
то в результате были бы выбраны элементы с заголовком `X-Drweb-SpamScore: "<50`

Обратите внимание, что при помощи команды **select mime**(<ОБЛАСТЬ>) нельзя выбрать составной MIME-объект, кроме тех случаев, когда он является корневым (письмом).

Например, если письмо обладает структурой, представленной на картинке ниже, команда **select mime**(headers) выберет следующие объекты, содержащие заголовки:

- **Message** (корневой объект "письмо");
- **Nested-1**;
- **Nested-2**.

Объект **Container** выбран не будет, поскольку он составной (содержит объекты **Nested-1** и **Nested-2**), но не является письмом.



- **sender** <регулярное_выражение>, **recipient** <регулярное_выражение> – Выбор всего сообщения, если оно содержит соответствующую запись об отправителе (получателях). Данные об отправителе (получателях) берутся из конверта письма. В случае нахождения искомой последовательности символов действие аналогично команде **select message** (выделение всего сообщения).

  Пример:

  Если письмо предназначено для администратора, можно добавить в конец письма приветствие с помощью команды **append_text** (см. ниже):

  ```
  select recipient "root@localhost", \
  append_text "hello, root"
  ```

  Ясно, что если письмо предназначено не администратору, то выбранное множество элементов будет пусто, поэтому в этом случае фактически команда **append_text** применена не будет. Обратите внимание на эту особенность, она позволяет не применять операторы проверки условия, если альтернативное действие (когда нужные

элементы не найдены) отсутствует.

- **select_mimes** – Команда позволяет перейти от выборки заголовков к выборке MIME-объектов, их содержащих. Это позволяет ускорить работу подключаемого модуля в случаях, когда требуется сперва выбрать некие заголовки, а потом сам объект по одному и тому же критерию. Для выборки объекта достаточно, чтобы был выбран хотя бы один компонент этого объекта, не считая вложенных MIME-объектов для составного MIME-объекта.

Пожалуйста, обратите внимание, что **во всех правилах модификации**, использующих значение некоторого заголовка или части письма, следует **учитывать кодировку**, в которой встречается искомый текст.

Правила работы со значениями заголовков в произвольных кодировках рассмотрены в разделе Работа со строковыми значениями.

Для того, чтобы выбрать из письма элементы, соответствующие нескольким критериям, используются логические операторы объединения и пересечения последовательно указанных подвыборок. Для этого в одном операторе **select** указывается несколько аргументов выборки, которые соединяются логическими операторами:

- **and** – оставить в выборке, полученной предыдущим аргументом, только те элементы, которые попадут под указанное условие.
- **nand** – оставить в выборке, полученной предыдущим аргументом, только те элементы, которые НЕ попадут под указанное условие.
- **or** – добавить в выборку, полученную предыдущим аргументом, те элементы, которые попадут под указанное условие.
- **nor** – добавить в выборку, полученную предыдущим аргументом, те элементы, которые НЕ попадут под указанное условие.

Обратите внимание, что операторы объединения и пересечения выборок работают только с выборками, содержащими MIME-объекты, но не работают с выборками, содержащими сами элементы, т.е. не применимы к синтаксису **select mime.**<ОБЛАСТЬ>

Пример:

Выбрать фрагменты, написанные на html, и содержащие слово "<script":

```
select mime(headers) Content-type html and mime(body) "\<script"
```

По сути это две выборки, применяющихся последовательно. Первая выбирает все MIME-объекты, содержащие в заголовке Content-type слово "html", а вторая оставляет в списке выбранных объектов только те, которые содержат последовательность символов "<script" в любом регистре.

Пример:

```
select mime(headers) Content-type html nand mime(body) "\<script"
```

В соответствии с первым критерием будут выбраны все MIME-объекты, содержащие в заголовке Content-type слово "html". В соответствии со вторым критерием из этой выборки будут исключены все MIME-объекты, содержащие последовательность символов "<script" в любом регистре.

Пример:

```
select mime(headers) Content-type html or mime(body) "\<script"
```

В соответствии с первым критерием будут выбраны все MIME-объекты, содержащие в заголовке Content-type слово "html". В соответствии со вторым критерием к этой выборке будут добавлены также все MIME-объекты, содержащие последовательность символов "<script" в любом регистре.

Пожалуйста, обратите внимание на типичные ошибки в использовании операторов выборки:

- Если указывается несколько операторов `select` подряд, то последующий `select` вытесняет собой результаты выборки, полученные предыдущим оператором `select`:

  Пример:

  ```
  select mime(headers) Content-type html, select mime(body) "\<script"
  ```

  ```
  select mime(headers) Content-type html and select mime(body) "\<script"
  ```

  Полученная в итоге выборка будет осуществлена только в соответствии со вторым критерием – т.е. будут выбраны только фрагменты письма (MIME-объекты), содержащие последовательность символов `"<script"` в любом регистре.

- Если ни логических операторов, ни команды `select` перед последующим аргументом не указано, то он игнорируется и выборка не изменяется.

  Пример:

  ```
  select mime(headers) Content-type html mime(body) "\<script"
  ```

  Выборка будет осуществлена только в соответствии с первым критерием – т.е. будут выбраны только фрагменты письма (MIME-объекты), содержащие в своем заголовке `Content-type` слово `"html"`.

2. Операторы действия

Действия всегда применяются по результатам выборки оператора **`select`**. Действия разделяются на три типа:

- Действия, применяемые ко всему письму (отклонить, перенаправить, сформировать уведомление MailD и т.п.);
- Действия, применяемые к выбранным элементам письма (удаление, добавление подписи, замена или модификация текста и т.п.);
- Действия, изменяющие счет письма (для антиспам).

Если множество выбранных элементов пусто, то указанные в правиле действия не выполняются (игнорируются).

2.1. Действия, воздействующие на письмо целиком

Существуют следующие операторы воздействия на письмо целиком:

- **`pass`**, **`accept`** – пропустить письмо. В случае глобальных правил модификации после получения любой из этих команд дальнейшая обработка письма не ведется. В случае локальных правил модификации после получения команды **`accept`** **Dr.Web Modifier** переходит к обработке письма с помощью глобальных правил модификации;
- **`reject`** – отклонить письмо с уведомлением об этом отправителя;
- **`discard`** – отклонить письмо, не уведомляя об этом отправителя;
- **`notify`** `<report_name>` – оповестить администратора, обработка письма не прекращается. После этой команды необходимо указать имя шаблона уведомления, который будет использован при оповещении, иначе при обработке письма будут возникать ошибки. Шаблоны находятся в каталоге, указанном в значении параметра **`TemplatesBaseDir`** в секции `[Notifier]` конфигурационного файла **Dr.Web MailD**.

  Пример :

  ```
  GlobalRules = select message, notify rule
  ```

  В данном случае просто выполняется отправка уведомления MailD. Так как любому оператору действия должен предшествовать оператор выборки, применяется оператор **`select message`**, просто выбирающий все письмо и гарантированно возвращающий непустой результат. Нужный префикс `admin_` и расширение `.msg` для формирования

уведомления будут автоматически подставлены компонентом **Dr.Web Notifier**.

- **tempfail** – уведомить отправителя о сбое сервера;
- **redirect** – перенаправить письмо на заданный адрес;
- **quarantine** – отправить письмо в **Карантин**.
- **stop** – прекращает обработку правил модификации. В отношении письма принимается решение согласно ранее выполненным командами **pass**, **accept**, **reject** и т.д., в зависимости от того, какая команда была выполнена последней. Команда **accept** равноценна комбинации **pass+stop**, за исключением того, что **stop** прекращает обработку полностью, а **accept** – только для локальных правил модификации. Для глобальных правил модификации **accept** равноценна **pass**.

Команды **reject**, **discard** и **tempfail** являются "решающими" – после них обработка письма прекращается, вне зависимости от того, указаны ли в этом правиле модификации еще какие-либо команды.

2.2. Действия, преобразующие выбранные элементы письма

Эти действия действуют только на содержимое выбранных элементов, если не указано иное.

- **replace** <выражение для замены> <заменяемое регулярное выражение>, **replace_all** <новый текст> – Данные действия заменяют в выбранных элементах текст, удовлетворяющий <заменяемому регулярному выражению> на текст, указанный в <выражении для замены>.

  Пример:

  Поиск и переименование исполняемых файлов во вложениях (изменяется содержимое заголовков Content-disposition):

  ```
  select mime.headers Content-disposition "filename=.*\\.exe",\
  or mime.headers Content-type "name=.*\\.exe",\
  replace "\\.ex_" "\\.exe", pass
  ```

  Эти команды не работают для многокомпонентных частей сообщений. Т.е. для сообщения, состоящего, к примеру, из многокомпонентного MIME-объекта с двумя вложенными объектами (как, например, объект **Container** на картинке, приведенной выше) правило модификации:

  ```
  select message, replace_all "text"
  ```

  не произведет никакого эффекта, поскольку многокомпонентные объекты сами по себе не содержат данных, а лишь служат контейнерами для других объектов.

  Для команд **replace** и **replace_all** в аргументах <выражение для замены> и <новый_текст> можно использовать вызовы функций в виде **${func_name}**. Аргументом для этих функций является текущее заменяемое выражение. Реализованы следующие функции:

  - **urlencode** – кодирование аргумента в строку, которую можно использовать в качестве URL;
  - **self** – вернуть само выражение без изменений.

  Пример:

  ```
  select mime.headers "Subject" "^.*$", replace_all "[SPAM] ${self}"
  ```

  В начало заголовка Subject письма будет вставлена строка "[SPAM]". Например, заголовок "This is Subj" будет заменен на "[SPAM] This is Subj".

Пример:

```
select mime.body ".*", replace "http://check-url.com?url=${urlencode}"
"http://\\S+"
```

В теле письма текст, соответствующий указанному шаблону, например: "Visit http://vasya.com?id=3", будет заменен на "Visit http://check-url.com?url=http%3A%2F%2Fvasya%2Ecom%3Fid%3D3".

- **remove** – Данная команда удаляет любые виды выбранных объектов, кроме корневого MIME-объекта "письмо" (т.е. при помощи нее письмо целиком удалить нельзя!).

  Пример:

  Например, нельзя использовать команду **remove** в правилах модификации вида:

```
GlobalRules = select mime(body) "text", remove, pass
GlobalRules  = select mime(body) "script", remove, pass
```

- **prepend_text**, **append_text**, **prepend_html**, **append_html** – Данные команды добавляют в выбранные MIME-объекты фрагмент в формате plain-text или html.

  Команды имеют необязательный аргумент [[7b:]encoding]. В нем слово encoding указывает название кодировки добавляемого текста, а префикс "7b:" указывает на использование 7-битной кодировки context transfer encoding (CTE) 7bit. Если префикс "7b:" не указан, то будет использоваться кодирование CTE 8bit. Если не указан encoding, то используется кодировка, указанная в параметре **Encoding** в конфигурационном файле подключаемого модуля.

  Пример:

```
select message, append_html "<h1>checked by antispam</h1>"
```

Кроме того, в качестве источника данных данных для вставки текстов в конкретной кодировке может также служить языковой lng-файл подключаемого модуля **Dr.Web Modifier**. Для использования строк из lng-файла следует использовать формат записи $n, где n – номер строки в lng-файле. Подробнее об использовании языковых файлов см. в разделе Языковые файлы.

Пример:

Пусть в lng-файле есть строка:

```
...
782 = "строка текста"
...
```

тогда выражение **append_text** $782 будет эквивалентно выражению **append_text** "строка текста".

Обратите внимание, что команды prepend_text, append_text, prepend_html, append_html вставляют в выбранные объекты не просто текст, а полноценный MIME-объект с текстовым содержимым (в начало или в конец выбранных по **select** объектов соответственно), в результате чего измененные объекты становятся составными. Поэтому добавление текстовой информации всегда ведет к сбросу списка выбранных объектов. Для продолжения обработки письма после вставки в него какого-либо текста следует повторить операцию выборки объектов **select**.

- **addheader** – Добавление заголовков в выбранные MIME-объекты.

  Пример:

  ```
  select message, addheader "foo:bar"
  ```

  Эта команда добавит к письму заголовок с именем foo и значением bar. Имя и значение заголовка отделяются друг от друга двоеточием (":").

Как для поиска текста, так и для его вставки следует учитывать кодировку вставляемого текста. Текст всегда вставляется в той кодировке, в которой создан конфигурационный файл (за исключением команд **prepend_text**, **append_text**, **prepend_html** и **append_html**, использующих для вставки текста либо кодировку, явно указанную в команде, либо кодировку, заданную в параметре **Encoding** конфигурационного файла подключаемого модуля). Для использования других кодировок следует использовать явное кодирование строковых значений.

2.3. Действия, изменяющие счет письма

Существует возможность проверки или изменения счета, присвоенного письму. Письму присваивается числовая оценка (score – счет), первоначально равная нулю. При обработке письма подключаемые модули могут менять эту оценку. При помощи команд **add_score** и **set_score** можно произвести изменение этой оценки. В том числе – уменьшить ее, для чего в команде **add_score** можно указывать отрицательное число.

Пример:

```
set_score 10
```

Устанавливает для письма счет 10.

Пример:

```
add_score 11
```

Увеличивает счет письма на 11.

Аргумент **score** может быть 32-битным целым числом из диапазона от -2 млрд до +2 млрд. Следует учитывать возможность переполнения score при операциях с ним и вызванной этим переполнением некорректной работы других модулей, обрабатывающих письмо. Поэтому настоятельно рекомендуется избегать использования неоправданно больших значений score в правилах модификации (к примеру, не задавать 2 000 000 000 в качестве параметра действия **add_score**).

2.4. Применимость действий к различным объектам письма

В **Таблице 1** показывается, какой эффект оказывают действия в зависимости от того, к какому множеству, выбранному **select**, они применяются.

Таблица 1. Воздействие команд модификации на различные типы выбранных элементов

	remove	replace	replace all	append text	prepend text	append html	prepend html	add header	add score	set score	accept	discard	reject	tempfail	notify	redirect	quarantine
mime.headers	+	+	+	-	-	-	-	-	+	+	+	+	+	+	+	+	+
mime.prologue	+	+	+	-	-	-	-	-	+	+	+	+	+	+	+	+	+
mime.epilogue	+	+	+	-	-	-	-	-	+	+	+	+	+	+	+	+	+
mime.body	+	+	+	-	-	-	-	-	+	+	+	+	+	+	+	+	+
mime(headers)	+	*	*	+	+	+	+	+	+	+	+	+	+	+	+	+	+
mime(prologue)	+	*	*	+	+	+	+	+	+	+	+	+	+	+	+	+	+
mime(epilogue)	+	*	*	+	+	+	+	+	+	+	+	+	+	+	+	+	+
mime(body)	-	*	*	+	+	+	+	+	+	+	+	+	+	+	+	+	+
sender	-	-	-	+	+	+	+	+	+	+	+	+	+	+	+	+	+
recipient	-	-	-	+	+	+	+	+	+	+	+	+	+	+	+	+	+
message	-	-	-	+	+	+	+	+	+	+	+	+	+	+	+	+	+

В таблице используются следующие условные обозначения:

***** – такое же, как для **mime.body**;

+ – применимо;

- – игнорируется.

Обратите внимание, что действия **prepend_text**, **prepend_html**, **append_text**, **append_html** и **addheader** можно применять только в том случае, если при помощи **select** выбран составной, а не простой объект (такой, например, как значение заголовка) поскольку они всегда вставляют в выбранный объект дополнительный MIME-объект, который не может являться значением заголовка.

3. Операторы проверки условий

Операторы проверки условий применяются для изменения порядка действий над сообщением в зависимости от того, выполняется условие, или нет. Основные виды условий:

1. Ветвление по наличию выборки:

```
if [not] found, <действие или список действий>, \
[else, <действие или список действий>,] endif
```

Условие **if found** истинно, если выборка, сформированная предшествующим оператором **select**, не пуста. Условие **if not found** истинно в противоположном случае. В случае истинности условия будет выполнен список действий, указанных после него, до тех пор, пока не встретится оператор окончания ветвления **endif** или оператор альтернативной ветви **else**. Альтернативная ветвь **else**, <действие или список действий> может отсутствовать. В этом случае при ложности условия осуществляется переход сразу к оператору, следующему за **endif**.

В разделе <действие или список действий> может также выполняться формирование новой выборки при помощи оператора **select**, и, соответственно, выполняться проверка вложенных условий (**if** или **goto**). Новая выборка будет замещать предыдущую (которая использовалась для проверки условия). Например:

```
select <A>, if found, select <B>, reject, endif
```

В данном случае из письма выбирается некоторое множество элементов, обозначенное как <A>. Если оно пусто, то переход к **endif**, окончание работы правила модификации (никакого решения по письму не принято). Иначе из этого же письма формируется новая выборка, обозначенная в примере как <B> (при этом она замещает собой для последующих операторов действия ранее выбранную выборку <A>) Если она не пустая, то письмо отклоняется. В противном случае никакого решения по письму не принимается (действие **reject** игнорируется для пустой выборки <B>).

2. Ветвление по значению счета письма:

```
if score <op_value>, <действие или список действий>, \
[else, <действие или список действий>,] endif
```

Условие **if score** истинно, если текущее значение счета письма соответствует заданному выражению сравнения. Команда **if score** работает аналогично команде **if found**, но проверяет не наличие выбранных элементов, а только оценку, т.е. она игнорирует результаты предыдущих команд **select**.

Аргумент <op_value> должен быть записан одной строкой, без пробелов – т.е. '<100', но не '< 100', и должен состоять из символа операции сравнения и целочисленного значения. Для **if score** возможны следующие операции сравнения:

- **if score** <N – если score меньше N;
- **if score** >N – если score больше N;
- **if score** =N – если score равно N.

Аргумент N может быть 32-битным целым числом из диапазона от -2 млрд до +2 млрд. Следует учитывать возможность переполнения score при операциях с ним, и вызванной этим переполнением некорректной работы других модулей, обрабатывающих письмо. Поэтому настоятельно рекомендуется избегать использования неоправданно больших значений score в правилах модификации (к примеру, не задавать 2 000 000 000).

3. Операторы перехода:

- **goto** N – безусловный переход на N команд в списке действий вперед;
- **goto**(y) N – условный переход на N команд в списке действий вперед, если множество элементов не пустое;
- **goto**(n) N – условный переход на N команд в списке действий вперед, если множество элементов пустое.

Аргументом служит положительное целое число, указывающее, сколько команд следует пропустить.

Примеры:

```
GlobalRules = select mime.headers Subject "word1|word2|wordN", \
if found, notify rule, quarantine, reject, endif
```

При наличии в заголовке письма слов "word1", "word2", либо "wordN" письмо копируется в **Карантин**, после чего администратору передаётся уведомление MailD и письмо отклоняется.

```
GlobalRules = select mime.headers Subject "word1|word2|wordN", \
if found, reject, notify rule, quarantine, endif
```

В данном примере команды **notify** и **quarantine** не будут выполняться, поскольку на команде **reject** письмо отклоняется и его обработка останавливается (команда **reject** – "решающая").

```
GlobalRules = select mime(header) Content-type "executable", \
goto(n) 1, reject
```

Этот набор команд позволяет отклонить письма, в которые вложены исполняемые файлы.

```
GlobalRules = select mime.headers "X-DrWeb-SpamState" "yes", \
if found, select mime(headers) Content-type "image", \
remove, endif
```

Этот набор команд позволяет удалить картинки из письма, отмеченного подключаемым модулем **Vaderetro** как спам.

Примеры на изменение счета письма:

Установка счета 10 для письма, удовлетворяющего некоему условию:

```
select <...> if found, set_score 10, endif
```

Если счет письма превышает 100, то письмо отклоняется. В противном случае его счет уменьшается на 5:

```
select <...> if score >100, reject, else, \
add_score -5, endif
```

Более подробные примеры правил модификации подключаемого модуля **Dr.Web Modifier** показаны в разделе Примеры.

Правила экранирования служебных символов

В правилах модификации, задаваемых для подключаемого модуля, **крайне не рекомендуется** использовать значения и регулярные выражения, содержашие символ кавычки (") или символ обратного слэша "\", поскольку их экранирование (символом "\") может привести к ошибке разбора строки в конфигурационном файле **Dr.Web MailD**. Однако если без использования этих символов не обойтись, то помните следующие правила:

- при использовании в правилах модификации кавычек может потребоваться экранировать их несколькими символами "\". В текущей версии **Dr.Web MailD** для экранирования кавычки требуется шесть символов "\".
- для экранирования символа "\" перед ним необходимо повторить его же 7 раз.
- одиночная кавычка (апостроф) ' не экранируется.

Примеры:

Отклонение писем с темой, содержащей кавычку и состоящей из одного символа '\' соответственно:

```
GlobalRules = select mime.headers Subject ".*\\\\\\"", \
if found, reject, endif
```

```
GlobalRules  = select mime.headers Subject "^\\\\\\\\$", \
if found, reject, endif
```

Отклонение писем с темой, содержащей текст `text'"\quoted\"text`:

```
GlobalRules = select mime(headers) Subject \
"text'\\\\\\"\\\\\\\\quoted\\\\\\\\\\\\\\\"text", \
if found, reject, endif
```

Подключение модуля

Чтобы подключить модуль **Dr.Web Modifier** к программному комплексу **Dr.Web для почтовых серверов UNIX**, достаточно в конфигурационном файле **Dr.Web MailD** добавить строку `modifier` в список подключаемых модулей, обрабатывающих письма.

Если письма должны обрабатываться модулем **Dr.Web Modifier** до помещения в базу данных, то его название следует добавлять в список значений параметра **BeforeQueueFilters** секции `[Filter]` конфигурационного файла **Dr.Web MailD**.

Пример:

```
BeforeQueueFilters = modifier
```

Если же письма должны обрабатываться модулем уже после помещения в базу данных, то его название добавляется в список значений параметра **AfterQueueFilters** секции `[Filter]` конфигурационного файла **Dr.Web MailD**.

Пример:

```
AfterQueueFilters = modifier
```

Настройка модуля

Все основные параметры работы подключаемого модуля задаются (по умолчанию) в файле `%etc_dir/plugin_modifier.conf`. Устройство конфигурационного файла и краткое описание его параметров приведены в в разделе Конфигурационные файлы.

В секции `[Modifier]` собраны общие настройки работы подключаемого модуля **Dr.Web Modifier**:

Секция `[Modifier]`

GlobalRules = {список правил}	Список глобальных правил модификации, используемых модулем для обработки писем.
	Пример:
	Данное правило модификации добавляет в письмо строку подписи в формате html:
	`GlobalRules = select message, append_html "<b>Проверено!</b>"`
	Это правило модификации удаляет письма от выбранных пользователей:
	`GlobalRules = select mime(headers) From "weirdohacker@server.net", if found, reject, endif`
	Значение по умолчанию:
	`GlobalRules =`

Encoding = {кодировка} R	Кодировка, которую модуль будет указывать для текста, вставляемого командами `append_text` и `prepend_text` непосредственно из правил модификации. Значение по умолчанию: `Encoding = koi8-r`

В тех случаях, когда сообщение блокируется (`reject`) подключаемым модулем в синхронном режиме обработки, ответ **Dr.Web MailD** клиенту состоит из кода SMTP (`55*` или `250`, в зависимости от значения параметра **`ReturnReject`** в секции `[Receiver]`), а также текстового сообщения, содержание которого может задаваться идущими далее параметрами. Значения параметров должны быть заключены в кавычки.

`UseCustomReply` = `{логический}`	R Отправлять в качестве SMTP-ответа сообщение, заданное в параметре **`ReplyRuleFilter`**, если входящее сообщение отклонено подключаемым модулем **Dr.Web Modifier**. Значение по умолчанию: **`UseCustomReply`** =
`ReplyRuleFilter` = `{текст}`	R Настраиваемое сообщение в SMTP сессии для случаев, когда сообщение отвергается подключаемым модулем **Dr.Web Modifier**. Значение по умолчанию: **`ReplyRuleFilter`** =

Если **`UseCustomReply`** = `No` или соответствующая строка текста не задана, будет выдано стандартное сообщение `"The message has been rejected by the Dr.Web MailD"`.

Примеры

Примеры настройки правил модификации подключаемого модуля **Dr.Web Modifier** (на примере глобальных правил):

1. **Выбрать из письма элементы, отвечающие двум условиям. Если такие элементы имеются, удалить все письмо, иначе – найти все вложенные в письмо исполняемые файлы и удалить их. После этого добавить в конец письма текстовую надпись "проверено!":**

```
GlobalRules = select mime(headers) Content-type "text" \
and mime(body) "typical spam", goto(n) 1, discard, \
select mime(headers) Content-disposition ".exe", \
remove, select message, append_text "проверено!"
```

2. **Удаление писем от выбранных пользователей:**

```
GlobalRules = select mime(headers) \
From "weirdohacker@server.net", if found, \
reject, endif
```

3. **Перенаправление писем:**

```
GlobalRules = select mime.headers \
To "someaddress@my-net.com", \
redirect "anotheraddress@my-net.com"
```

В этом случае оригинал письма будет доставлен по адресу `someaddress@my-net.com`, а его копия будет направлена на `anotheraddress@my-net.com`. Если вы не хотите, чтобы письмо было доставлено оригинальному получателю, то можете использовать правило модификации, приведенное ниже.

4. **Выбор сообщений по указанным признакам, перенаправление найденных сообщений на указанный адрес, удаление исходных сообщений, чтобы они не были доставлены оригинальным получателям:**

```
GlobalRules = select mime.headers Subject "Help", \
if found, select mime.headers To "someaddress@my-net.com", \
if found, redirect "anotheraddress@my-net.com", \
discard, endif, stop, endif
```

5. **Перенаправление писем, приходящих на общий корпоративный ящик, в зависимости от их темы:**

a) письма с темой, соответствующей выражению "техподдержка|проблем[аы]|помощь" - в техподдержку

b) отправка писем с темой, соответствующей выражению "цен[аы]|купить|заказ" - в отдел продаж

c) перенаправление всех прочих писем на иной почтовый ящик

```
GlobalRules = \
select mime.headers Subject "техподдержка|проблем[аы]|помощь", \
if found, select mime.headers To "@company.com", \
if found, redirect "support@company.com", \
endif, pass, endif, \
select mime.headers Subject "цен[аы]|купить|заказ", \
if found, select mime.headers To "@company.com", \
if found, redirect "sell@company.com", \
endif, pass, endif, \
select mime.headers To "@company.com", \
redirect "inbox@company.com", pass
```

6. **Поиск и переименование исполняемых файлов во вложениях:**

```
GlobalRules = select mime.headers \
Content-disposition "filename=.*\\.exe", \
or mime.headers Content-type "name=.*\\.exe", \
replace "\\.ex_" "\\.exe", pass
```

Работа со строковыми значениями

Пожалуйста, обратите внимание, что **во всех правилах** и **параметрах** подключаемых модулей, использующих значение некоторого заголовка или части письма, следует **учитывать кодировку**, в которой встречается искомый текст.

Значения не на латинице, заданные «напрямую», будут совпадать только в том случае, если кодировка искомого текста совпадет с кодировкой, в которой создан конфигурационный файл. Если требуется использовать значения в других кодировках, следует вставить в качестве значения строку, преобразованную в нужную кодировку и закодированную с использованием транспортного кода, например **base64**. Итоговое строковое значение должно быть приведено к выражению вида:

```
=?<source_enc>?<B|Q>?<coded_text>?=
```

где:

- `<source_enc>` – исходная кодировка текста (например, `UTF-8`, `CP1251` и т.п.);
- `<B|Q>` – обозначение использованной транспортной кодировки (`B - base64`, `Q - quoted-printable`);
- `<coded_text>` – закодированное строковое значение.

Обратите внимание, что если полученное значение должно трактоваться как регулярное выражение, то символы, которые играют в регулярных выражениях роль специальных (например, сам символ ?), должны быть дополнительно экранированы двойным обратным слэшем '\\'.

Пример:

Пусть требуется искать письма, содержащие в заголовке слово «тест», написанное на русском языке, с использованием кириллицы в кодировке `CP1251`. В этой кодировке, если ее перевести в транспортную кодировку **base64**, строка будет иметь вид `8uXx8go=`.

Для получения строк в нужной кодировке воспользуйтесь, например, утилитами **iconv** и **base64**. Нижеприведенный пример преобразует строку символов «тест», введенную в кодировке `UTF-8`, в строку символов в кодировке `CP1251`, а затем получается ее представление с использованием **base64**:

```
echo "тест" | iconv -f utf-8 -t cp1251 | base64
```

Примеры использования в подключаемых модулях:

1. Чтобы модуль **Dr.Web Modifier** мог выбрать заголовок, состоящий из слова «тест» в кодировке `CP1251`, необходимо написать следующее условие выборки в правиле модификации:

```
select mime.headers Subject "=\\?cp1251\\?B\\?8uXx8go=\\?="
```

2. Чтобы модуль **Vaderetro** прибавлял к заголовкам писем слово «тест» в кодировке `CP1251`, необходимо написать:

```
SubjectPrefix = "=?cp1251?B?8uXx8go=?="
```

3. Чтобы модуль **Dr.Web HeadersFilter** отклонял письма, заголовок которых содержит слово «тест» в кодировке `CP1251`, необходимо написать:

```
RejectCondition = Subject = "=\\?cp1251\\?B\\?8uXx8go=\\?="
```

Интеграция с почтовыми системами

В данной главе рассматриваются особенности интеграции программного комплекса **Dr.Web для почтовых серверов UNIX** с различными почтовыми системами.

Предусматривается три основных способа интеграции **Dr.Web для почтовых серверов UNIX** с почтовыми системами, изображенные на рисунке ниже (4 способ, указанный на рисунке, является комбинированным).

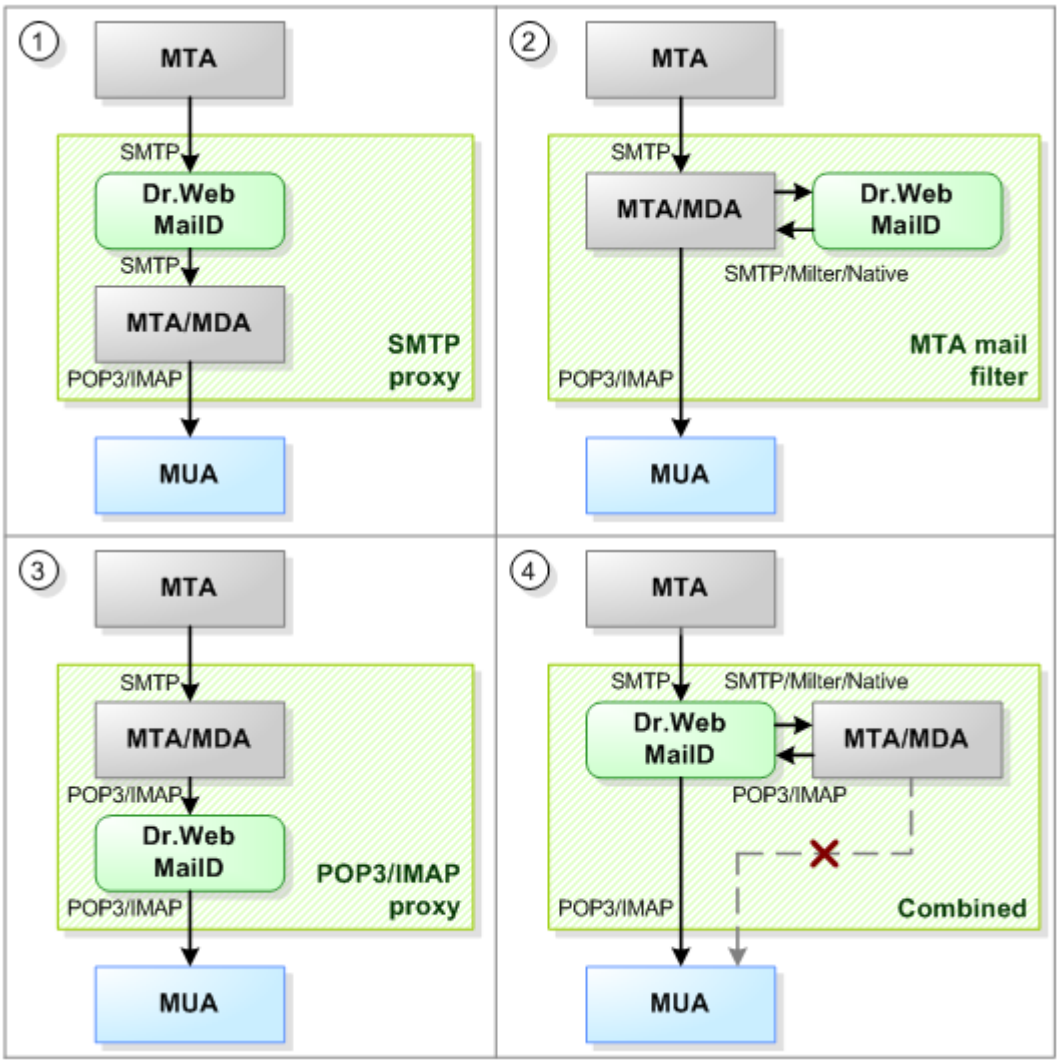


Рис. 20. Способы интеграции Dr.Web для почтовых серверов UNIX с почтовыми системами

Обратите внимание, что во всех способах интеграции, непосредственно с почтовыми системам взаимодействует только один компонент программного комплекса **Dr.Web для почтовых серверов UNIX** – комплексный компонент обработки почты **Dr.Web MailD**.

1. Режим интеграции **SMTP/LMTP-прокси**. Базовый способ интеграции, универсальный и применимый во всех случаях. Подходит для интеграции с любыми MTA, поскольку использует только стандартные почтовые протоколы SMTP/LMTP. В этом случае **Dr.Web MailD** образует посредника между внешним MTA, с которого на сервер поступает корреспонденция, и внутренним MTA/MDA, отвечающим за дальнейшее хранение проверенных писем и взаимодействие с получателями (MUA), или передачу на другие почтовые системы. При необходимости этот режим также можно использовать для организации проверки писем в режиме callback (обратите внимание, что требуются особые дополнительные настройки).

Обратите внимание, что это способ не требует, чтобы защищаемый MTA находился на том же сервере, на котором работает программный комплекс **Dr.Web для почтовых серверов UNIX**. Подробнее о настройках этого режима интеграции и о его особенностях см. в разделе Работа в режиме SMTP proxy.

2. Режим интеграции **MTA Mail filter**. При этом способе интеграции почтовая система сама осуществляет коммуникации как с внешними MTA, с которых на сервер поступает корреспонденция, так и сама отвечает за хранение проверенных писем и взаимодействие с получателями (MUA), или передачу на другие почтовые системы. При этом **Dr.Web MailD** используется этой почтовой системой только как внешнее приложение-фильтр, которому почтовая система передает для проверки принятые письма и получает от него результаты проверки, по результатам которых принимается решение о дальнейших действиях с письмом.

   Для взаимодействия **Dr.Web MailD** и почтовой системы в режиме подключенного фильтра могут использоваться как стандартные протоколы, такие, как `Milter` или `SMTP`, так и нативные, характерные только для этой почтовой системы. Для этого в состав **Dr.Web MailD** включаются специальные модули взаимодействия, реализованные для подключения к некоторым почтовым системам в режиме фильтра. В режиме подключаемого фильтра **Dr.Web MailD** может быть интегрирован со следующими почтовыми системами:

   - **CommuniGate Pro** (см. Описание настройки интеграции);
   - **Sendmail** (см. Описание настройки интеграции);
   - **Postfix** (см. Описание настройки интеграции);
   - **Exim** (см. Описание настройки интеграции);
   - **Qmail** (см. Описание настройки интеграции);
   - **Courier** (см. Описание настройки интеграции);
   - **Zmailer** (см. Описание настройки интеграции);

   В случае если **Dr.Web MailD** может быть интегрирован с некоторой почтовой системой в режиме **MTA Mail filter**, то этот способ является более предпочтительным, нежели универсальный режим **SMTP proxy**, поскольку он требует меньшей нагрузки на вычислительные мощности сервера: в этом случае **Dr.Web для почтовых серверов UNIX** выполняет лишь антивирусные и антиспам-функции, и не отвечает за прием и передачу корреспонденции.

   Обратите внимание, что способ интеграции в режиме **MTA Mail filter** в основном предполагает, что защищаемый MTA находится на том же сервере, на котором работает программный комплекс **Dr.Web для почтовых серверов UNIX**.

3. Режим интеграции **POP3/IMAP proxy**. В этом случае программный комплекс **Dr.Web для почтовых серверов UNIX** используется для проверки писем не на этапе получения их почтовой системой извне, а в момент передачи их MUA конечного получателя по клиентскому почтовому протоколу `IMAP` или `POP3`. Эта схема интеграции может быть реализована только в том случае, когда защищаемая при помощи **Dr.Web для почтовых серверов UNIX** почтовая система является не промежуточной, а конечной, то есть обслуживает запросы от конечных MUA.

   В этом случае специальный компонент **Dr.Web для почтовых серверов UNIX** внедряется в режиме прокси между MUA и MDA, и передает письма, возвращаемые MDA по запросу MUA, на проверку в **Dr.Web MailD**. Этот способ не требует, чтобы MDA или MUA находились на том же сервере, на котором работает программный комплекс **Dr.Web для почтовых серверов UNIX**. Подробнее о настройках этого режима интеграции и о его особенностях см. в разделе Работа с почтовыми клиентами POP3/IMAP.

4. Режим интеграции **Комбинированный**. Этот режим является комбинацией режима 1 или 2 (любого из них) и режима 3. Т.е. приниципиально возможна, хотя и не имеет особого разумного смысла, реализация двойной проверки писем – при их поступлении на защищаемый MTA/MDA извне, и при передаче писем от MDA к MUA. Такой тип настройки может потребоваться, к примеру, если часть писем следует через MTA дальше (и они должны

быть проверены на уровне SMTP), а для части писем этот MTA является конечным MDA, и тогда эти письма могут быть проверены по запросу от конечных MUA при их получении.

Обратите внимание, что в этом случае потребуется тонкая настройка логики комплексного компонента **Dr.Web MailD**, включая задание Правил обработки почты и, возможно, правил маршрутизации исходящих писем (задаются параметром **Router**) в настройках компонента **Sender**.

Для упрощения процесса интеграции в комплект поставки **Dr.Web для почтовых серверов UNIX** входят установочные пакеты и скрипты настройки для разных почтовых систем.

Скрипт `configure_mta.sh` отвечает за настройку взаимодействия между программным комплексом **Dr.Web для почтовых серверов UNIX** и используемой почтовой системой. При запуске он проверит, установлена ли нужная почтовая система. В случае ее отсутствия скрипт завершит свою работу, а в случае обнаружения предложит ответить в интерактивном режиме на ряд вопросов про отдельные настройки конфигурации используемого MTA. В соответствии с полученными ответами скрипт вносит требуемые изменения в параметры соответствующих конфигурационных файлов. Также настройка может быть выполнена вручную: ее особенностям для каждой почтовой системы посвящены следующие разделы руководства.

Скрипт `configure_mta.sh` настраивает MTA следующим образом:

- Для **Exim** реализуется вариант подключения подключения с использованием специального транспорта;
- Для **Postfix** реализуется схема подключения AfterQueue;
- **Zmailer** настраивается для использования в режиме контекстного фильтра на этапе SMTP-соединения.
- Для **Qmail** реализуется схема проксирования.

Таким образом, например, для настройки **Postfix** для работы по протоколу `Milter`, скрипт `configure_mta.sh` запускать не нужно. Вместо этого необходимо произвести настройку согласно действиям, описанным в соответствующем разделе.

Работа в режиме SMTP/LMTP-proxy

Так как **Dr.Web MailD** может работать как прокси-сервер для почтовых протоколов SMTP/LMTP, это позволяет использовать его совместно с большинством почтовых систем. В этом режиме в роли сервера SMTP/LMTP (т.е. компонента **Receiver**, принимающего входящие сообщения) выступает модуль **drweb-receiver**, а в роли клиента SMTP/LMTP (т.е. компонента **Sender**, отправляющего исходящие сообщения) – модуль **drweb-sender**.

Так как модуль **drweb-sender** имеет возможность передавать письма непосредственно локально установленной почтовой системе, то он используется практически во всех схемах интеграции с MTA

В состав **drweb-receiver** входит высокопроизводительный SMTP-сервер, основанный на современных мультиплексорах (`epoll`, `kevent`, `/dev/poll`), являющийся многопоточным, поддерживающий несколько соединений на каждый поток, работу по протоколу `IPv6`, а также следующие SMTP-расширения:

- `PIPELINING` (RFC 2920)
- `8BITMIME` (RFC 1652)
- `ENHANCEDSTATUSCODES` (RFC 3463)
- `SIZE` (RFC 1870)
- `AUTH` (RFC 4954)

При работе **Dr.Web MailD** в режиме **SMTP/LMTP-прокси** в системе должны быть запущены следующие модули (регулируется в mmc-файле **Dr.Web Monitor**):

- **drweb-notifier**
- **drweb-sender**
- **drweb-receiver**
- **drweb-maild**

Все параметры настройки модулей **drweb-receiver** и **drweb-sender** сосредоточены в секциях [Receiver] и [Sender] конфигурационного файла **Dr.Web MailD** и описаны в главах Секция [Receiver] и Секция [Sender] данного руководства.

Дополнительно следует обеспечить, чтобы компонент **Dr.Web Monitor** запускался с правами root (значения параметров **User** и **Group** в секции [Monitor] конфигурационного файла monitor.conf должны быть установлены в значение root).

При работе напрямую с внешними MTA через Internet полезны специальные антиспам-технологии, встроенные в **drweb-receiver** и позволяющие фильтровать письма непосредственно на этапе SMTP-сессии:

- Технология контроля ограничений SMTP-сессии.
- Технологии оценивания репутации соединения (отправителя) **Reputation IP Filter** и **Unified Score**.

Также, пожалуйста, обратите внимание на рекомендации по использованию синхронного и асинхронного режимов обработки входящих сообщений.

Callback-режим SMTP

Dr.Web MailD может работать не только в качестве прокси-сервера почтовых протоколов SMTP/LMTP, но и использоваться в качестве сервиса для проверки почты в режиме «callback». В этом режиме MTA, принимающие почту, пересылают ее для проверки на **Dr.Web MailD**, как на внешний фильтр, используя протокол SMTP/LMTP. После проверки письма **Dr.Web MailD** должен вернуть его, используя протокол SMTP/LMTP, на тот же сервер, с которого он его получил.

Схема реализации режима «callback» приведена на рисунке ниже.

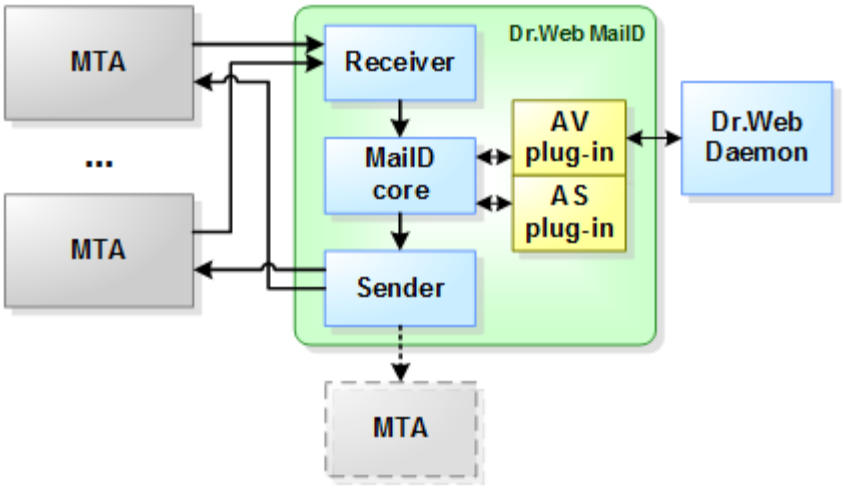


Рис. 21. Интеграция Dr.Web MailD с почтовыми системами в режиме "callback"

По составу запущенных модулей и основным настройкам этот режим не отличается от обычного режима интеграции SMTP/LMTP-прокси.

Особые настройки Callback-режима

Особенность настроек заключается в том, что **Sender** должен обеспечить возврат письма на тот же хост, с которого **Receiver** его получил (но на другой порт). Для этого требуется добавить в настройки Правила, изменяющие правила отправки писем. Правила указываются в секции `[Rules]` основного конфигурационного файла **Dr.Web MailD**. Для указания адреса возврата проверенных писем в Правилах следует использовать параметр **`SenderAddress`**.

Типовое Правило в этом случае имеет следующий вид:

```
<CONDITION> cont SenderAddress = inet:<pont-num>@CLIENT-IP
```

где `<CONDITION>` – условие срабатывания Правила (в простейшем случае, для безусловного срабатывания Правила, достаточно указать `true`), а `<port-num>` – номер порта на сервере, на котором MTA ожидает возвращения письма (например, 10025). `CLIENT-IP` – это специальный макрос, используемый для подстановки в адрес для отправки IP-адреса, запомненного **Receiver** при приеме письма.

Обратите внимание, что все действия с письмами как для **Receiver**, так и для подключаемых модулей, следует настроить таким образом, чтобы избежать отказов в приеме писем для проверки (не использовать действия `reject`, `discard`, `tempfail`), или настроить MTA таким образом, чтобы они правильно реагировали на отказ «callback»-фильтра при приеме письма.

Кроме того, если в конфигурации **Dr.Web MailD** предусмотрено разрешение на генерацию DSN (например, используется асинхронный режим и параметр **`SkipDSNOnBlock`** установлен в `No`) и уведомлений, то в этом случае в `<CONDITION>` Правила должно стоять условие, не позволяющее отправлять служебные письма на тот адрес, на котором MTA ожидает проверенные письма. Для этого в `<CONDITION>` следует использовать проверки, использующие параметр **`from (sender)`**. Все DSN всегда отправляются с пустым `From`, а все уведомления отправляются с полем `From`, содержащим адрес, взятый из параметра **`FilterMail`** (секция `[Notifier]`).

Пример:

```
!(from:"" || from:"root@localhost") cont SenderAddress = inet:10025@CLIENT-IP
```

Данное Правило отправит исходящее письмо на порт 10025 хоста, с которого оно получено, если в его поле `From` не пустое и не содержит адрес `root@localhost`.

Поскольку параметр Правил **`SenderAddress`** только динамически переопределяет значение параметра **`Address`**, указанное в секции `[Sender]`, то, если Правило не сработает, письмо будет отправлено на MTA, адрес которого указан в настройках **Sender**. Рекомендуется указать там адрес действующего почтового сервера, чтобы оперативно получать письма, информирующие о проблемах в обработке писем (на рисунке этот MTA указан пунктиром).

Работа с почтовыми клиентами POP3/IMAP

Программный комплекс **Dr.Web для почтовых серверов UNIX** может быть использован для проверки писем не на этапе получения их почтовой системой извне, а в момент передачи их MUA конечного получателя по клиентскому почтовому протоколу `IMAP` или `POP3`. Эта схема интеграции может быть реализована только в том случае, когда защищаемая при помощи **Dr.Web для почтовых серверов UNIX** почтовая система является не промежуточной, а конечной, то есть обслуживает запросы от конечных MUA.

Для реализации этого механизма в состав **Dr.Web для почтовых серверов UNIX** входят два специальных прокси-компонента:

- **POP3 filter** – используется для перехвата сообщений протокола `POP3` при взаимодействии

MUA и MDA. Реализован в виде модуля **drweb-pop3**;

- **IMAP filter** – используется для перехвата сообщений протокола IMAP при взаимодействии MUA и MDA. Реализован в виде модуля **drweb-imap**.

Схема подключения комплексного компонента обработки почты **Dr.Web MailD** при работе с почтовыми клиентами показана на рисунке ниже.

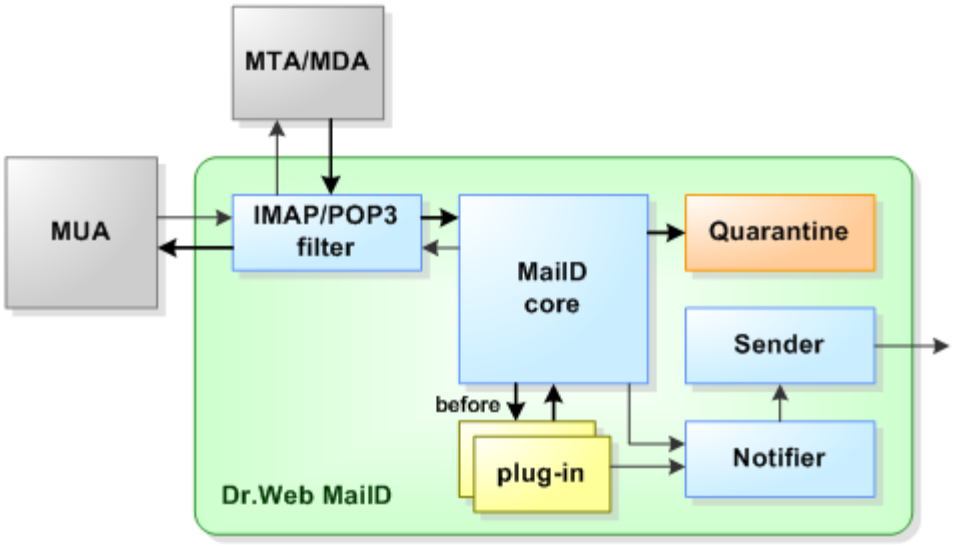


Рис. 22. Работа с почтовыми клиентами

Общие принципы работы:

1. Компонент-фильтр пользовательского протокола (**POP3 filter** или **IMAP filter**, в зависимости от используемого протокола) настраивается в качестве прокси таким образом, чтобы принимать сообщения соответствующего протокола, передаваемые между MUA и MDA.
2. Принятые от MUA сообщения передаются компонентом на целевой MDA (он может располагаться как локально, так и удаленно относительно **Dr.Web MailD**).
3. Письмо, поступающее из MDA в MUA, при прохождении через компонент-фильтр передается им для проверки в **MailD core** (используется интерфейс, используемый компонентом **Receiver**).
4. Компонент **MailD core** выполняет проверку письма (с использованием Правил обработки писем и настроенных подключаемых модулей).
5. Если от **MailD core** поступает положительный результат проверки, то письмо далее передается MUA, в противном случае MUA вместо запрошенного письма будет передано письмо с отчетом об угрозе, обнаруженной в проверенном письме. Это письмо-отчет формируется компонентом **Notifier**.
6. Если в настройках **MailD core** включена настройка отправки отчетов об обнаруженных угрозах, то соответствующие отчеты, формируемые компонентом **Notifier**, отправляются получателям через компонент **Sender** (он отправляет их для окончательной доставки на MTA, указанный в его настройках).

Обратите внимание, что в случае использования **Dr.Web MailD** для проверки писем через клиентские протоколы есть **ограничения по настройке**:

- Все используемые подключаемые модули должны быть помещены только в очередь **BeforeQueueFilters**, т.е. проверка писем в асинхронном режиме с сохранением их в хранилище не допускается (это связано с особенностью работы протоколов POP3 и IMAP).
- Нельзя использовать в Правилах и настройках подключаемых модулей действие redirect, поскольку письмо, следующее от MDA к MUA пользователя, не может быть перенаправлено на

другой адрес.

Характеристики компонентов фильтрации:

1. Компонент фильтрации IMAP filter

Поддерживает работу с серверами IMAP (включая функцию кэширования). Является прокси-сервером между **MailD core** **(drweb-maild)** и сервером IMAP (MDA). Фильтрует письма, отправляемые сервером пользователю. IMAP-сервер MDA может находиться как на локальном, так и на удаленном компьютере.

Функции компонента реализует модуль **drweb-imap**. Настройки модуля находятся в секции `[IMAP]` основного конфигурационного файла **Dr.Web MailD.**

Компонент фильтрации **IMAP filter** кэширует основные заголовки сообщений в оперативной памяти для ускорения доступа к ним. Теоретически возможно исчерпать доступную память или замедлить работу фильтра, пропуская через него большое количество специальным образом сформированных писем, содержащих большое количество заголовков.

Для борьбы с этим **IMAP filter** имеет настройку **MaxCachedHeadersPerMail**, контролирующую наибольший суммарный размер кэшированных заголовков. Обратите внимание, что если это значение будет слишком мало, то у пользователей могут прекратить корректно отображаться названия и типы MIME-вложений.

По умолчанию фильтр выключен. Чтобы активизировать его запуск, необходимо в mmc-файле компонента **Dr.Web Moritor** для управления компонентами программного комплекса **Dr.Web для почтовых серверов UNIX** (`maild_<MTA>.mmc`) раскомментировать строку:

```
drweb-imap  local:/var/drweb/ipc/.agent  15  30  MAIL  drweb:drweb
```

2. Компонент фильтрации POP3 filter

Поддерживает работу с серверами POP3. Является прокси-сервером между **MailD core** **(drweb-maild)** и сервером POP3 (MDA). Фильтрует письма, отправляемые сервером пользователю. POP3-сервер MDA может находиться как на локальном, так и на удаленном компьютере.

Функции компонента реализует модуль **drweb-pop3**. Настройки модуля **drweb-pop3** находятся в секции `[POP3]` основного конфигурационного файла **Dr.Web MailD.**

При каждом подключении **POP3 filter** выделяет имя пользователя из POP3-команды `USER username` и сохраняет его на все время сессии. При успешной аутентификации на сервере фильтр разрешает передачу писем от сервера к клиенту. При этом все команды и данные передаются в неизменном виде, за исключением ответа сервера на команду `RETR` (извлечение письма).

Ответ от MDA на эту команду передается **MailD core** для обработки, а MUA возвращается обработанный ответ.

По умолчанию фильтр выключен. Чтобы активизировать его запуск, необходимо в mmc-файле компонента **Dr.Web Moritor** для управления компонентами программного комплекса **Dr.Web для почтовых серверов UNIX** (`maild_<MTA>.mmc`) раскомментировать строку:

```
drweb-pop3  local:/var/drweb/ipc/.agent  15  30  MAIL  drweb:drweb
```

При работе **Dr.Web MailD** в режиме **POP3/IMAP proxy** в системе должны быть запущены следующие модули (регулируется в mmc-файле **Dr.Web Monitor**):

- **drweb-notifier**
- **drweb-sender**
- **drweb-maild**
- **drweb-pop3** или **drweb-imap** (в зависимости от перехватываемого пользовательского

протокола)

Обратите внимание, что часть имени файла <MTA> зависит от названия MTA, с которым сопряжен **Dr.Web для почтовых серверов UNIX**

Интеграция с почтовой системой CommuniGate Pro

Настройка CommuniGate Pro

Чтобы **CommuniGate Pro** (далее **CGP**) мог передавать и принимать письма от **Dr.Web MailD**, необходимо выполнить следующие действия:

- Подключиться к управляющему Web-интерфейсу **CGP** (он должен быть подключен как модуль в утилите удаленного администрирования **WebAdmin**);
- По вкладкам перейти на страницу `Settings -> General -> Helpers`;
- В раздел **Content Filtering** добавить новый внешний фильтр, указав для него следующие параметры (фильтр добавляется в последней, пустой, строке перечня имеющихся фильтров):

  `Enabled` (выбрать в выпадающем списке)
  Указать имя фильтра в текстовое поле: `DrWeb Maild`
  `Log Level: Problems`
  `Program Path: %bin_dir/drweb-cgp-receiver`
  `Time-out: 2 min` (выбрать в выпадающем списке)
  `Auto-Restart: 15 sec` (выбрать в выпадающем списке)

- Проверить, достаточен ли уровень привилегий, с которыми исполняется **CGP**, для запуска модуля **drweb-cgp-receiver**;
- По вкладкам перейти на страницу `Settings -> Mail -> Rules`;
- Добавить новое правило проверки. Для этого:
  - Выбрать и ввести имя правила (например, **drweb-filter**) в текстовое поле в верхней части списка правил, и нажать кнопку **Add Rule**;
  - Нажать в строке созданного правила ссылку **Edit** и на появившейся странице настройки правила установить в выпадающем списке **Action** значение `ExternalFilter`;
  - В поле **Parameter** ввести имя фильтра, созданного на предыдущем шаге (на вкладке `Settings -> General -> Helpers`). В данном примере это "`DrWeb Maild`".

Рекомендуется добавить к созданному правилу дополнительные настройки:

1. Чтобы избежать многократной проверки писем, пришедших из `GROUP`, `LIST` или `RULES` (http://www.communigate.com/CommuniGatePro/Transfer.html), добавьте к созданному правилу также настройку:

   ```
   "Submit Address", "not in", "GROUP*,LIST*,RULES*"
   ```

   Для этого выберите тип проверяемого поля и операцию сравнения из выпадающих списков и введите проверяемое значение в текстовое поле.

2. При загрузке писем через `PIPE` у них теряется флаг `authenticated`. Следовательно, если в **Dr.Web MailD** имеются подключаемые модули, указанные в очереди **AfterQueueFilters**, то имеет смысл также добавить к правилу следующую настройку:

   ```
   Any Recipient not in alldomains@<main_domain>,all@*
   ```

где <main_domain> – главный домен сервера **CGP**.

За инструкциями по более детальной настройке (в частности, для управления возможностью включать и выключать фильтрацию для каждого пользователя в отдельности) обратитесь к документации, поставляемой в комплекте с почтовой системой **CGP**.

Настройка Dr.Web MailD

При работе с **CGP** в **Dr.Web MailD** в качестве компонента **Sender** выступает модуль **drweb-cgp-sender**, запущенный с привилегиями группы mail для того, чтобы иметь возможность писать в каталог cgp. А функции компонента **Receiver** выполняет модуль **drweb-cgp-receiver**, запускаемый самой почтовой системой **CGP** с правами root.

В такой конфигурации для нормальной работы программного комплекса необходимо либо явно указать пользователя, от имени которого запускаются остальные модули, указав это имя в значении параметра **ChownToUser** секции настроек [CgpReceiver] конфигурационного файла **Dr.Web MailD**, либо установить для этого параметра пустое значение, и запускать весь программный комплекс с правами root.

Взаимодействие **Dr.Web MailD** с почтовой системой **CGP** имеет следующие особенности: производится локально, через механизм PIPE, **Dr.Web MailD** работает как контент-фильтр, а потому не имеет возможности модифицировать заголовки писем. В связи с этим, когда **Dr.Web MailD** требуется изменить заголовки письма, например, внести пометку о спаме (обычно выглядит как добавка строки «[SPAM]» к теме письма), применяется следующий способ изменения писем: **Dr.Web MailD** шлет почтовой системе **CGP** уведомление с требованием отклонить исходное письмо, а одновременно с этим во входную очередь почтовой системы добавляется письмо, подвергнутое модификации. Так как в этом случае письмо снова попадет на проверку в **Dr.Web MailD**, используется следующий механизм защиты от зацикливания:

- Модуль **drweb-cgp-receiver** пропускает без проверки все письма, поступившие в почтовую систему через механизм PIPE. Так как **drweb-cgp-sender** загружает новые письма в **CGP** через PIPE, то этим исключается повторная проверка писем, помещенных **Dr.Web MailD** в очередь **CGP**. Но это приводит к тому, что пропускаются без проверки все письма, помещаемые во входящую очередь **CGP** любой программой, работающей через PIPE.
- Во избежание пропуска посторонних писем, помещенных в очередь **CGP** через PIPE, в измененные письма, отправляемые в **CGP,** рекомендуется добавлять специальный заголовок. Использование этого заголовка задается параметрами **UseSecureHash** и **SecureHash** секции настроек [CgpSender] конфигурационного файла **Dr.Web MailD**. Если параметр **UseSecureHash** имеет значение Yes, то такой заголовок, с именем **X-DrWeb-Hash**, будет добавляться к письму при его отправке во входную очередь **CGP,** а параметр **SecureHash** задает значение этого заголовка.
- В этом случае письма, полученные от почтовой системы, будут отправляться на доставку, минуя проверку, не только если они попали в очередь **CGP** через PIPE, но и в них содержится заголовок **X-DrWeb-Hash** со значением, указанным в параметре **SecureHash**. Модуль **drweb-cgp-receiver** будет передавать эти письма для конечной доставки получателю, предварительно очистив значение заголовка (заполнив его пробелами). Письма, не имеющие такого заголовка, отправятся на проверку.

Обратите внимание, что, поскольку **Dr.Web MailD** в режиме контент-фильтра не имеет возможности удалять заголовки, то если письмо проходило через цикл повторного приема, конечные получатели получат его со включенным, но пустым (заполненным пробелами) заголовком **X-DrWeb-Hash**, что не окажет никакого влияния на отображение содержимого письма почтовым клиентом.

Обратите внимание, что поскольку этот параметр используется не только **Sender**, но и **Receiver**, то после изменения значения этого параметра не достаточно отправить сигнал `HUP` компоненту **Dr.Web Monitor** (это заставит перечитать конфигурацию компонент **Sender**). Необходимо также перезапустить почтовую систему **CGP**, поскольку именно она запускает компонент **Receiver**, и только ее перезапуск заставит компонент **Receiver** перечитать измененное значение этого параметра конфигурации.

Подробности настройки параметров **Dr.Web MailD** для работы с **CGP**, представленных в секциях `[CgpReceiver]` и `[CgpSender]` конфигурационного файла **Dr.Web MailD**, рассмотрены в главах CgpReceiver и CgpSender соответственно.

При работе **Dr.Web MailD** с **CGP** в системе должны быть запущены следующие модули (регулируется в mmc-файле **Dr.Web Monitor**):

- **drweb-notifier**
- **drweb-cgp-sender**
- **drweb-maild**

Дополнительно следует обеспечить, чтобы компонент **Dr.Web Monitor** запускался с правами `root` (значения параметров **User** и **Group** в секции `[Monitor]` конфигурационного файла `monitor.conf` должны быть установлены в значение `root`).

Принцип работы

Dr.Web MailD работает с почтовой системой **CGP** следующим образом:

1. Письмо приходит в **CGP**.
2. После проверки своих настроек, **CGP** при необходимости отправляет сообщение на проверку в `helper`, в роли которого выступает компонент **drweb-cgp-receiver**.
3. При получении письма компонент **drweb-cgp-receiver** ищет в нем заголовок `SecureHash`:
   - если заголовок найден, **drweb-cgp-receiver** возвращает **CGP** ответ `OK` и письмо передается для дальнейшей обработки в **CGP**;
   - в противном случае сообщение передается для проверки в **drweb-maild**;
4. **drweb-maild** применяет к письму подключаемые модули, которые могут изменить его (например, добавить заголовки).
   - если вирусы не обнаружены и письмо не было изменено, в **CGP** передается ответ `OK`;
   - если в процессе обработки письмо было изменено, то **CGP** передается ответ `DISCARD` и передача письма осуществляется средствами **drweb-maild**. Это связано с тем, что в протоколе `helper` нельзя вернуть измененное письмо.
5. Письмо передается в компонент **Sender**, и, после добавления заголовка `SecureHash` (при значении параметра **UseSecureHash**=`yes`), перемещается в каталог для отправляемых сообщений `/var/CommuniGate/Submitted/`, периодически проверяемую **CGP**.

Значение параметра **SubmitDir** конфигурационного файла **Dr.Web MailD** должно быть равно `/var/CommuniGate/Submitted`. В противном случае письма, проверенные **Dr.Web MailD,** не будут доходить до получателей.

6. После проверки каталог `/var/CommuniGate/Submitted/` и получения письма, **CGP** переходит к пункту 2:
   - В случае корректных настроек, письмо не будет проверяться повторно.
   - В случае неточностей в настройках, письмо будет передано обратно в **CGP** после

проверки значения заголовка `SecureHash`.

- В случае некорректной настройки возможно зацикливание проверки письма.

Известные проблемы

В системах семейства **Linux** после изменения и обновления командной строки через настройку `Helpers` предыдущий процесс фильтра остается в состоянии `zombie` до перезагрузки **CGP**.

Описание:

В процессе запуска **drweb-cgp-receiver** выводятся сообщения вида:

```
/usr/libexec/ld-elf.so.1: Shared object "libstdc++.so.6"
not found, required by "libboost_thread.so"
```

Решение:

Система не может найти необходимые библиотеки, находящиеся в каталоге `%bin_dir/lib/`. Необходимо скопировать библиотеки (или сделать на них линки) **libstdc++.so.6** и **libgcc_s.so.1** из `%bin_dir/lib/` в каталог системных библиотек.

Интеграция с почтовой системой Sendmail

Для совместной работы программного комплекса **Dr.Web для почтовых серверов UNIX** и почтовой системы **Sendmail**, последней требуется поддержка `Milter API`. Если в установленной у вас почтовой системе **Sendmail** поддержка данного API отключена, необходимо пересобрать **Sendmail** с поддержкой библиотеки `Milter API`. За дополнительной информацией по этой операции обратитесь к соответствующей документации по сборке **Sendmail**.

Примечание: Чтобы проверить, собран ли имеющийся у вас **Sendmail** с поддержкой `Milter API`, выполните следующую команду:

```
# sendmail -bt -d0 < /dev/null
```

Если в полученном выводе на консоль фигурирует строчка "`milter`", значит, имеющийся у вас **Sendmail** собран с поддержкой `Milter API`.

Dr.Web MailD полностью совместим с **Sendmail** 8.12.3 и выше. При работе с более ранними версиями могут возникать проблемы (см. раздел Известные проблемы). Подробные инструкции для подключения в настоящей документации актуальны для **Sendmail** версии 8.14.0 и выше.

Взаимодействие между почтовой системой **Sendmail** и **Dr.Web MailD** осуществляется через `Milter API` (в качестве компонента **Receiver** используется модуль **drweb-milter**) и происходит следующим образом:

- Через транспортное соединение, определяемое со стороны модуля **drweb-milter** транспортным адресом `__ADDRESS__`, системе **Sendmail** передаются внутренние команды `Milter API` и почтовое сообщение. При этом сообщение передается не сразу целиком, а по частям, в зависимости от фазы почтовой сессии (`helo`, `mail from:`, `rcpt to:` и т.д.), поэтому оно сохраняется модулем **drweb-milter** во временных файлах. Посредством `Milter API` модуль **drweb-milter** передает системе **Sendmail** указания, что делать с данным сообщением.

  `Milter API` является многопоточной библиотекой, т.е. одновременно в процессе может находиться несколько почтовых сессий. В данной схеме взаимодействия **Sendmail** является клиентом, а **drweb-milter** – сервером, поэтому в конфигурационном файле почтовой системы `sendmail.cf` указывается адрес модуля **drweb-milter**, а система **Sendmail** для этого соединения выбирает подходящий клиентский адрес;

- Через другое транспортное соединение модуль **drweb-milter** передает модулю **drweb-maild** команды и ждет ответа.

В приведенной схеме модуль **drweb-milter** является простым посредником (или преобразователем) между интерфейсом почтовой системы **Sendmail** и модулем **drweb-maild**.

Пожалуйста, обратите внимание на особенности работы через `Milter` в синхронном и асинхронном режимах.

Sendmail и модуль **drweb-milter** могут быть запущены на разных компьютерах, но модули **drweb-milter** и **drweb-maild** должны быть запущены локально.

Настройка почтовой системы Sendmail

Для настройки взаимодействия между почтовой системой **Sendmail** и **Dr.Web MailD** необходимо внести изменения в конфигурационные файлы `sendmail.mc` и `sendmail.cf`.

Если пересобирать конфигурационный файл `sendmail.cf` нежелательно, можно просто вставить в него или добавить (если соответствующие определения в файле уже есть) следующие строки:

Для версий 8.14.0 и выше:

```
############################
# Input mail filters
############################
O InputMailFilters=drweb-milter
O Milter.LogLevel=6
############################
#   Xfilters
############################
Xdrweb-milter, S=__ADDRESS__,
F=T,T=C:1m;S:5m;R:5m;E:1h
```

Чтобы иметь возможность проверять сообщения, отправленные локально (через вызов `mail` или `sendmail`), необходимо продублировать все изменения, сделанные в файле `sendmail.cf`, в файлы `submit.cf` и `submit.mc`.

Пожалуйста, обратите внимание, что по умолчанию файлы `submit.cf` и `submit.mc` защищены от записи, поэтому прежде, чем добавлять в них строки, необходимо изменить права доступа к этим файлам, разрешив запись в них. Кроме того, необходимо добавить к параметру o **PrivacyOptions** значение `nobodyreturn`.

Пример:

```
# privacy flags
O PrivacyOptions=goaway,noetrn,nobodyreturn
```

Или в `{sendmail_src}/cf/cf/feature/msp.m4`:

```
define(`confPRIVACY_FLAGS'
`goaway,noetrn,nobodyreturn,restrictqrun')
```

Для случая, когда фильтр недоступен, установите следующие флаги (`F=`):

- `R` – отказать в доставке;
- `T` – временно отложить доставку.

Если ни `F=R`, ни `F=T` не указаны, то сообщение пропускается без проверки.

Также можно добавить в `sendmail.mc` следующие строки:

Для версий 8.14.0 и выше:

```
INPUT_MAIL_FILTER(`drweb-milter', `S=__ADDRESS__,
F=T, T=C:1m;S:5m;R:5m;E:1h')
define(`confMILTER_LOG_LEVEL',`6')
```

Величину времени ожидания лучше выбирать в соответствии с величинами времени ожидания **Sendmail**:

```
O Timeout.datablock=XX
```

(по умолчанию эта величина равна 1 часу, `XX=>1h`).

После внесения изменений файл `sendmail.cf` необходимо пересобрать.

`__ADDRESS__` – строка, задающая адрес транспорта для подключения модуля **drweb-milter**. Она имеет формат и значение, идентичные формату и значению параметра **Address** секции `[Milter]` конфигурационного файла **Dr.Web MailD**.

Для TCP-сокетов адрес задается в формате:

```
inet:__PORT__@__HOST__
```

где `__PORT__` и `__HOST__` должны иметь конкретные значения (например, `inet:3001@localhost`).

Для UNIX-сокетов адрес задается в формате:

```
local:__SOCKPATH__
```

где `__SOCKPATH__` должен указывать путь, доступный с теми правами, с которыми будет запущен фильтр (наример, `local:/var/run/drweb-milter.sock`).

Тонкости настройки фильтра можно найти в документации на **Sendmail**. После установки всех необходимых параметров следует перезапустить **Sendmail**.

Чтобы модуль **drweb-maild** мог при выводе сообщений в лог указывать идентификаторы почтовых сообщений **Sendmail** (`sendmails message ID`). Также для передачи модулю **drweb-maild** информацию об успешной авторизации, в файле `sendmail.cf` должна присутствовать следующая строка:

```
O Milter.macros.envfrom=i,{auth_type}, ...
```

(многоточием обозначены остальные параметры, значение которых не важно).

Чтобы **Dr.Web MailD** мог определить IP-адрес и имя хоста, от которого принято сообщение, а также мог передавать модулю **drweb-maild** адрес интерфейса, на который было принято письмо, в файле `sendmail.cf` должна присутствовать следующая строка:

```
O Milter.macros.connect=_,{if_addr}, ...
```

(многоточием обозначены остальные параметры, значение которых не важно).

Для подавления вывода в **syslog** сообщений вида:

```
X-Authentication-Warning: some.domain.com: drweb set sender to DrWeb-
DAEMON@some.domain.com using -f
```

необходимо внести того пользователя, от имени которого работает **drweb-milter** (по умолчанию - drweb) в список trusted-users в файле submit.cf.

Это можно сделать, добавив пользователя в список непосредственно в файлах submit.cf и sendmail.cf:

```
####################
#  Trusted users    #
####################
Tdrweb
```

Либо добавив в файл submit.mc строку:

```
define(`confTRUSTED_USERS', `drweb')
```

Настройка Dr.Web MailD

Все параметры работы модуля **drweb-milter** и компонента **Sender** сосредоточены в секциях [Sender] и [Milter] конфигурационного файла **Dr.Web MailD** и описаны в главах Секция [Sender] и Секция [Milter] соответственно.

В обязательном порядке должно быть задано значение параметра **SecureHash** секции [Sender] конфигурационного файла **Dr.Web MailD** (значением параметра может быть произвольная строка, рекомендуемая длина строки – не менее 10 символов), и должно быть установлено значение Yes для параметра **UseSecureHash** из этой же секции. Эти параметры регулируют добавление в обрабатываемые письма специального заголовка, предотвращающего зацикливание письма при его проверке (поскольку оно может быть еще раз помещено во входную очередь почтовой системы в случае модификации в ходе проверки).

При работе **Dr.Web MailD** с почтовой системой **Sendmail** в системе должны быть запущены следующие модули (регулируется в mmc-файле **Dr.Web Monitor**):

- **drweb-notifier**
- **drweb-sender**
- **drweb-maild**
- **drweb-milter**

Известные проблемы

Описание:

При использовании UNIX-сокета для коммуникации между фильтром и почтовой системой **Sendmail** библиотека поддержки Milter API (поставляемая вместе с **Sendmail**) не удаляла (до версии 8.12.2) используемый под сокет файл.

Решение:

Для версий 8.12.x следует использовать исправление **listener-8.12.0-1.patch**. Для версий 8.11 и выше данный файл нужно удалять вручную или из скрипта, осуществляющего управление фильтром. Эта проблема решена в **Sendmail** 8.12.2.

Описание:

При использовании локального сканирования и демо-ключа, после прохождения фильтра значение размера сообщения, передаваемое следующему серверу, увеличивается вдвое (само сообщение либо остается неизменным, либо к нему дописывается небольшое сообщение, т.н. "баннер").

Решение:

Данная проблема решена в **Sendmail** 8.12.3 и более поздних версиях.

Описание:

При работе фильтра на загруженных машинах в почтовом логе можно наблюдать записи следующего вида:

```
... Milter (drweb-milter): select(read): interrupted system call
```

Решение:

Эта проблема решена в **Sendmail** 8.12.3 и более поздних версиях.

Описание:

При работе фильтра на загруженных машинах в почтовом логе можно наблюдать записи следующего вида:

```
... Milter (drweb-milter): select(read): timeout before data write
... Milter (drweb-milter): to error state
```

Решение:

Проблема связана с тем, что **Sendmail** не может установить соединение с фильтром за заданное время ожидания (таймаут). В версиях 8.11 и выше он равен 5 секундам и не может быть изменен, в версиях 8.12 и выше это время ожидание изменяется в описании фильтра (значение C):

```
Xdrweb-milter, S=__ADDRESS__, F=T, T=C:1m;S:5m;R:5m;E:1h
```

Интеграция с почтовой системой Postfix

Принцип работы

Dr.Web MailD может быть подключен к **Postfix** тремя различными способами:

- в режиме **after-queue** (http://www.postfix.org/FILTER_README.html#advanced_filter);
- в режиме **before-queue** (http://www.postfix.org/SMTPD_PROXY_README.html);
- с использованием протокола `Milter` (http://www.postfix.org/MILTER_README.html).

Для работы в режиме `Milter` требуется версия **Postfix** не ниже 2.3.3.

Работа в режимах before-queue и after-queue

Dr.Web MailD работает с почтовой системой **Postfix** в режиме **after-queue** следующим образом:

1. Модуль сервера SMTP/LMTP **drweb-receiver** (компонент **Receiver**) получает новое письмо от SMTP-модуля системы **Postfix,** после чего передает его модулю **drweb-maild** (компонент **MailD core**) для анализа.
2. По результатам анализа это письмо либо высылается почтовой системе (возможно, модифицированным), либо блокируется (в таком случае почтовой системе могут отсылаться дополнительные уведомления).
3. Пересылка писем почтовой системе **Postfix** осуществляется через клиент SMTP/LMTP **drweb-sender** (компонент **Sender**), который передает почтовые сообщения демону **smptd**.

За более подробной информацией по настройке фильтров в **Postfix** обращайтесь к документации **Postfix**, например, по адресу http://www.postfix.org/FILTER_README.html.

Dr.Web MailD может работать с сервером **Postfix** также и в режиме **before-queue** (но этот режим не рекомендуется использовать при больших нагрузках на систему). Подробности по настройке этого режима можно найти, например, по адресу http://www.postfix.org/SMTPD_PROXY_README.html.

Работа по протоколу Milter

Взаимодействие с **Postfix** по протоколу `Milter` происходит следующим образом:

- Через транспортное соединение, определяемое со стороны модуля **drweb-milter** (который работает как компонент **Receiver**) транспортным адресом, системе **Postfix** передаются внутренние команды `Milter API` и сообщение. При этом сообщение передается по частям, в зависимости от фазы почтовой сессии (`helo`, `mail from:`, `rcpt to:` и т.д.). Части сообщения сохраняются модулем **drweb-milter** во временных файлах. Посредством `Milter API` модуль **drweb-milter** передает системе **Postfix** указания о действиях, которые необходимо совершить над данным сообщением.

  `Milter API` является многопоточной библиотекой, что позволяет одновременно обрабатывать несколько почтовых сессий. В описываемом режиме взаимодействия **Postfix** является клиентом, а **drweb-milter** – сервером, поэтому в конфигурационном файле системы **Postfix** `main.cf` указывается адрес модуля **drweb-milter**, а система **Postfix** выбирает подходящий клиентский адрес для этого соединения;
- Через другое транспортное соединение модуль **drweb-milter** передает модулю **drweb-maild** команды и ждет ответа.

В приведенной схеме модуль **drweb-milter** является простым посредником (или преобразователем) между интерфейсом почтовой системы **Postfix** и модулем **drweb-maild**.

Пожалуйста, обратите внимание на особенности работы через `Milter` в синхронном и асинхронном режимах.

Postfix и модуль **drweb-milter** могут быть запущены на разных компьютерах, но модули **drweb-milter** и **drweb-maild** должны быть запущены локально.

Настройка Postfix

Для работы в режиме after-queue

Для настройки работы в режиме **after-queue** в конфигурационный файл системы **Postfix** `main.cf` необходимо добавить следующие строки:

```
content_filter=scan:<_ADDR_REC_>
receive_override_options=no_address_mappings
```

где <_ADDR_REC_> – адрес слушающего модуля **drweb-receiver** (параметр **Address** секции [Receiver] конфигурационного файла **Dr.Web MailD**). Например, 127.0.0.1:8025.

В конфигурационный файл системы **Postfix** master.cf необходимо добавить следующие строки:

```
scan unix - - n - <NN> smtp
  -o smtp_send_xforward_command=yes
<_ADDR_SEN_> inet n - n - <NN> smtpd
  -o content_filter=
  -o receive_override_options=no_unknown_recipient_checks,no_header_body_checks
  -o smtpd_helo_restrictions=
  -o smtpd_client_restrictions=
  -o smtpd_sender_restrictions=
  -o smtpd_recipient_restrictions=permit_mynetworks,reject
  -o mynetworks=127.0.0.0/8
  -o smtpd_authorized_xforward_hosts=127.0.0.0/8
```

где <_ADDR_SEN_> – адрес, к которому подключается модуль **drweb-sender** для отправления писем (параметр **Address** секции [Sender] конфигурационного файла **Dr.Web MailD**). Например, 127.0.0.1:8026.

Желательно, чтобы число <NN> (максимальное количество процессов, исполняемых сервером **Postfix**) совпадало с числом потоков в пулах модулей **drweb-receiver** и **drweb-sender** (параметры **PoolOptions** секции [Receiver] и **OutPoolOptions** секции [Sender] конфигурационного файла **Dr.Web MailD**). Чтобы убрать это ограничение, укажите знак минус ("-"), вместо числа <NN>.

При установке **Dr.Web для почтовых серверов UNIX** вышеуказанные изменения вносятся в конфигурационные файлы **Postfix** автоматически с помощью скрипта настройки configure_mta.sh. Соответственно, по умолчанию **Dr.Web для почтовых серверов UNIX** и **Postfix** будут настроены для работы в режиме **after-queue**.

После задания значений всех необходимых параметров следует перезапустить **Postfix**.

Для работы по протоколу Milter

Для работы в этом режиме требуется версия **Postfix** не ниже 2.3.3.

Поскольку по умолчанию **Dr.Web для почтовых серверов UNIX** и **Postfix** настроены для работы в режиме **after-queue**, то новые настройки для работы по протоколу Milter должны будут быть внесены в конфигурационные файлы **Postfix** вместо уже имеющихся (т.е. вместо параметра **content_filter** должен быть указан параметр **smtpd_milters**, а описанные выше изменения в файле master.cf должны быть удалены). В случае необходимости указания ограничений (restrictions), они могут быть заданы отдельно – непосредственно в конфигурационных файлах **Postfix**.

Адрес транспортного соединения, через которое осуществляется взаимодействие между **Postfix** и модулем **drweb-milter** может быть задан как в формате TCP-сокета, так и в формате UNIX-сокета.

Адрес задается в параметре **smtpd_milters** конфигурационного файла системы **Postfix** main.cf. В том случае, если соединение осуществляется через TCP-сокет, значение этого параметра записывается в формате inet:host:port (например, **smtpd_milters**=inet:127.0.0.1:3001). Если же соединение осуществляется через UNIX-сокет, то формат параметра – unix:pathname, где pathname – абсолютный путь к UNIX-сокету.

В случае если адрес задается в формате UNIX-сокета, необходимо, чтобы **Postfix** имел право записи в файл, указанный в качестве сокета

Адрес транспортного соединения между системой **Postfix** и модулем **drweb-milter** также должен быть задан в параметре **Address** секции [Milter] конфигурационного файла **Dr.Web MailD**. Значение и формат этого параметра должны быть идентичны формату и значению параметра **smtpd_milters** файла main.cf.

Кроме транспортного адреса, в конфигурационный файл main.cf необходимо добавить следующие параметры:

- **milter_content_timeout** = 300s – это важный таймаут системы **Postfix**, устанавливающий в том числе и максимальное время проверки письма **Dr.Web MailD** в синхронном режиме. Желательно, чтобы значение этого параметра было больше, чем значение параметра **ProcessingTimeout** секции [Milter] конфигурационного файла **Dr.Web MailD**.
- **milter_default_action** = tempfail – параметр определяет действия **Postfix** при ошибках взаимодействия с модулем **drweb-milter**;
- **milter_protocol** = 6 – требуемая версия протокола Milter;
- **milter_mail_macros** = _ – задание этого параметра необходимо, чтобы **Dr.Web MailD** мог определить IP-адрес и имя хоста, от которого принято сообщение.
- **milter_end_of_data_macros** = i {auth_type} – задание этого параметра позволяет получить идентификатор письма и информацию об авторизации для вывода информации в журнал **drweb-milter**.

Также, пожалуйста, обратите внимание на особенности работы через Milter в синхронном и асинхронном режимах.

Настройка Dr.Web MailD

Если программный комплекс запускается посредством компонента **Dr. Web Monitor**, то в качестве компонента **Receiver** должен запускаться модуль **drweb-milter**. Для этого в файле %etc_dir/monitor/maild_postfix.mmc должна быть раскомментирована строка запуска модуля **drweb-milter** (и, желательно, закомментирована строка запуска модуля **drweb-receiver**). В результате в файле maild_postfix.mmc должны быть строки, близкие к приведенным ниже:

```
# drweb-receiver local:%var_dir/ipc/.agent 15 30 MAIL drweb:drweb
drweb-milter local:%var_dir/ipc/.agent 15 30 MAIL drweb:drweb
```

Кроме того, необходимо настроить работу модуля **drweb-sender**. Для этого в секции [Sender] конфигурационного файла **Dr.Web MailD** должны быть заданы следующие параметры:

```
Address = /usr/local/sbin/sendmail
Method = pipe
MailerName = postfix
```

В параметре **Address** задается путь к утилите **sendmail** из пакета **Postfix**.

После установки всех необходимых параметров следует запустить или перезапустить сначала **Dr.Web MailD**, а затем **Postfix**.

Все параметры работы модуля **drweb-milter** и компонентов **Sender** и **Receiver** сосредоточены в секциях [Receiver], [Sender] и [Milter] конфигурационного файла **Dr.Web MailD** и описаны в главах Секция [Receiver], Секция [Sender] и Секция [Milter] соответственно.

При работе **Dr.Web MailD** с почтовой системой **Postfix** в системе должны быть запущены следующие модули (регулируется в mmc-файле **Dr.Web Monitor**):

- **drweb-notifier**
- **drweb-sender**
- **drweb-maild**
- **drweb-receiver**

Интеграция с почтовой системой Exim

Описание подключения в настоящей документации актуально только для **Exim** версии 4.xx, за настройками более ранних версий **Exim** (3.xx) обращайтесь к соответствующей документации (например, http://www.exim.org/index.html).

При работе **Dr.Web MailD** с почтовой системой **Exim** функции компонента **Receiver** выполняет модуль **drweb-receiver**, а функции компонента **Sender** – модуль **drweb-sender**. Подключение почтовой системы **Exim** к **Dr.Web MailD** может производиться двумя различными способами:

- С помощью специального транспорта.

  Преимущества: нет необходимости в перекомпиляции **Exim** и возможна работа системы с относительно старыми версиями **Exim**.

  Недостатки: меньше производительность системы.

- Через функцию **Exim** `local_scan`. В этом случае компонент **Receiver** получает свою конфигурационную информацию не через компонент **Dr.Web Agent**, как остальные компоненты, а через конфигурационный файл самой почтовой системы **Exim**.

  Преимущества: большая производительность системы.

  Недостатки: необходимость перекомпиляции **Exim**. Требуется версия **Exim** от 4.50.

Обратите внимание, что входящий в поставку **Dr.Web MailD** инициализационный скрипт **configure_mta.sh** (расположен в каталоге `%bin_dir/maild/scripts/`) позволяет выполнить в интерактивном диалоговом режиме предварительную настройку интеграции **Dr.Web MailD** и **Exim** по варианту через специальный транспорт.

Настройка Exim

Первичная настройка системы одинакова для обоих методов подключения:

Сначала необходимо добавить пользователя `drweb` в список доверенных пользователей в секции `MAIN CONFIGURATION SETTINGS` конфигурационного файла почтовой системы **Exim**:

```
##############################################
#          MAIN CONFIGURATION SETTINGS       #
##############################################
trusted_users = drweb
```

Также следует заметить, что в случае, если **Exim** производит доставку почты сразу после ее получения от модуля **drweb-sender**, и в этой доставке случаются значительные задержки (например, она происходит по протоколу SMTP), возможно истечение времени ожидания, заданного значением параметра **PipeTimeout** секции `[Sender]` конфигурационного файла **Dr.Web MailD**, так как **Exim** не возвращает код успешного получения модулю **drweb-sender** до окончания длительного процесса доставки. Чтобы избежать этой проблемы, можно настроить

Exim таким образом, чтобы он отправлял все полученные от **Dr.Web MailD** письма сначала в очередь, и только потом производил их доставку.

Для этого следует добавить новый `acl`:

```
acl_check_drweb_scanned:
warn
condition = ${if and {{def:received_protocol}{eq {${received_protocol}}\
{drweb-scanned}}} {yes}{no}}
control = queue_only
accept
```

а затем подключить его:

```
acl_not_smtp = acl_check_drweb_scanned
```

Подключение с использованием специального транспорта

Приведенное ниже описание ориентировано на версию **Exim** 4.xx, за настройками более ранних версий **Exim** (3.xx) обращайтесь к соответствующей документации (например, по адресу http://www.exim.org/index.html).

В настройках **Exim** необходимо добавить специальный транспорт и роутер. Найдите в конфигурационном файле почтовой системы секцию настройки роутеров. Она начинается со следующего заголовка:

```
#############################################
#       ROUTERS CONFIGURATION               #
# Specifies how remote addresses are handled #
#############################################
#         ORDER DOES MATTER                 #
#  A remote address is passed to each in    #
#      turn until it is accepted.           #
#############################################
```

и сразу после строки:

```
begin routers
```

добавьте в нее следующее описание роутера:

```
drweb_router:
  driver = accept
  condition = "${if eq {$received_protocol}{drweb-scanned}{0}{1}}"
# check_local_user
  retry_use_local_part
  transport = drweb_transport
```

Если необходима проверка получателей в системе, то надо также раскомментировать параметр `check_local_user`.

Далее, в конфигурационном файле **Exim** найдите секцию описания транспортов. Она начинается со следующего заголовка:

```
##############################################
#     TRANSPORTS CONFIGURATION               #
##############################################
#        ORDER DOES NOT MATTER               #
#  Only one appropriate transport is called  #
#          for each delivery.                #
##############################################
```

В эту секцию необходимо добавить описание требуемого транспорта:

```
drweb_transport:
  driver = lmtp
  socket = __ADDRESS__
  batch_max = 100
  timeout = 5m
  user = drweb
# headers_add = "X-Maild-Checked: DrWEB for Exim"
```

Где `_ADDRESS_` – адрес слушающего модуля **drweb-receiver** (параметр **Address** секции `[Receiver]` конфигурационного файла **Dr.Web MailD**), например UNIX-сокет `%var_dir/ipc/.drweb_maild`.

Следующим шагом необходимо в параметре **Address** секции `[Sender]` конфигурационного файла **Dr.Web MailD** указать путь к почтовой системе **Exim**, например `/usr/exim/bin/exim/`, а в параметре **MailerName** секции `[Sender]` указать значение **Exim**.

Подключение с использованием функции local_scan

Работа с **Dr.Web MailD** в этом режиме возможна с почтовой системой **Exim** версии 4.50 или выше.

Обратите внимание, что рассмотренные шаги описывают настройку подключения через функцию `local_scan` в предположении, что при этом конфигурационный файл **Exim** не изменен, т.е. в нем отсутствуют настройки специального транспорта, рассмотренные в предыдущем разделе. Таким образом, если предварительно выполнялась настройка подключения по схеме "через специальный транспорт", перед настройкой подключения через функцию `local_scan` следует привести конфигурационный файл **Exim** к исходному виду, удалив оттуда настройки роутера и транспорта.

Подготовка системы проходит в несколько этапов. Сначала необходимо перекомпилировать **Exim** с поддержкой функции `local_scan`. Для этого:

- Скопируйте файл `%bin_dir/doc/maild/local_scan/local_scan.c` в каталог `exim*/Local/`.
- Добавьте в `Makefile` системы **Exim**, расположенный в каталоге `exim*/Local/`, параметры, заданные в файле `%bin_dir/doc/maild/local_scan/Makefile.sample`. Если соответствующие параметры уже заданы в `Makefile`, можно просто раскомментировать или отредактировать их.

- Укажите в `Makefile` системы **Exim** имя пользователя, с привилегиями которого запускается система **Exim**, такое же, как и для всего программного комплекса. Имя пользователя задается параметром **`EXIM_USER`**. При установках **Dr.Web MailD** по умолчанию, для этого параметра должно быть задано следующее значение:

  `EXIM_USER` = `drweb`

- Скомпилируйте и установите систему **Exim**. Если выполнение **`make`** или **`make`** `install` прерывается с сообщениями об ошибках вида:

  ```
  /libexec/ld-elf.so.1: Shared object "libgcc_s.so.1" not found, required by
  "libboost_thread.so"
  ```

  то есть два варианта:

  - Можно скопировать библиотеки (или организовать на них одноименные ссылки) **`libstdc++.so.6`** и **`libgcc_s.so.1`** из `%bin_dir/lib/` в каталог системных библиотек.
  - Можно выполнить в консоли команду

    ```
    $ export LD_LIBRARY_PATH=%bin_dir/lib/:$LD_LIBRARY_PATH
    ```

    и затем в ней же повторить компиляцию и установку **Exim**.

Далее систему **Exim** следует настроить. Для быстрой настройки можно воспользоваться значениями параметров из файла `%bin_dir/doc/maild/local_scan/configure.sample`, просто скопировав строки с параметрами из этого файла в секцию `local_scan` конфигурационного файла системы **Exim**.

Выполнив команду:

```
$ PATH_TO_BIN_DIR/exim -bP local_scan
```

можно выяснить, с какими настройками будет выполняться компонент **Receiver** (`PATH_TO_BIN_DIR` – путь к каталогу исполняемых файлов **Exim**).

Ниже приведено описание дополнительных параметров для конфигурационного файла **Exim**:

`DrwebTimeout` = `{время}`	Период, в течение которого **Exim** ожидает **`drweb-maild`** для сканирования сообщения. Рекомендуется, чтобы значение этого параметра было больше, чем значение параметра **`SendTimeout`** в секции настроек `[MailBase]`. Значение по умолчанию: **`DrwebTimeout`** = `60s`
`DrwebBaseDir` = `{логический}`	Базовый каталог **Dr.Web MailD**, в котором хранятся сокеты, база данных и т.д. Значение по умолчанию: **`DrwebBaseDir`** = `%var_dir/`
`DrwebProcessingError` = `{действие}`	Действия для писем сообщениям, вызвавших ошибки сканирования (например, если антивирусному модулю не хватает памяти, либо он не может подключиться к **`drweb-maild`**). Допустимые действия: `pass, discard, reject, tempfail` Если для значение параметра **`DrwebProcessingError`** не задано в конфигурационном файле, или по ошибке задано несколько различных значений (например, `discard` и `pass`), то будет применяться действие по умолчанию – `tempfail`

	Значение по умолчанию: **DrwebProcessingError** = tempfail
DrwebLogLevel = {уровень подробности}	Уровень подробности ведения файла журнала. Допустимо использовать следующие уровни: • Quiet • Error • Alert • Info • Debug Значение по умолчанию: **DrwebLogLevel** = Debug
DrwebIpcLevel = {уровень подробности}	Устанавливает уровень подробности журнала работы библиотеки IPC. Допустимо использовать следующие уровни: • Quiet • Error • Alert • Info • Debug Значение по умолчанию: **DrwebLogLevel** = Debug
DrwebSyslogFacility = {метка syslog}	Тип подсистемы, через которую системный сервис **syslog**, осуществляющий журналирование, выдает сообщения о событиях. Значение по умолчанию: **DrwebSyslogFacility** = Daemon
DrwebMaxSize = {размер}	Максимальный размер проверяемого сообщения. При значении 0 ограничения на размер отсутствуют. Значение по умолчанию: **DrwebMaxSize** = 200 k

Настройка Dr.Web MailD

Для настройки совместной работы **Dr.Web MailD** и системы **Exim** следует задать в параметре **Address** секции [Sender] конфигурационного файла **Dr.Web MailD** путь к почтовой системе **Exim**, например /usr/exim/bin/exim/, а в параметре **MailerName** секции [Sender] задать значение Exim.

Так как в режиме подключения **Exim** через функцию local_scan компонент **Receiver** встраивается в саму систему **Exim**, нет необходимости в запуске модуля **drweb-receiver**. Если запуск программного комплекса происходит через компонент **Dr.Web Monitor**, закомментируйте строку запуска модуля **drweb-receiver** в файле %etc_dir/monitor/maild_exim.mmc, например следующим образом:

```
#drweb-receiver local:%var_dir/ipc/.agent 15 30 MAIL drweb:drweb
```

Далее следует запустить **Dr.Web MailD**, а вслед за ним и почтовую систему **Exim**.

Настройки всех параметров работы **Dr.Web MailD** с системой **Exim** (т.е. компонентов **Sender** и **Receiver**) сосредоточены в секциях `[Receiver]` и `[Sender]` конфигурационного файла **Dr.Web MailD** и описаны в главах Секция [Receiver] и Секция [Sender] соответственно.

При работе **Dr.Web MailD** с почтовой системой **Exim** в системе должны быть запущены следующие модули (регулируется в mmc-файле **Dr.Web Monitor**):

- **drweb-notifier**
- **drweb-sender**
- **drweb-maild**
- **drweb-receiver**

Дополнительно следует обеспечить, чтобы компонент **Dr.Web Monitor** запускался с правами `root` (значения параметров **User** и **Group** в секции `[Monitor]` конфигурационного файла `monitor.conf` должны быть установлены в значение `root`).

Известные проблемы

Если при запуске почтовая система **Exim** выдает ошибку вида:

```
transport drweb_transport: cannot find transport driver "lmtp"
```

то это означает, что почтовая система **Exim** была собрана без поддержки LMTP-транспорта. Можно либо перейти на использование SMTP-транспорта (что подробно описано в документации **Exim**, например, по адресу http://www.exim.org/), либо пересобрать **Exim** с поддержкой LMTP-транспорта.

При выборе последнего варианта в файл `/Local/Makefile` системы **Exim** следует добавить строку **TRANSPORT_LMTP** `= yes` или раскомментировать ее, если она уже там есть.

Интеграция с почтовой системой Qmail

Принцип работы фильтра для **Qmail** основан на замещении (проксировании) почтовой системы. Через интерфейс, определенный для модуля **qmail-queue** (основной исполняемый файл почтовой системы **Qmail**), фильтр получает письмо, проверяет его, и если оно является "чистым", переправляет дальше, в оригинальный **qmail-queue**.

У фильтра, работающего в таком режиме, есть следующее ограничение: UNIX-сокеты, на которых **drweb-qmail** должен слушать запросы на проверку (задаются значением параметра **ListenUNIXSocket** секции `[Qmail]` конфигурационного файла **Dr.Web MailD**), должны располагаться в строго определенных каталогах. Список таких каталогов можно получить, запустив **qmail-queue** с параметром `--help`.

Для работы с **Dr.Web MailD** требуется **Qmail** версии не ниже 1.03. Установку фильтра следует производить после остановки **Qmail**, во избежание возможной потери проходящей корреспонденции.

Интеграция **Dr.Web MailD** с **Qmail** может быть выполнена как вручную (см. инструкцию ниже), так и с помощью конфигурационного скрипта **configure_mta.sh** (расположен в каталоге `%bin_dir/maild/scripts/`). Он настраивает взаимодействие, как указано в инструкции ниже, но не выполняет добавления пользователей в соответствующщие группы. Поэтому, если вы использовали для интеграции запуск скрипта настройки, не забудьте выполнить добавление пользователей в нужные группы (см. ниже).

Настройка Qmail

Для подключения **Dr.Web для почтовых серверов UNIX** к почтовой системе **Qmail** вручную необходимо осуществить следующую последовательность действий:

1. Перейдите в каталог исполняемых файлов **Qmail** <qmail_dir> (обычно это каталог /var/qmail/bin), и переименуйте файл **qmail-queue** в **qmail-queue.original**;

   Обратите внимание, что если вы по каким-либо причинам переместили оригинальный файл **qmail-queue** в другоем место, или присвоили ему имя, отличное от **qmail-queue.original**, то необходимо также будет поменять заданный по умолчанию путь к этому файлу в основном конфигурационном файле **Dr.Web MailD,** который задается параметром **QmailQueue** в секции [Qmail]

2. Вместо переименованного файла создайте в этом же каталоге символическую ссылку **qmail-queue** -> %bin_dir/**qmail-queue**;
3. Настройте права пользователей, от имени которых запускаются файлы.

   Наиболее удобна конфигурация, в которой **Dr.Web MailD** и **qmail-queue** работают от имени пользователя drweb. Для правильной работы такой конфигурации следует установить следующие права для %bin_dir/**qmail-queue** и <qmail_dir>/**qmail-queue**:

   ```
   -rws--x--x  X drweb  qmail SIZE DATE %bin_dir/qmail-queue
   -rws--x--x  X qmailq qmail SIZE DATE <qmail_dir>/qmail-queue.original
   ```

   Это можно сделать, выполнив следующие команды:

   ```
   # chown drweb:qmail %bin_dir/qmail-queue
   # chmod 4711 %bin_dir/qmail-queue
   # chown qmailq:qmail <qmail_dir>/qmail-queue.original
   # chmod 4711 <qmail_dir>/qmail-queue.original
   ```

4. Также следует добавить в группу пользователей **drweb** пользователей qmailq и qmaild, а в группу пользователей **qmail** – пользователя drweb.

Настройка Dr.Web MailD

Настройки всех параметров работы **Dr.Web MailD** с системой **Qmail** сосредоточены в секциях [Sender] и [Qmail] конфигурационного файла **Dr.Web MailD** и описаны в главах Секция [Sender] и Секция [Qmail] соответственно.

Рекомендуется задать значение параметра **SecureHash** секции [Sender] конфигурационного файла **Dr.Web MailD** (значением параметра может быть произвольная строка, рекомендуемая длина строки – не менее 10 символов) и установить в Yes значение параметра **UseSecureHash** из этой же секции. Эти параметры регулируют добавление в обрабатываемые письма специального заголовка, предотвращающего зацикливание письма при его проверке (поскольку оно может быть еще раз помещено во входную очередь почтовой системы в случае модификации в ходе проверки).

При работе **Dr.Web MailD** с почтовой системой **Qmail** в системе должны быть запущены следующие модули (регулируется в mmc-файле **Dr.Web Monitor**):

- **drweb-notifier**
- **drweb-sender**
- **drweb-maild**

- **drweb-qmail**

Известные проблемы

Описание:

При запуске **Qmail** выдает одну из следующих ошибок:

```
terminate called after throwing an instance of 'St9bad_alloc'
what(): St9bad_alloc
```

```
bash: xmalloc: cannot allocate 2 bytes (0 bytes allocated)
```

```
qmail-queue.real: error while loading shared libraries: libc.so.6: failed to
map segment from shared object:
Cannot allocate memory
```

```
/var/qmail/bin/qmail-smtpd:
error while loading shared libraries:
libc.so.6: failed to map segment from shared object:
Cannot allocate memory
```

Решение:

Проблема заключается в большом ограничении на используемую память в скрипте запуска. К примеру, если используются скрипты от Dave Sill, то необходимо увеличить значение в инструкции `softlimit -m 2000000`, например, до `20000000`.

Описание:

На все письма, полученные по протоколу SMTP, **Qmail** возвращает после получения тела сообщения строку вида:

```
451 qq trouble making network connection (#4.3.0)
```

Решение:

Возможно, у модуля **qmail-queue** не хватает прав для подключения к UNIX-сокету, созданному модулем **drweb-qmail** (который работает в качестве компонента **Receiver** комплексного компонента **Dr.Web MailD**), или данный UNIX-сокет не находится в путях по умолчанию **qmail-queue**. Проверьте правильность установленных прав, а также убедитесь, что значение параметра **ListenUNIXSocket** секции `[Qmail]` конфигурационного файла **Dr.Web MailD** соответствует путям по умолчанию (их список можно получить командой **qmail-queue** `--help`).

Описание:

На все письма, полученные по протоколу SMTP, **Qmail** после получения тела сообщения выводит в консоль сообщение вида:

```
qmail-inject: fatal: qq temporary problem (#4.3.0)
/usr/libexec/ld-elf.so.1: Shared object "libstdc++.so.6" not found,
required by "libboost_program_options.so"
```

Решение:

Система не может найти необходимые библиотеки, находящиеся в каталоге `%bin_dir/lib/`. Необходимо скопировать библиотеки (или сделать на них линки) **libstdc++.so.6** и **libgcc_s.so.1** из `%bin_dir/lib/` в каталог системных библиотек.

Интеграция с почтовой системой Courier

Настройка Courier

Для подключения **Dr.Web MailD** к почтовой системе **Courier** необходимо проделать следующую последовательность действий:

1. Установить права для модуля **drweb-courier**, выполнив следующие команды:

```
$ chown COURIER_USER:drweb "%bin_dir/drweb-courier"
$ chmod 6771 "%bin_dir/drweb-courier"
```

   где `COURIER_USER` – пользователь, от имени которого запускается почтовая система **Courier**.

   Также следует убедиться, что для всех каталогов и подкаталогов в `%var_dir` для группы `drweb` установлены права на чтение, запись и исполнение.

2. Скопировать модуль **drweb-courier** (или создать на него символическую ссылку) в каталог фильтров **Courier** (обычно это каталог `/usr/local/libexec/filters/`).
3. Зарегистрировать модуль **drweb-courier** в почтовой системе **Courier** как глобальный:

```
$ /usr/local/sbin/filterctl start drweb-courier
```

   В дальнейшем, для выключения фильтрации необходимо будет выполнить команду:

```
$ /usr/local/sbin/filterctl stop drweb-courier
```

4. Создать (отредактировать) управляющий файл `enablefiltering` для задания сервисов для проверки
   (`esmtp` или `uucp` – если указывается несколько, то они разделяются пробелами).
5. Убедиться, что параметры **BaseDir** и **SocketDirs** секции `[Courier]` конфигурационного файла **Dr.Web MailD** соответствуют конфигурации установленной у вас почтовой системы **Courier**. Для получения дополнительной информации достаточно выполнить команду `man courierfilter`.
6. Включить фильтрацию в системе **Courier**:

```
$ /usr/lib/courier/sbin/courierfilter start
```

Пользователь `drweb`, с правами которого работает компонент **Dr.Web Daemon**, должен быть добавлен в группу `courier`, чтобы иметь доступ к чтению файлов, которые создает в спуле почтовая система **Courier**.

Передача обработанных писем в систему Courier

Настройка передачи обработанных писем в систему **Courier** осуществляется в секции `[Sender]` конфигурационного файла. Для этого должны быть заданы следующие параметры:

```
MailerName = Courier
Method = pipe
Address = <путь к системе для отправки сообщений>
#(по умолчанию: /usr/lib/courier/bin/sendmail)
```

Настройка Dr.Web MailD

Настройка всех параметров работы **Dr.Web MailD** с системой **Courier** осуществляется в секциях `[Sender]` и `[Courier]` конфигурационного файла **Dr.Web MailD** и описана в главах Секция [Sender] и Секция [Courier] соответственно.

При работе **Dr.Web MailD** с почтовой системой **Courier** в системе должны быть запущены следующие модули (регулируется в mmc-файле **Dr.Web Monitor**):

- **drweb-notifier**
- **drweb-sender**
- **drweb-maild**
- **drweb-courier**

Интеграция с почтовой системой ZMailer

Модуль **drweb-zmailer** совместим с почтовой системой **ZMailer** только начиная с версии 2.99.55.

Dr.Web MailD можно использовать вместе со **ZMailer** в двух режимах:

- В режиме контент-фильтра на этапе SMTP-соединения.

  Преимущества: возможность заблокировать письмо от клиента уже на этапе SMTP-соединения.

  Недостатки: возможны проблемы с производительностью при высокой нагрузке, осуществляется проверка только SMTP-трафика.

- В режиме контент-фильтра на стадии маршрутизации.

  Преимущества: нет проблем при высокой нагрузке, осуществляется проверка всех писем, проходящих через **ZMailer** (в том числе локальных и проходящих по протоколу `UUCP`).

  Недостатки: невозможно заблокировать письмо на этапе его приема (т.е. фактически действия `reject` и `tempfail` аналогичны действию `discard`), возникает необходимость использовать **SecureHash** для того, чтобы повысить производительность и избежать зацикливания писем.

В качестве компонента **Receiver** комплекса **Dr.Web MailD** для **ZMailer** используется модуль **drweb-zmailer**.

Для корректной работы **drweb-zmailer** и фильтров рекомендуется установить исправления (в том случае, если это возможно).

Для установки исправлений необходимо:

- перейти в каталог:

  ```
  $(ZMAILER_SRCHOME)/smtpserver
  ```

  где `ZMAILER_SRCHOME` – путь к каталогу, содержащему исполняемые файлы **Zmailer**;

- выполнить команду:

  ```
  $ patch < smtpdata.c.XXX.patch
  ```

  где `XXX` – версия **Zmailer**, для которой устанавливается исправление.

При работе **Dr.Web MailD** с почтовой системой **Zmailer** в системе должны быть запущены следующие модули (регулируется в mmc-файле **Dr.Web Monitor**):

- **drweb-notifier**
- **drweb-sender**
- **drweb-maild**

Режим контент-фильтра на этапе SMTP-соединения

Чтобы включить поддержку **Dr.Web MailD** в **ZMailer** необходимо:

- скопировать (или создать символическую ссылку) файл **drweb-zmailer.sh** в каталог `$MAILBIN` (его расположение указано в файле `zmailer.conf`);
- отредактировать файл `smtpserver.conf`, добавив в него следующую строку (или модифицировав существующую):

```
PARAM contentfilter $MAILBIN/drweb-zmailer.sh.
```

Так как параметры командной строки нельзя указывать в `contentfilter`, то их следует задать непосредственно в скрипте запуска **drweb-zmailer.sh**.

Режим контент-фильтра на этапе маршрутизации

Все письма, обрабатываемые почтовым сервером, проходят через этап маршрутизации. Поэтому предпочтительным участком обработки писем для подключения фильтра является момент окончания этапа маршрутизации. Для такого подключения фильтра требуется следующее изменение файла `$MAILBIN/cf/process.cf`:

В этом файле, сразу после текста:

```
LOGMSG=() # This is a LIST of files where to log..
#| The LOGMSG variable is used by the intercept facility (in crossbar.cf)
#| to make sure only a single copy of a message is saved when required.
#| Each sender - recipient address pair can cause an intercept which can
#| specify a file to save the message to. This variable is appended to
#| elsewhere, and processed at the end of this function.
```

следует добавить подобную конструкцию:

```
###-> Dr.Web MailD support
ch='"DEFAULT_BIN_PATH/drweb-zmailer.sh" --hash __EDIT_THIS__ --file
$POSTOFFICE/router/$file'
     case "$ch" in
            -1*) #reject or disacrd
                   /bin/rm -f "$file"
                   return
                   ;;
            1*) #tempfail
                   /bin/rm -f "$file"
                   return
                   ;;
            *);;
     esac
###-> end of Dr.Web MailD support
```

в которой `__EDIT_THIS__` (значение параметра `--hash`) следует заменить на значение, равное значению параметра **SecureHash** в секции `[Sender]` конфигурационного файла **Dr.Web MailD**, и обязательно установить значение `Yes` для параметра **UseSecureHash** в той же секции.

Дополнительная настройка Zmailer

Если необходим простой и быстрый способ запрета получения сообщений с пустым конвертом (SMTP envelope) отправителя (так обычно рассылаются сообщения DSN об ошибках либо о запрете доставки писем, однако рассылка таких писем также является популярным приемом у спамеров), можно установить файл исправления `policytest.c.XXX.patch`. Установка этого файла исправления аналогична процессу установки файла исправления `smtpdata.c.XXX.patch`.

Поскольку **ZMailer** запускает модуль **drweb-zmailer** отдельно практически для каждого обрабатываемого письма, в целях оптимизации работы сопряжения **Dr.Web MailD** — **Zmailer** все настройки модуля **drweb-zmailer** указываются в командной строке (их можно задать, например, в скрипте **drweb-zmailer.sh)** .

Параметры, которые могут быть заданы в командной строке модуля **drweb-zmailer**, см. в разделе Параметры командной строки.

Известные проблемы

Проблема:

При больших нагрузках, либо когда недоступен один из сканирующих демонов **Dr.Web Daemon**, адреса которых перечислены в настройках подключаемого модуля **Drweb**, а также если в его настройках указана большая величина тайм-аута сканирования (параметр `Timeout`), то при перезапуске комплекса **Dr.Web MailD** по сигналу `SIGHUP` может наблюдаться следующая картина: компоненты **MailD core** и **Notifier** завершаются с ошибкой, в результате чего компонент **Dr.Web Monitor** перезапускает эти компоненты и отправляет по электронной почте уведомление для администратора об инциденте. Записи в журнале **Dr.Web Monitor** имеют следующий вид (пример):

```
monitor ERROR component "drweb-maild" terminated by signal 6 (Aborted)
monitor DEBUG component "drweb-maild" cannot stop
monitor DEBUG send notification From:<e-mail@address>#012To:<e-
mail@address>. Command: /usr/sbin/sendmail -t
```

Решение:

Ошибка вызвана тем, что компоненты **MailD core** и **Notifier** создают столь большое количество активных потоков, занимающихся обработкой трафика, что при поступлении сигнала `SIGHUP` они не успевают завершиться корректно за период времени, заданный параметром **MaxTimeoutForThreadActivity**. Это признак того, что программно-аппаратный комплекс, занимающийся обработкой почтового трафика, перегружен (как минимум – на пиках поступления трафика). Следует:

- Увеличить величину тайм-аута **MaxTimeoutForThreadActivity** или ограничить число активных потоков в пулах потоков этих компонентов.

Крайне рекомендуется выполнить следующий комплекс мер по стабилизации работы программного комплекса:

- Измерить объем обрабатываемого почтового трафика на пиках его поступления (используйте журналы и статистику);
- Выполнить мероприятия по оптимизации использования **Dr.Web MailD** системных ресурсов;
- В случае невозможности оптимизации, или если она не приносит желаемых результатов – выполнить модернизацию аппаратного обеспечения (увеличение объемов оперативной памяти и количества доступных процессорных ядер).

В качестве временной меры вы можете объявить переменную окружения **DW_FORCE_EXIT** (с любым значением). В этом случае **Dr.Web Monitor** при зависании компонентов при завершении их работы не будет слать администратору уведомлений, а просто завершит работу всего комплекса и завершится сам (используется для совместимости с предыдущими версиями **Dr.Web MailD**).

Проблема:

При запуске **Dr.Web MailD** компонент **Dr.Web Monitor** аварийно завершает компонент **MailD core** и отправляет по электронной почте уведомление для администратора об инциденте. Записи в журнале **Dr.Web Monitor** имеют следующий вид (пример):

```
monitor DEBUG DEBUG component "drweb-maild" not answer
monitor DEBUG Component::stopPid() # drweb-maild <comp name="drweb-maild"
argv="local:/var/drweb/ipc/.agent" start="120" stop="30" log="2"
user="drweb:drweb" fd="-1" pid="7194"/>
monitor DEBUG kill -TERM pid=7194 name="drweb-maild"
```

Решение:

Ошибка вызвана тем, что компонент **MailD core** создает при запуске столь большое количество потоков, что он не успевает стартовать за период времени, заданный в настройках запуска у **Dr.Web Monitor**. Следует:

- Ограничить минимальное число активных потоков в пулах потоков **MailD core**.

Крайне рекомендуется выполнить следующий комплекс мер по стабилизации работы программного комплекса:

- Измерить объем обрабатываемого почтового трафика на пиках его поступления (используйте журналы и статистику);
- Выполнить мероприятия по оптимизации использования **Dr.Web MailD** системных ресурсов;
- В случае невозможности оптимизации, или если она не приносит желаемых результатов – выполнить модернизацию аппаратного обеспечения (увеличение объемов оперативной памяти и количества доступных процессорных ядер).

Проблема:

При использовании ОС **Solaris**, при генерации уведомлений по обработанному письму или в ходе самой обработки письма, **Dr.Web MailD** фиксирует в журнале WARN-сообщения вида

```
notifier WARN Decoding string ' \362\345\361\362
notifier WARN because of iconv error: Invalid argument
```

Решение:

Для решения проблемы необходимо заменить версию системного конвертера **iconv** и использовать **iconv** из пакета `libiconv`, который распространяется под лицензией GNU (Загрузку можно выполнить по URL http://www.gnu.org/software/libiconv/#downloading).

Проблема:

При использовании ОС **Solaris** версии 10, 32-бит, при большой нагрузке на **Dr.Web MailD** может возникнуть следующая проблема:

Не принимаются к обработке сообщения, **Dr.Web MailD** фиксирует в журнале сообщения вида `"Too many open files"`.

Решение:

Возникновение ошибки вызвано исчерпанием количества файловых дескрипторов, доступных **Dr.Web MailD** (в том числе – дескрипторов сокетов). Следует выполнить процедуры по Оптимизации работы и использования системных ресурсов.

Проблема:

Не удается выполнить подключение к источникам данных **MySQL**, **Dr.Web MailD** фиксирует в журнале сообщения вида

```
Cannot load library: Cannot load shared library libmysqlclient_r.so.18
because libmysqlclient_r.so.18: Undefined symbol "strnlen"
```

Решение:

Возникновение ошибки характерно только для ОС **FreeBSD**, в случае если в состав решения

Dr.Web для почтовых серверов UNIX включена библиотека `/usr/local/drweb/lib64/libc.so.7`.

Для решения проблемы необходимо заменить в каталоге `/usr/local/drweb/lib64` библиотеку **`libc.so.7`** на одноименную символическую ссылку на системную библиотеку **`/lib/libc.so.7`**.

Проблема:

В журнале работы **Dr.Web MailD** присутствуют сообщения вида

```
Can not send msg from temp dir ('"<dir_path>"') -> remove dir and forget about it
```

где `<dir_path>` – путь к некоторому каталогу в хранилище писем (например, `/var/drweb/msgs/db/A/00000B9A`).

Решение:

Описанная ситуация не является индикатором ошибки обработки писем. Возникновение сообщения может быть спровоцировано попыткой повторной отправки писем в результате перезапуска **Dr.Web MailD** после аварийного завершения работы, и сигнализирует лишь о том, что сами письма уже были успешно отправлены в целевой MTA, но центральный компонент **MailD core** не успел очистить более не нужные каталоги этих писем в файловом хранилище.

Проблема:

Пользователям, которые используют почтовый сервер **MS Exchange**, в случае недоставки их писем, возвращаются нечитаемые DSN.

Решение:

Описанная проблема связана с особенностями реализации **MS Exchange,** которая не полностью соответствует требованиям `RFC 3464`. Для ее решения следует заменить стандартный шаблон уведомлений DSN `dsn.msg`, на специальную версию шаблона, разработанного для **MS Exchange.** Этот шаблон хранится в файле `dsn_for_exchange.msg`.

Способы замены шаблона рассмотрены в описании Обработки уведомлений.

Консоль Dr.Web для почтовых серверов UNIX

Настройка программного комплекса **Dr.Web для почтовых серверов UNIX** может быть осуществлена через специально разработанный веб-интерфейс **Консоль Dr.Web для почтовых серверов UNIX** (далее **Консоль**). Он реализован в виде дополнения к интерфейсу **Webmin** (подробная информация об интерфейсе **Webmin** доступна на официальном сайте производителя: http://www.webmin.com/).

Для успешной работы **Консоли** необходимо, чтобы в системе были установлены следующие модули **Perl**:

- **XML::Parser** – модуль для преобразования документов в формате XML;
- **XML::XPath** – набор модулей для преобразования инструкций `XPath`;
- **Encode** – модуль для управления функцией преобразования кодировки;
- **Date::Parse** – модуль для преобразования даты в UNIX-формат;
- **CGI** – модуль для работы с Common Gateway Interface;
- **CGI::Carp** – модуль для создания HTTPD-отчета об ошибках;
- **JSON** — модуль для преобразования данных в формате JSON (JavaScript Object Notation).
- **Digest::MD5** – модуль для подсчета контрольных сумм;
- **MIME::Words** – модуль для работы с кодировкой RFC 2047.
- **MIME::Entity** – модуль для преобразования и раскодирования MIME-сообщений;
- **MIME::Parser** – модуль для преобразования MIME-потоков;
- **MIME::Head** – модуль для преобразования заголовков MIME-сообщений;
- **File::Stat** – модуль интерфейса для встроенных функций `stat()`.
- **File::Find** – модуль интерфейса для осуществления поиска по дереву каталогов.
- **Encode::CN** – модуль для работы с китайской кодировкой.
- **Encode::HanExtra** – модуль с дополнительным набором китайских кодировок.
- **Switch** – модуль для использования конструкций `switch-case`.

Недостающие модули рекомендуется устанавливать из командной строки. Для установки требуются права `root`. Имена модулей могут различаться, однако, как правило, они содержатся в пакетах `perl-Convert-BinHex`, `perl-IO-stringy`, `perl-MIME-tools`, `perl-XML-Parser`, `perl-XML-XPath`. Для установки в `rpm` системах рекомендуется выбирать `noarch.rpm` пакеты.

При запуске в разных браузерах и при использовании разных версий **Webmin** во внешнем виде веб-интерфейса могут наблюдаться отличия от приведенных скриншотов.

Ввиду особенности реализации **Webmin**, интерфейс **Консоли** не может быть корректно отображен в браузере **Internet Explorer** 7. В случае возникновения проблем с отображением страниц, попробуйте воспользоваться **Internet Explorer** 8 или 9 (и более поздними версиями), или использовать другой браузер.

Установка

Для начала работы с **Консолью** необходимо:

- установить **Webmin**;
- подключить модуль **Dr.Web консоль для почтовых серверов UNIX** к **Webmin** (расположен в каталоге `%bin_dir/web/`).

Подключение модуля, а также настройка дополнительных параметров самого **Webmin** осуществляется через его веб-интерфейс.

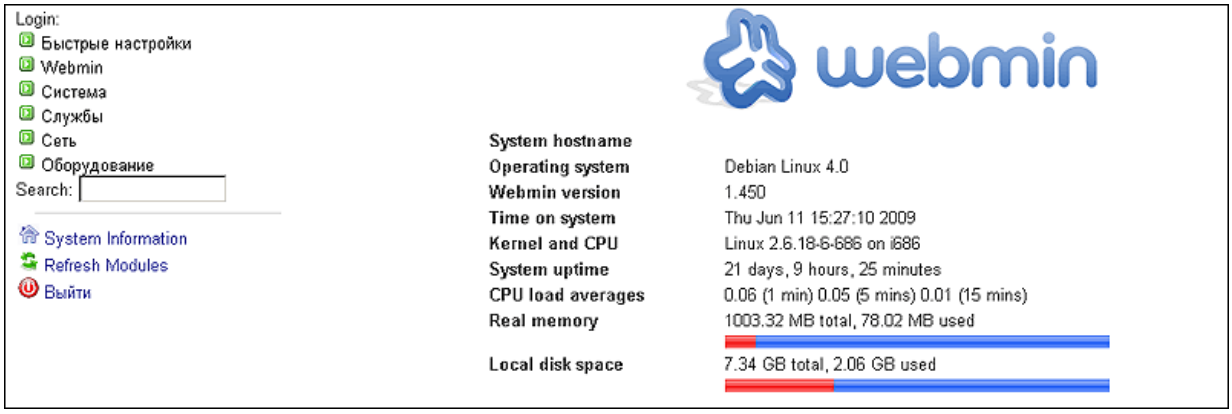


Рис. 23. Главная страница Webmin

Установка дополнительных модулей происходит в разделе **Настройка Webmin** секции **Webmin** основного меню, в подразделе **Модули Webmin**.

Рис. 24. Настройка Webmin

Чтобы установить нужный модуль, в открывшемся окне **Модули Webmin** нажмите кнопку **Обзор** напротив строки **Из локального файла**. Откроется отдельное окно браузера для навигации по списку файлов и каталогов вашей системы, в котором вы сможете выбрать соответствующий установочный пакет (по умолчанию - `%bin_dir/web/drweb-maild-web.wbm.gz`).

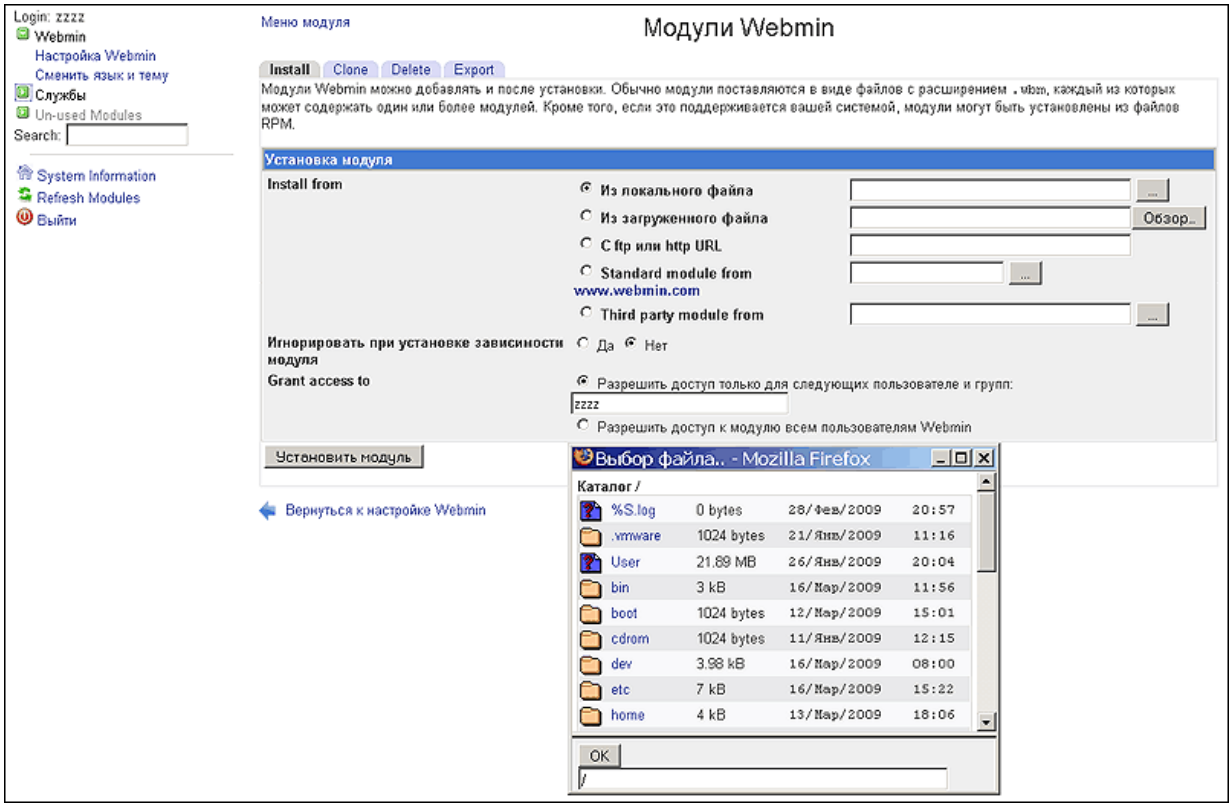


Рис. 25. Добавление модулей Webmin

После одного клика левой кнопкой мыши на какой-либо элемент списка в строке ввода прописывается путь к этому элементу.

После повторного клика левой кнопкой мыши на пиктограмму или название каталога он открывается.

Повторным кликом левой кнопкой мыши на иконку или название файла вы выбираете соответствующий модуль для установки в **Webmin**. Соответственно, окно выбора файла закрывается, а путь к этому файлу появляется в поле **Из локального файла**. Также вы можете нажать кнопку **OK** после того, как выбор нужного файла будет сделан.

Выбрав необходимый файл, нажмите кнопку **Установить модуль**. По завершении установки в секции **Службы** основного меню появится ссылка на новый раздел **Dr.Web консоль для почтовых серверов Unix**.

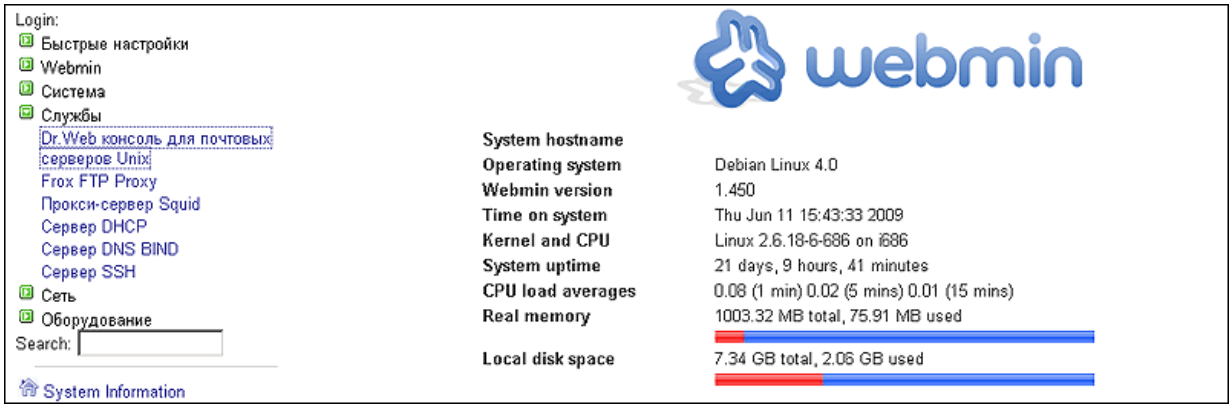


Рис. 26. Новый пункт меню "Dr.Web консоль для почтовых серверов Unix"

Кроме того, для **Webmin** версии 1.680 или старше, требуется также добавить в файл конфигурации **Webmin** (как правило, это файл `/etc/webmin/config`) следующую строчку:

```
no_content_security_policy=1
```

Настройка

Базовые настройки модуля **Dr.Web консоль для почтовых серверов UNIX** можно найти, если пройти по ссылке **Настройка интерфейса** в самом верху страницы соответствующего раздела. На открывшейся странице вы сможете указать используемую почтовую систему, путь к конфигурационному файлу, пути к init-скрипту и скрипту отсылки почты, используемый по умолчанию адрес электронной почты для подстановки в поле **From** писем с уведомлениями о работе **Dr.Web для почтовых серверов UNIX**, а также режим работы.

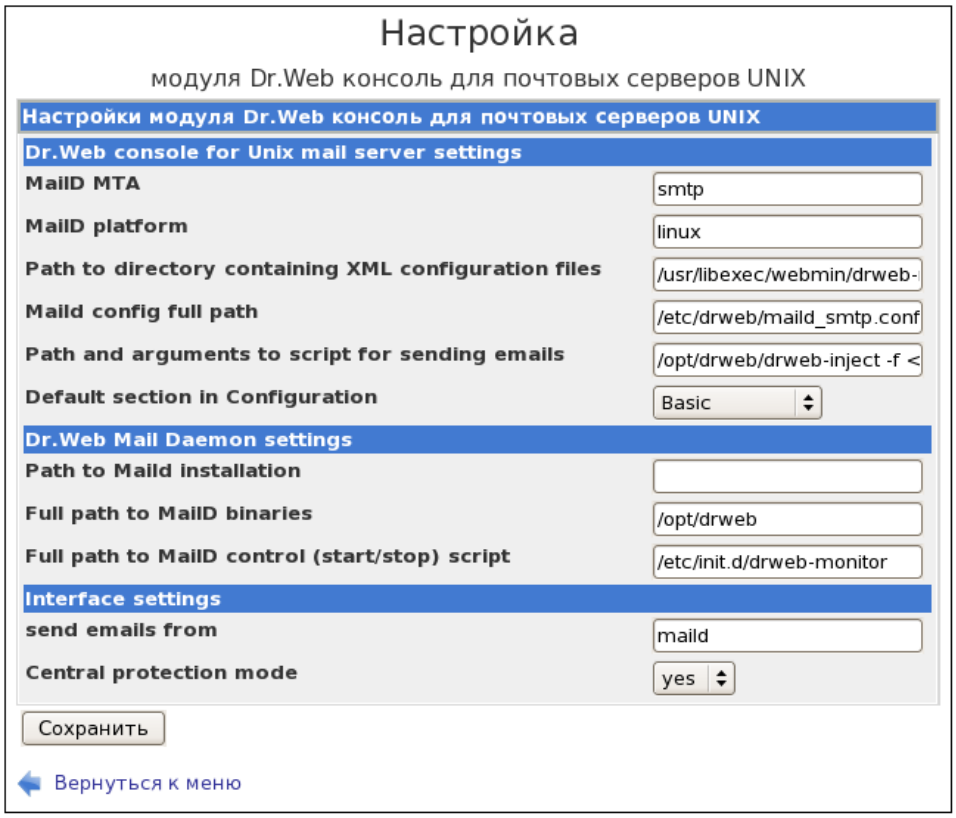


Рис. 27. Настройка модуля

Не забудьте изменить значение по умолчанию поля **send emails from**. В противном случае письма, отправленные из **Карантина** (например, в случае ложного срабатывания фильтра), а также письма с уведомлениями о работе системы могут не доходить до получателей.

Пользовательский интерфейс

Пожалуйста, обратите внимание на то, что при навигации внутри разделов **Консоли** невозможно перейти на предыдущую страницу при помощи стандартной функции браузера **Назад**. Если вы нажмете кнопку **Назад** или соответствующую комбинацию клавиш, вы попадете к предыдущему разделу главного меню.

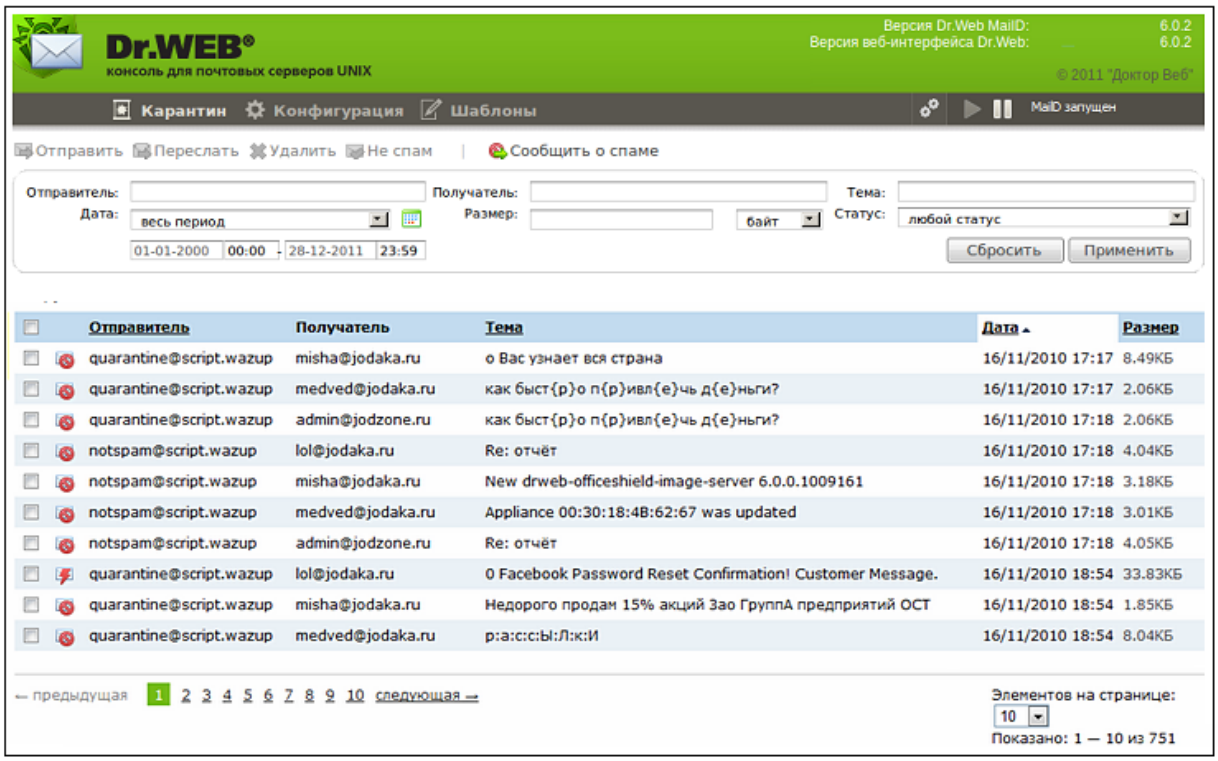


Рис. 28. Dr.Web консоль для почтовых серверов UNIX

Справа от заголовка модуля вы найдете информацию о текущей версии **Dr.Web MailD** и веб-интерфейса **Консоль Dr.Web для почтовых серверов UNIX**.

Под заголовком модуля расположены три секции: **Карантин**, **Конфигурация** и **Шаблоны**. По умолчанию при входе в раздел открывается главная страница секции **Карантин**.

Рядом с заголовками секций расположены три кнопки: **Настройка интерфейса,** **Запустить Dr.Web MailD** и **Остановить Dr.Web MailD**, а также текущее состояние **Dr.Web MailD**. При работе в режиме централизованной защиты кнопка **Остановить Dr.Web MailD** остановит также все прочие локальные сервисы **Dr.Web для почтовых серверов UNIX**, запущенные в режиме централизованной защиты.

Если **Dr.Web для почтовых серверов UNIX** работает в режиме централизованной защиты, то после изменении прав доступа к настройкам в **Центре Управления Dr.Web ES**, необходимо вручную перезагрузить страницу с веб-интерфейсом, чтобы изменения вступили в силу.

Карантин

В **Карантин** перемещаются письма, отфильтрованные каким-либо из антивирусных и антиспам подключаемых модулей **Dr.Web для почтовых серверов UNIX** по причине содержания в них вирусов или спама. На вкладке **Карантин** веб-интерфейса **Dr.Web для почтовых серверов UNIX** расположены все необходимые инструменты для работы с письмами, помещенными в **Карантин**.

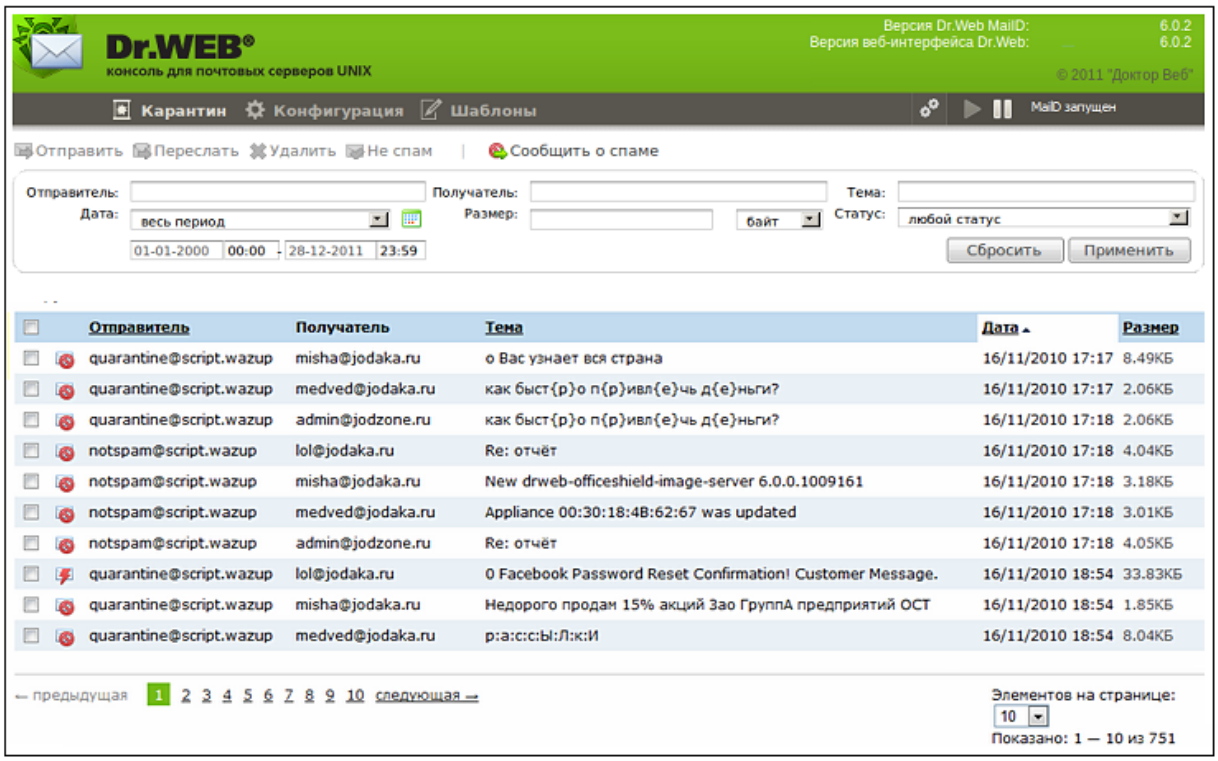


Рис. 29. Вкладка "Карантин"

Вкладка **Карантин** содержит следующие элементы:

- панель инструментов;
- панель фильтров;
- таблицу со списком писем, помещенных в **Карантин**;
- дополнительные средства навигации по списку и настройки его отображения.

Обратите внимание, что веб-интерфейс **Dr.Web для почтовых серверов UNIX** не предусматривает управление "потерянными" письмами, находящимися в каталоге `/out/failed` хранилища писем (подробнее о "потерянных" письмах см. в описании параметров компонента **Sender**).

Для обнаружения таких писем необходимо проверять этот каталог периодически вручную, и при обнаружении "потерянных" писем, их можно либо явно отправить получателям при помощи утилиты **`drweb-inject`**, либо удалить средствами ОС.

Панель инструментов

Элементы панели инструментов (за исключением кнопки **Пожаловаться на спам**) становятся активны только тогда, когда выделяется какое-либо из содержащихся в **Карантине** писем.

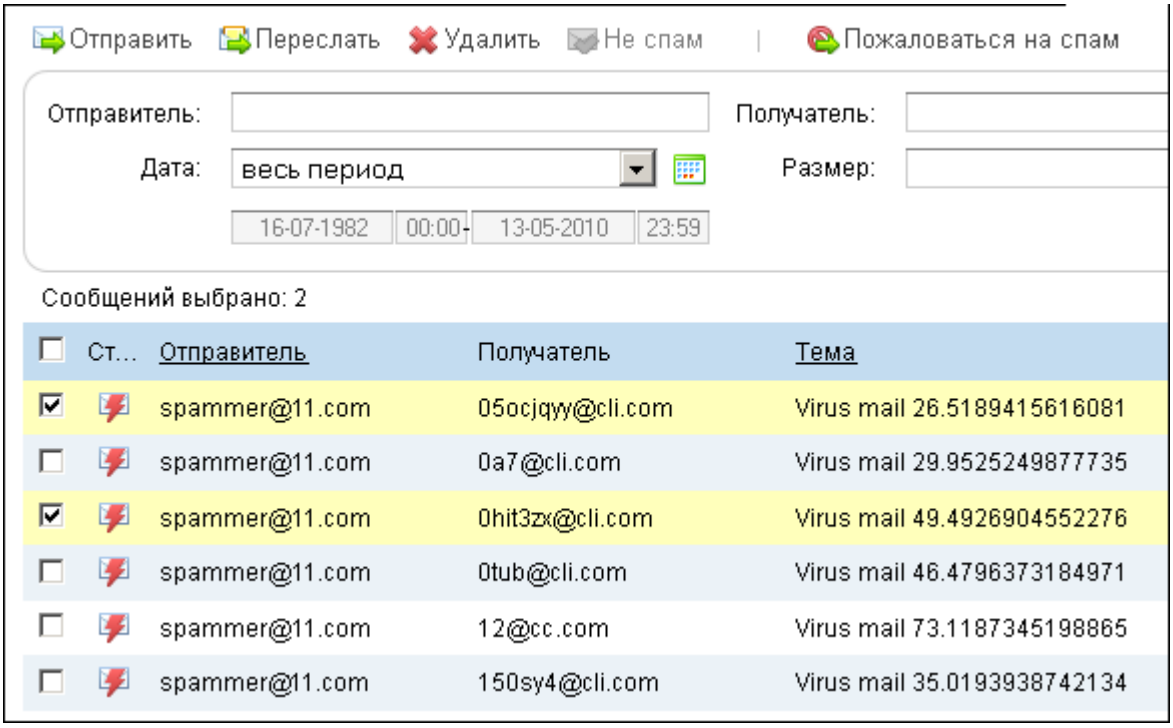


Рис. 30. Панель инструментов

С помощью панели инструментов можно:

- Отправить одно или несколько писем указанным в них получателям. Для этого нужно выбрать письма из списка и нажать на кнопку **Отправить.**
- Переслать одно или несколько писем. Для этого нужно выбрать письма из списка и нажать на кнопку **Переслать.** В результате этого действия откроется дополнительное окно с полями **Получатель** (адрес электронной почты), **Тема** (любой текст с темой письма), **Сообщение** (поле для сообщения), **Вложения** (пересылаемые письма в виде прикрепленных документов).
- Удалить одно или несколько писем. Для этого нужно выбрать письма из списка и нажать на кнопку **Удалить** либо воспользоваться клавишей DEL.
- Сообщить об ошибочном срабатывании антиспам-модуля. Для этого нужно выбрать в списке письма с ошибочно присвоенным статусом "спам" и нажать на кнопку **Не спам**. После этого автоматически будет сформировано и отправлено на адрес vrnonspam@drweb.com письмо, содержащее сообщение об ошибочном определении спама и отмеченные письма в качестве вложений. Сами отмеченные письма не будут отправлены получателям или удалены, а останутся в **Карантине**. Вы можете отправить их получателям или переслать на другой адрес, воспользовавшись соответствующими кнопками панели инструментов.
- Пожаловаться на спам. После нажатия на кнопку **Пожаловаться на спам** откроется дополнительное окно с предложением выбрать файл, в котором содержится письмо-спам, и отправить его на проверку в соответствующий отдел компании **«Доктор Веб»**.

Данная функция не предназначена для работы с элементами списка. Письмо, являющееся с точки зрения пользователя спамом, должно быть предварительно сохранено в файловой системе.

Панель фильтров

Панель фильтров предназначена для удобства при обработке писем, помещенных в **Карантин**.

Рис. 31. Панель фильтров

С помощью системы фильтров можно выбрать письма по следующим критериям:

- **Отправитель** - адрес электронной почты отправителя письма. В данное поле вводится либо адрес отправителя целиком, либо какая-нибудь его часть.
- **Получатель** - адрес электронной почты получателя письма. В данное поле вводится либо адрес получателя целиком, либо какая-нибудь его часть.
- **Тема** - любой текст. В результаты поиска попадут только те письма, в поле **Тема** которых будет найдено полное или частичное совпадение со введенным текстом.
- **Дата** - дата помещения письма в **Карантин**. Временной период можно выбрать из выпадающего списка или с помощью календаря по кнопке справа от выпадающего списка. Доступны следующие настройки:
  - **весь период** - выбрать все письма, помещенные в **Карантин** и хранящиеся в нем.
  - **за сегодня** - выбрать письма, помещенные в **Карантин** с начала текущих суток и по настоящий момент.
  - **за вчера** - выбрать письма, помещенные в **Карантин** с начала вчерашних суток и до начала текущих.
  - **за неделю** - выбрать письма, помещенные в **Карантин** за текущую неделю (с 00:00 часов понедельника и по настоящий момент).
  - **за месяц** - выбрать письма, помещенные в **Карантин** за период с 00:00 часов 1-го числа текущего месяца и по настоящий момент.
  - **другой период** - выбрать письма за любой другой период, который можно указать с использованием календаря.

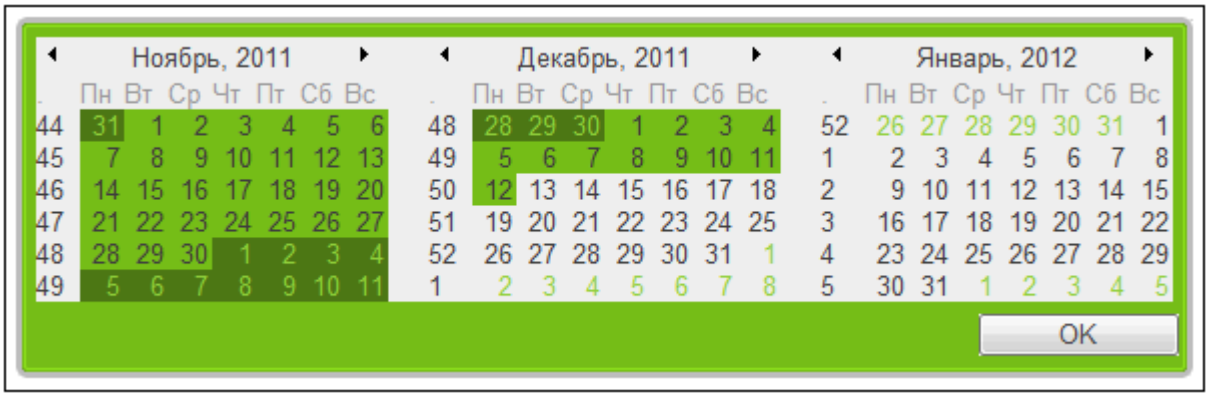


Рис. 32. Календарь

Окно календаря открывается автоматически при выборе пункта **другой период** или после нажатия на значок календаря . При использовании календаря необходимо указать границы временного интервала для поиска писем, после чего необходимо нажать на кнопку **ОК**. Окно календаря закроется, и в соответствующих полях ввода появятся выбранные значения.

Можно также указать точное время или интервал времени для поиска писем.

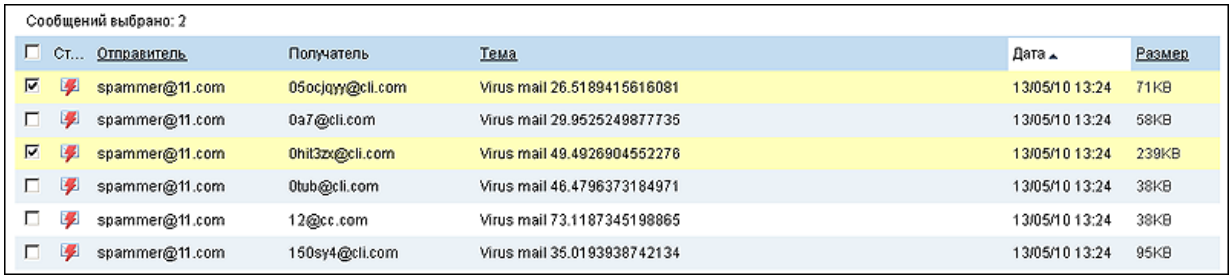
При указании интервала времени будут выбраны те письма, время помещения в **Карантин** которых попадает в указанный интервал, включая его границы. Таким образом, если в качестве начала и конца интервала указать один и тот же момент времени, то будут выбраны только те письма, которые пришли строго в указанное время.

- **Размер** – числовое значение. По умолчанию введенное значение рассматривается как размер письма в байтах, однако с помощью выпадающего списка можно задать размер письма в килобайтах и мегабайтах. При использовании этого критерия будут отобраны письма размер, которых больше или равен введенному значению. Если значение равно нулю, то данный критерий при поиске не учитывается.
- **Статус** – причина, по которой письмо было отправлено в **Карантин**. Доступны следующие причины, которые можно выбрать из выпадающего списка:
  - **Вирус** – письмо было отправлено в **Карантин** антивирусным модулем **Dr.Web для почтовых серверов UNIX** как содержащее вирусы;
  - **Спам** – письмо было отправлено в **Карантин** антиспам-модулем **Dr.Web для почтовых серверов UNIX** как содержащее спам;
  - **Правила** – письмо было отправлено в **Карантин** согласно внутренним Правилам обработки писем;
  - **Ошибка обработки** – письмо было отправлено в **Карантин**, т.к. вызвало ошибку в процессе обработки.

После ввода критериев поиска необходимо нажать на кнопку **Применить**, и список сообщений обновится. Для возврата к значениям фильтров по умолчанию необходимо нажать на кнопку **Сброс**.

Список писем

При наличии писем в **Карантине** на вкладке **Карантин** отображается список данных писем, представленный в виде таблицы.


Сообщений выбрано: 2

	Ст...	Отправитель	Получатель	Тема	Дата	Размер
☑		spammer@11.com	05ocjqyy@cli.com	Virus mail 26.5189415616081	13/05/10 13:24	71KB
☐		spammer@11.com	0a7@cli.com	Virus mail 29.9525249877735	13/05/10 13:24	58KB
☑		spammer@11.com	0hit3zx@cli.com	Virus mail 49.4926904552276	13/05/10 13:24	239KB
☐		spammer@11.com	0tub@cli.com	Virus mail 46.4796373184971	13/05/10 13:24	38KB
☐		spammer@11.com	12@cc.com	Virus mail 73.1187345198865	13/05/10 13:24	38KB
☐		spammer@11.com	150sy4@cli.com	Virus mail 35.0193938742134	13/05/10 13:24	95KB

Рис. 33. Список писем

Администратору доступны письма пользователей из всех подчиненных ему групп.

Данные в таблице хранятся в следующих колонках:

- **Статус** – содержит статусы писем (т.е. причины, по которым эти письма были помещены в **Карантин**). Все статусы отображены в виде соответствующих значков:

  – письмо содержит вирус,

  – письмо отмечено как спам,

  – письмо отправлено в **Карантин** согласно внутренним Правилам обработки писем,

  – письмо вызвало ошибку в процессе обработки.

  При наведении указателя мыши на значок статуса во всплывающей подсказке отображается подробное описание причины, по которой письмо помещено в **Карантин**.
- **Отправитель** – содержит адрес электронной почты отправителя. Возможна сортировка писем

по адресу отправителя в прямом или обратном алфавитном порядке.

- **Получатель** – содержит адрес электронной почты получателя. Возможна сортировка писем по адресу получателя в прямом или обратном алфавитном порядке.
- **Тема** – содержит тему письма. Возможна сортировка писем по теме письма в прямом или обратном алфавитном порядке.
- **Дата** – содержит дату помещения письма в **Карантин**. Для писем, помещенных в **Карантин** в течение текущих суток, отображается только время. Возможна сортировка по возрастанию или убыванию даты.
- **Размер** – содержит размер письма. Возможна сортировка по убыванию или возрастанию размера писем.

Выделить какое-либо из писем в списке можно, установив флаг в соответствующей ячейке слева от статуса письма. Чтобы выделить все письма, необходимо установить флаг в ячейке в заголовке таблицы.

Значения полей **Получатель**, **Тема** и **Дата** — ссылки, при клике на которые откроется для просмотра соответствующее письмо.

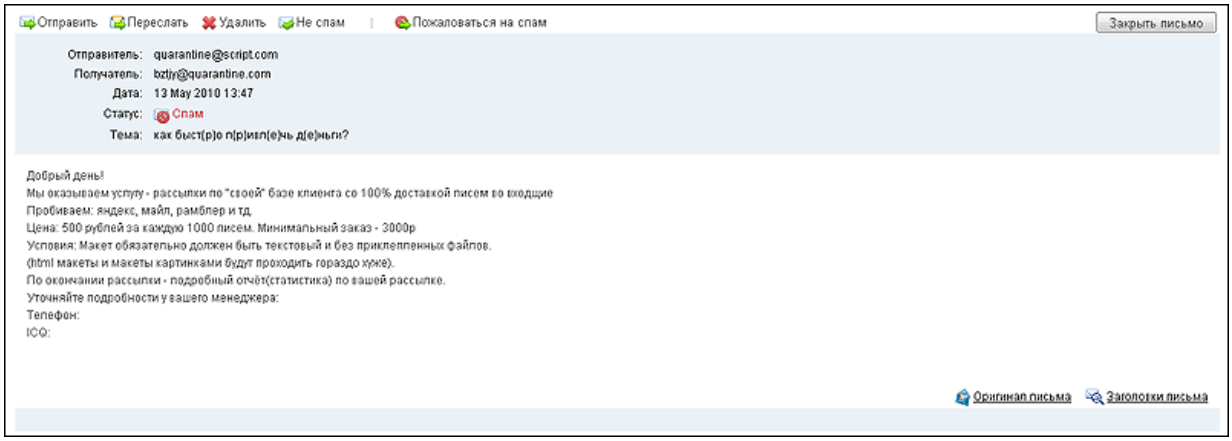


Рис. 34. Письмо

На странице письма можно просмотреть его содержимое, исходный код (нажав значок **Оригинал письма**), заголовки (нажав значок **Заголовки письма**) и вложения (если таковые имеются).

Для возвращения к главной странице секции **Карантина** нажмите кнопку Закрыть письмо.

Панель навигации

Рис. 35. Панель навигации

Дополнительные средства навигации по списку включают:

- навигатор для перехода на следующую или предыдущую страницу таблицы в виде ссылок **предыдущая** и **следующая** (одновременное нажатие клавиши CTRL и стрелок вправо и влево обеспечит переход на следующую или предыдущую страницы соответственно);
- представленные в виде ссылок номера страниц для быстрого перехода на нужную страницу таблицы. Ссылка на текущую страницу подсвечена зеленым цветом и неактивна.
- указатель количества сообщений. Содержит информацию об общем количестве писем в

списке, а также о том, сообщения с какими порядковыми номерами отображены на активной странице таблицы.

Настройка количества сообщений в таблице на одной странице реализована с помощью выпадающего списка с предлагаемыми значениями `10`, `20`, `50`, `100`. При выборе необходимого значения таблица автоматически переформатируется.

При переформатировании списка или при сортировке его содержимого выделение с элементов списка снимается.

Конфигурация

Вы можете выбирать нужные значения параметров из раскрывающихся списков, либо нажимать кнопку  в соответствующих местах, либо задавать эти значения вручную в полях ввода. Подробное описание каждого параметра вы найдете в интерактивной справке по ссылке **подробнее**.

После того, как вы изменили значение какого-либо параметра, вы можете всего лишь одним щелчком мыши по соответствующей иконке рядом с параметром немедленно отменить изменение  или восстановить настройки по умолчанию  . Последняя операция доступна всегда, даже после сохранения изменений.

Чтобы просмотреть все сделанные изменения, используйте кнопку **Предварительный Просмотр**. На появившейся странице вы можете выбрать те изменения, которые желаете сохранить, отметив флажком **Сохранять** соответствующую строку таблицы.

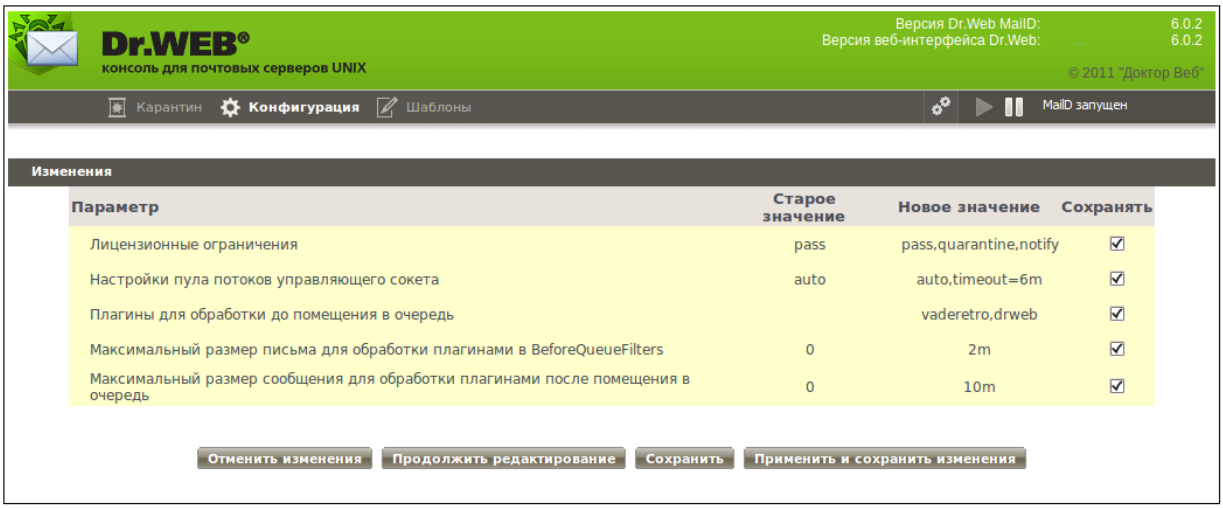


Рис. 36. Страница предпросмотра

- Нажатие кнопки **Отменить изменения** позволяет отказаться от изменения значений параметров, вернув их в состояние, в котором они были до начала редактирования.
- Если вы хотите внести дополнительные изменения, вы можете вернуться к предыдущей странице, нажав на кнопку **Продолжить редактирование**.
- Нажатие кнопки **Сохранить** позволяет сохранить внесенные изменения, а нажатие кнопки **Применить и сохранить изменения** – не только сохранить, но и немедленно применить изменения, внесенные в значения параметров.

Вкладка "Базовые настройки"

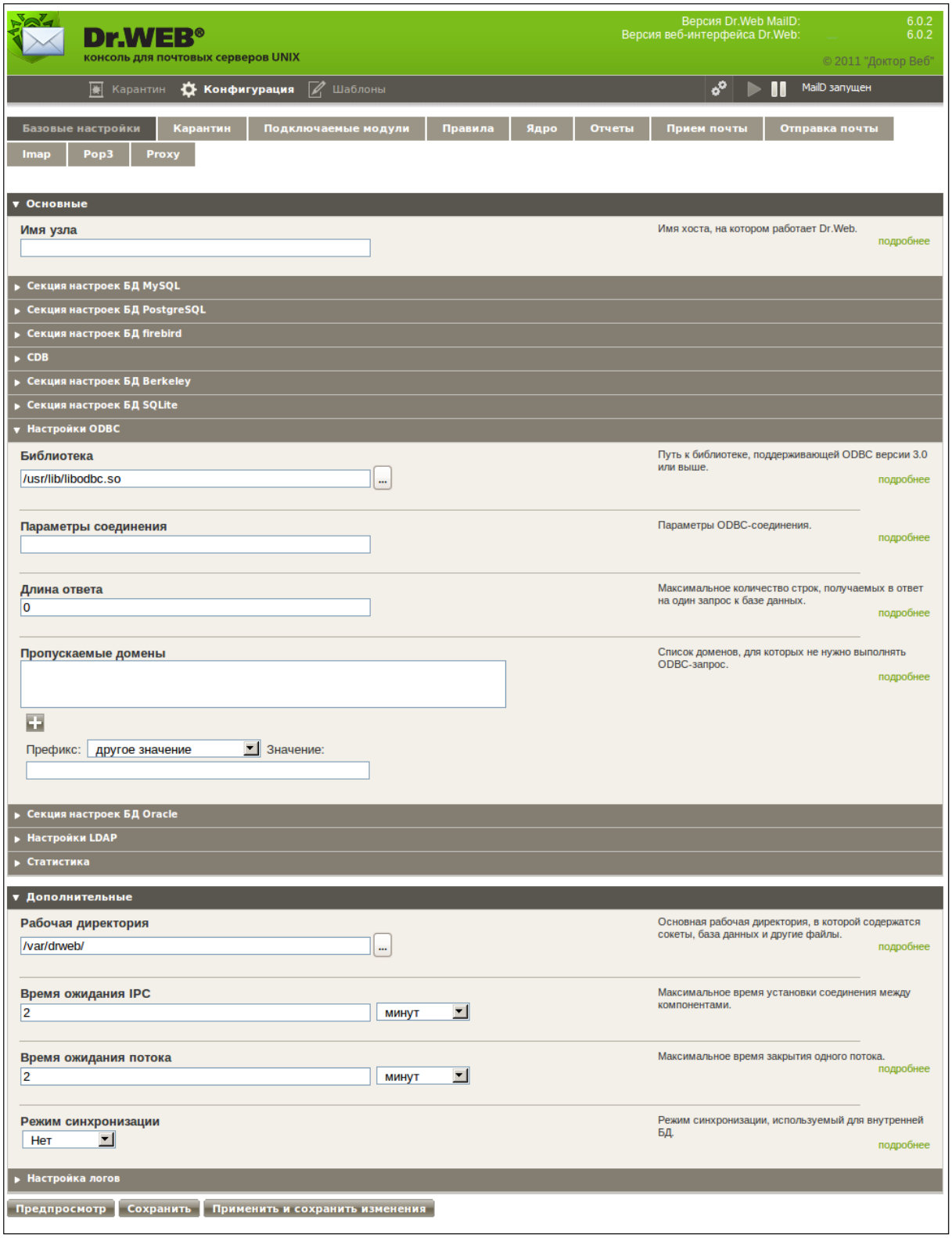


Рис. 37. Базовые настройки

На этой вкладке вы можете настроить экспорт статистики и взаимодействие **Dr.Web MailD** с различными базами данных. Значения параметров могут быть выбраны из раскрывающихся списков или заданы вручную в соответствующих полях ввода. Запросы к серверу LDAP должны

начинаться с двойного или тройного слеша.

Пример:

```
//127.0.0.1/dc=origin?description?sub?(cn=$u)
```

Форма записи с двойным слешем используется, когда необходимо указать адрес LDAP-сервера.

Пример:

```
///?description?sub?(cn=$u)
```

При записи запроса с использованием тройного слеша, используется сервер, указанный в значении параметра **Hostname** секции `[LDAP]` конфигурационного файла **Dr.Web MailD**.

Вкладка "Карантин"

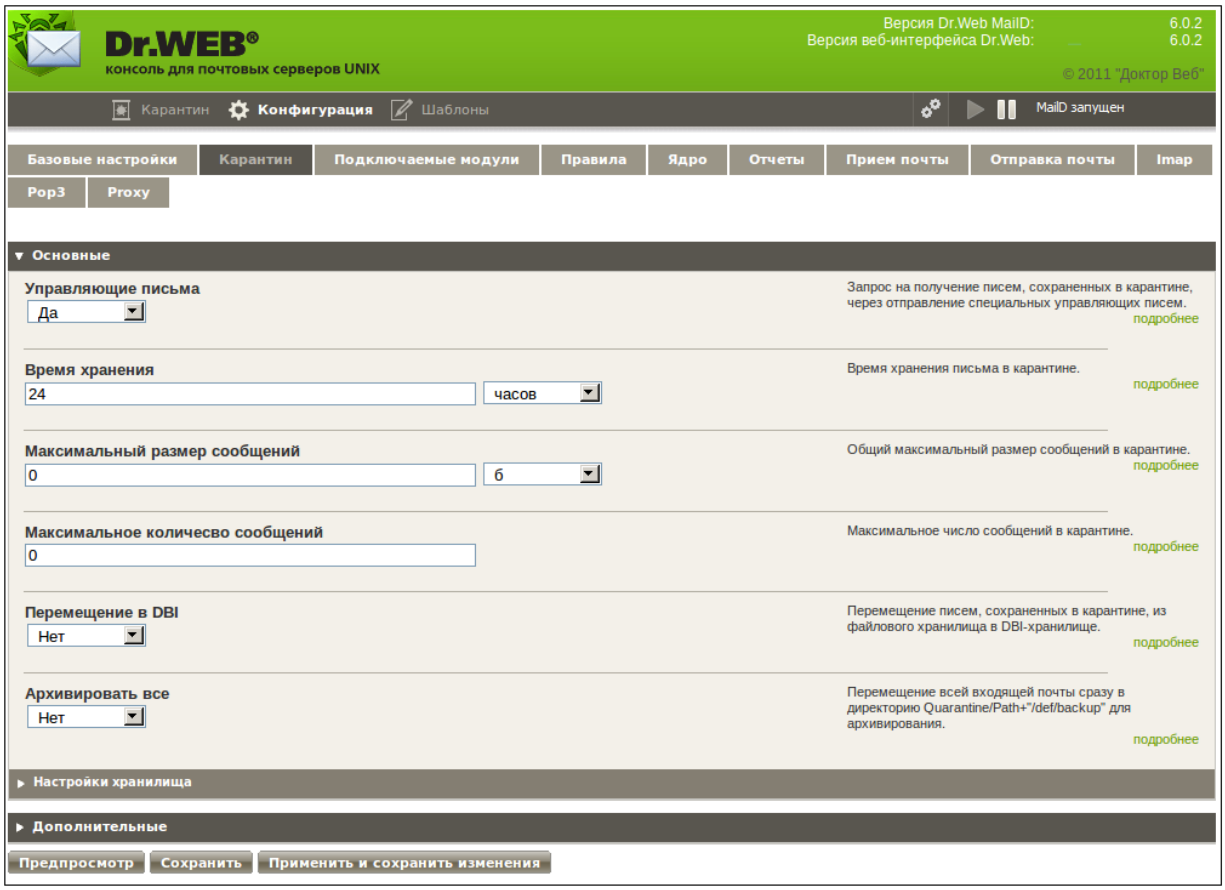


Рис. 38. Настройка Карантина

На данной вкладке вы можете управлять основными настройками секции **Карантин**: определять срок, в течение которого сообщения будут храниться в **Карантине**, устанавливать права доступа к этим сообщениям, задавать правила переименования помещаемых в **Карантин** сообщений, настраивать работу с хранилищем **DBI**.

Вкладка "Подключаемые модули"

На данной вкладке представлены общие настройки для всех подключаемых модулей, включенных в **Dr.Web MailD**. Для настройки параметров каждого конкретного модуля необходимо перейти на соответствующую вкладку.

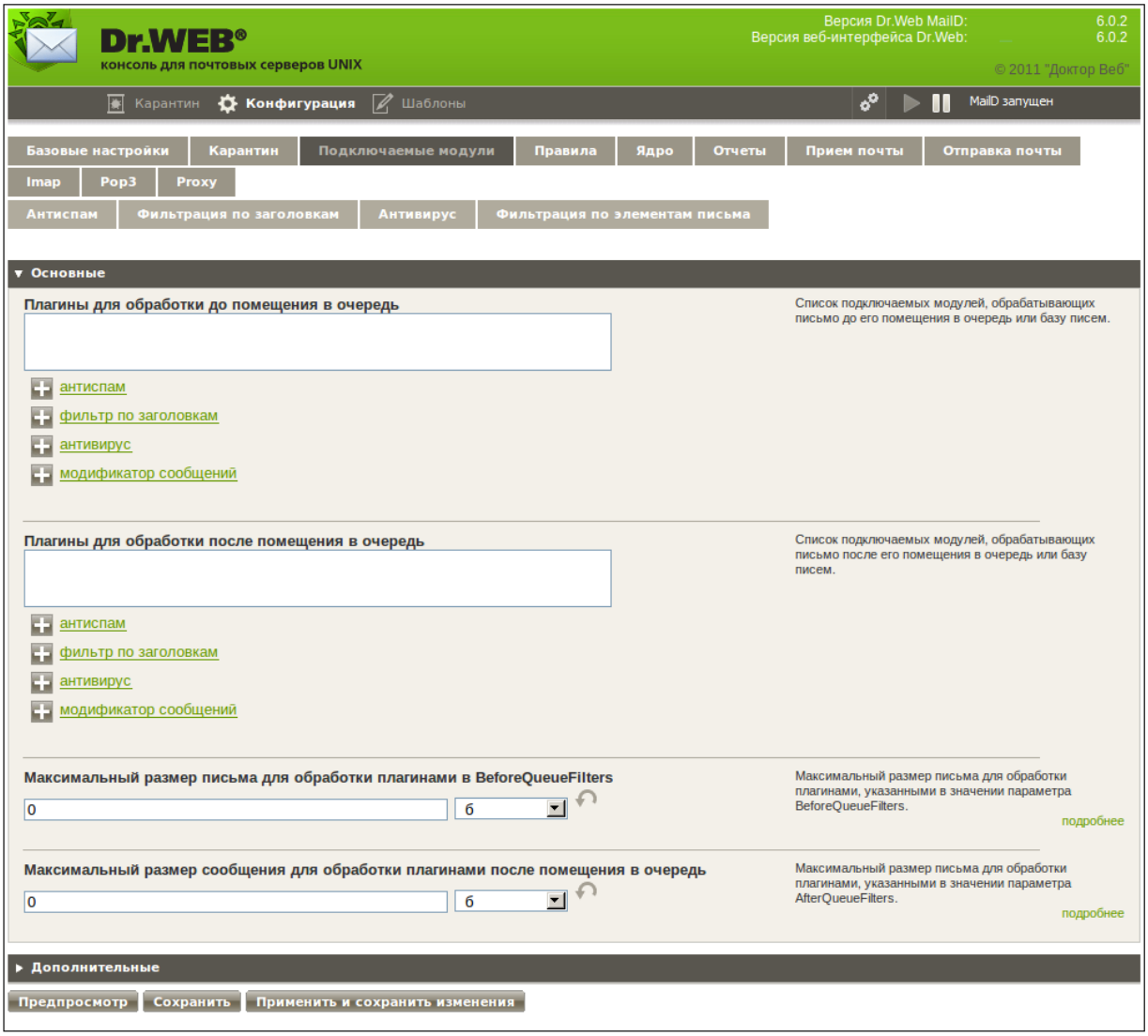


Рис. 39. Общие настройки работы подключаемых модулей

Значения дополнительных действий **`перенаправить`**, **`добавить заголовок`** и **`добавить счет`** не выделяются круглыми скобками "(" и ")". Т.е. задаётся непосредственное значение дополнительных действий:

- Для действия **`перенаправить`** вводится список адресов с разделителем "|" : `address1@domain | address2@domain | address3@domain`
- Для действия **`добавить счет`** в поле вводится только значение счета.
- Значение заголовка вводится в формате [`ИМЯ:`]`ЗНАЧЕНИЕ`, где `ИМЯ` - название заголовка (`X-DrWeb-MailD` по умолчанию), а `ЗНАЧЕНИЕ` - значение заголовка

Значение дополнительного действия **`добавить счет`** экранируется двойными кавычками при добавлении в конфигурационный файл (подробнее о экранировании см. Действия с зараженными и подозрительными объектами).

Например, при добавлении слова "Infected" в качестве заголовка для зараженных файлов во вкладке "Антивирус", в конфигурационный файл подключаемого модуля будет добавлена следующая строка:

```
Infected = cure,quarantine,notify,"add-header(infected!)"
```

Вкладка "Антиспам"

На этой вкладке осуществляется настройка антиспам-модуля **Vaderetro**, входящего в программный комплекс **Dr.Web для почтовых серверов UNIX**.

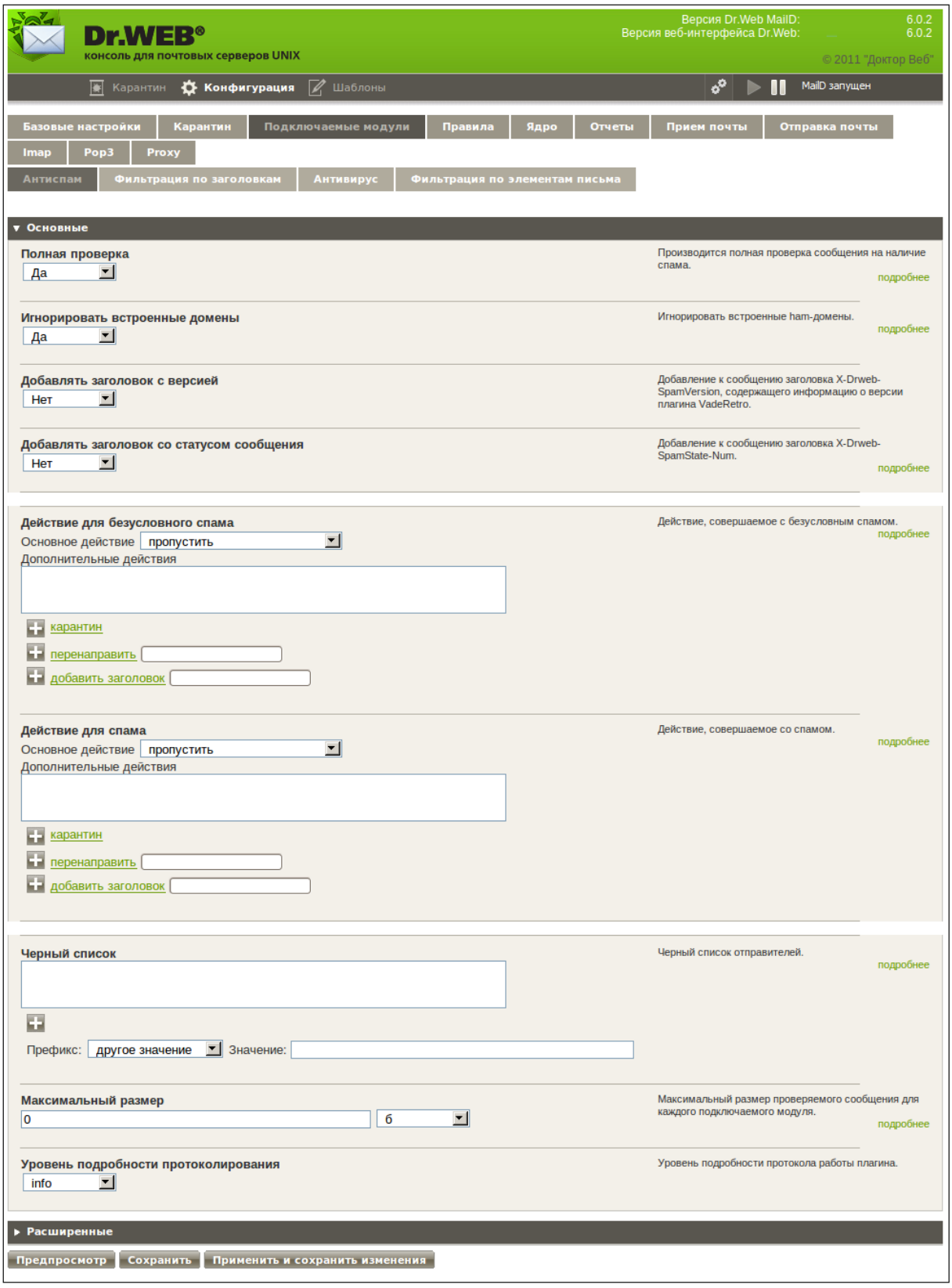


Рис. 40. Общие настройки антиспам-модуля

Вкладка "Фильтрация по заголовкам"

На данной вкладке осуществляется настройка модуля фильтрации по заголовкам **Dr.Web HeadersFilter**, позволяющего осуществлять фильтрацию почтовых сообщений на основе их заголовков.

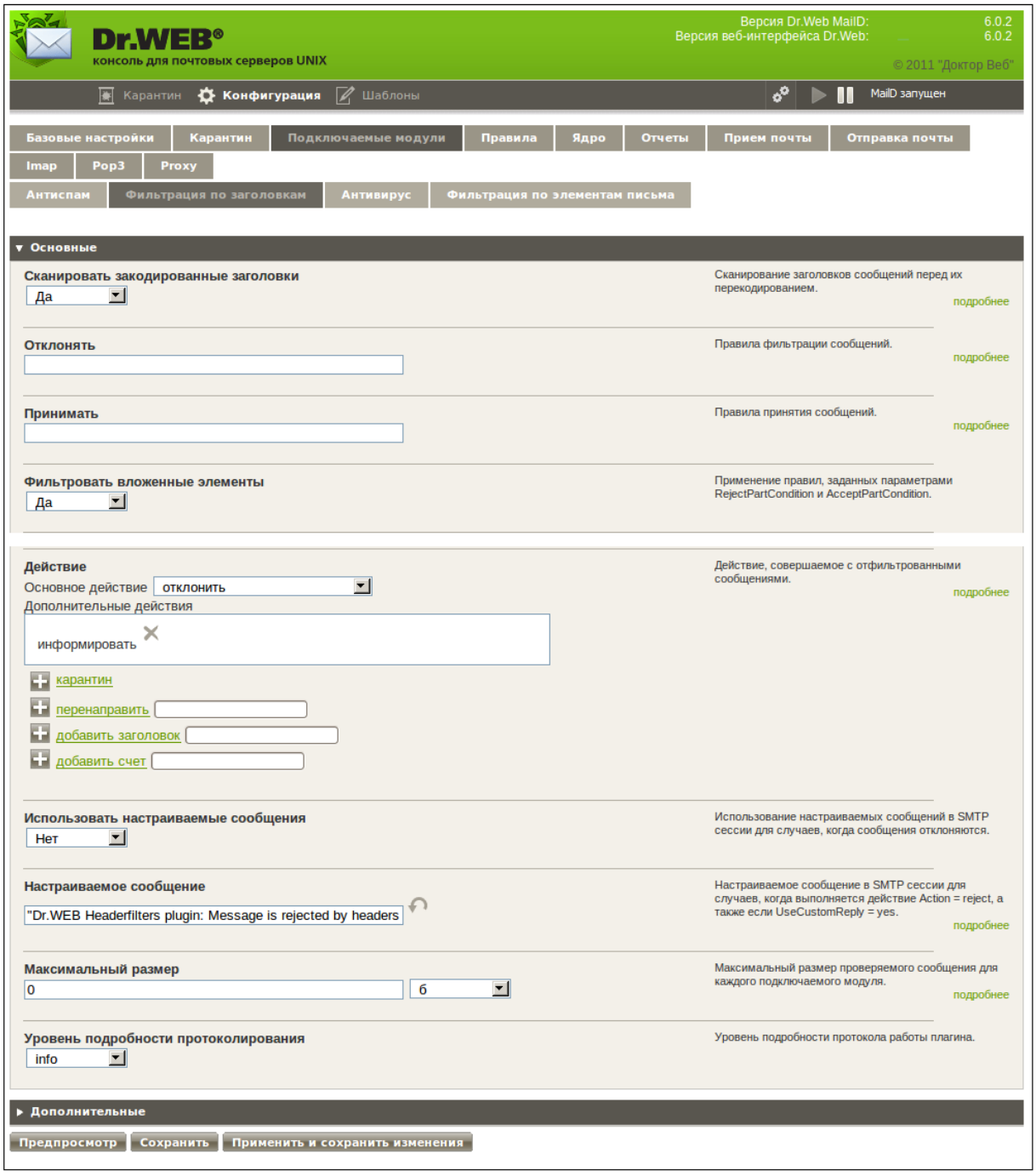


Рис. 41. Общие настройки модуля фильтрации по заголовкам

Строка "`HEADER = regular_expression`" должна быть указана полностью в поле **Значение** для всех параметров типа **`~Condition`**. В поле ввода рядом со значением **`перенаправить`** параметра **`Action`** можно указать любой адрес электронной почты, на который потом будут перенаправляться отфильтрованные сообщения.

Вкладка "Антивирус"

На этой вкладке осуществляется настройка антивирусного модуля **Drweb**, входящего в программный комплекс **Dr.Web для почтовых серверов UNIX**.

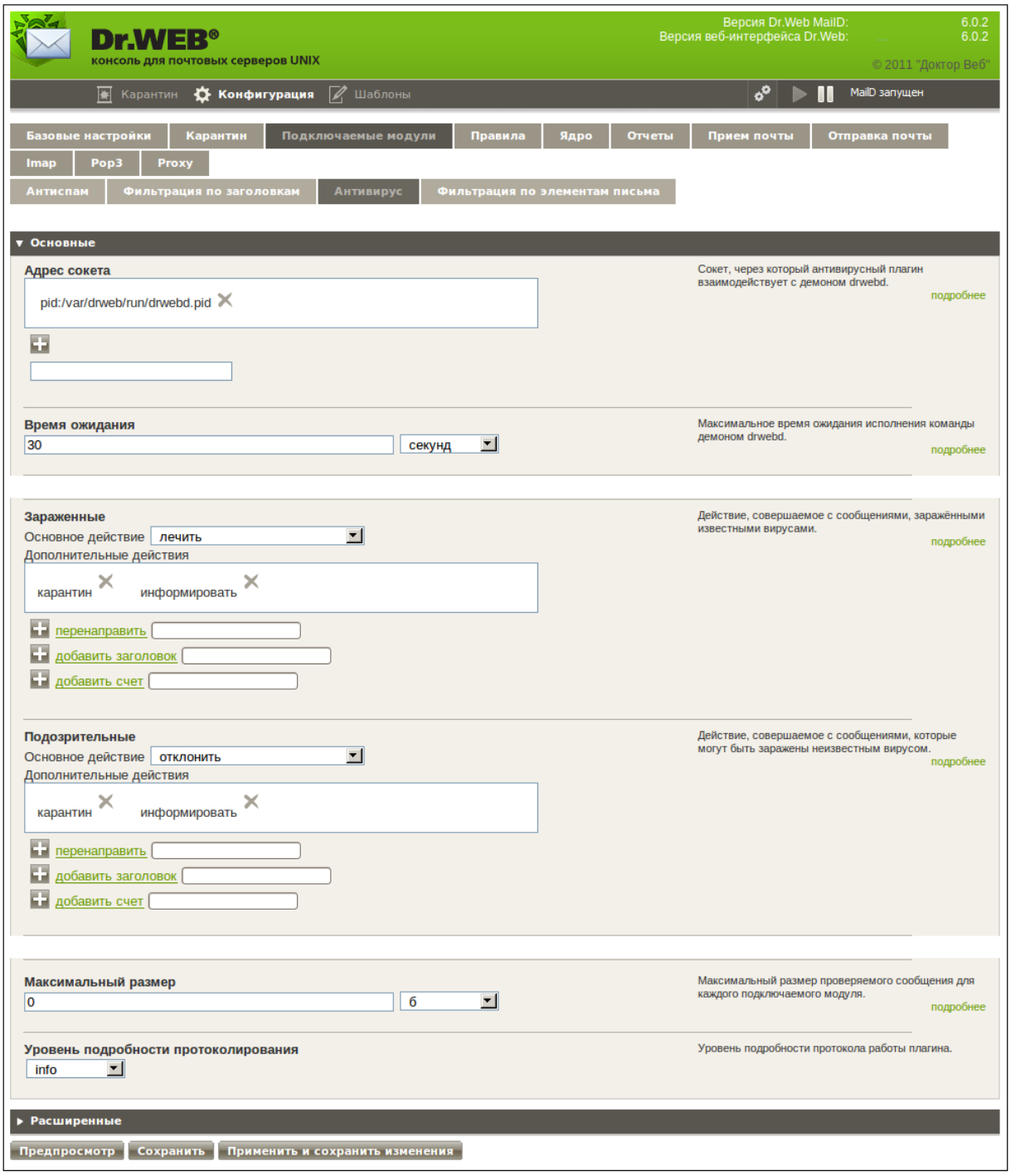


Рис. 42. Общие настройки антивирусного модуля

В поле ввода рядом со значением **`перенаправить`** любого параметра, управляющего действиями, совершаемыми над сообщениями, можно указать адрес электронной почты, на который потом будут перенаправляться отфильтрованные письма (по умолчанию используется адрес, заданный значением параметра **`RedirectMail`** во вкладке **Ядро**).

В меню расширенных настроек можно создавать тексты уведомлений, высылаемых

пользователям при блокировании письма.

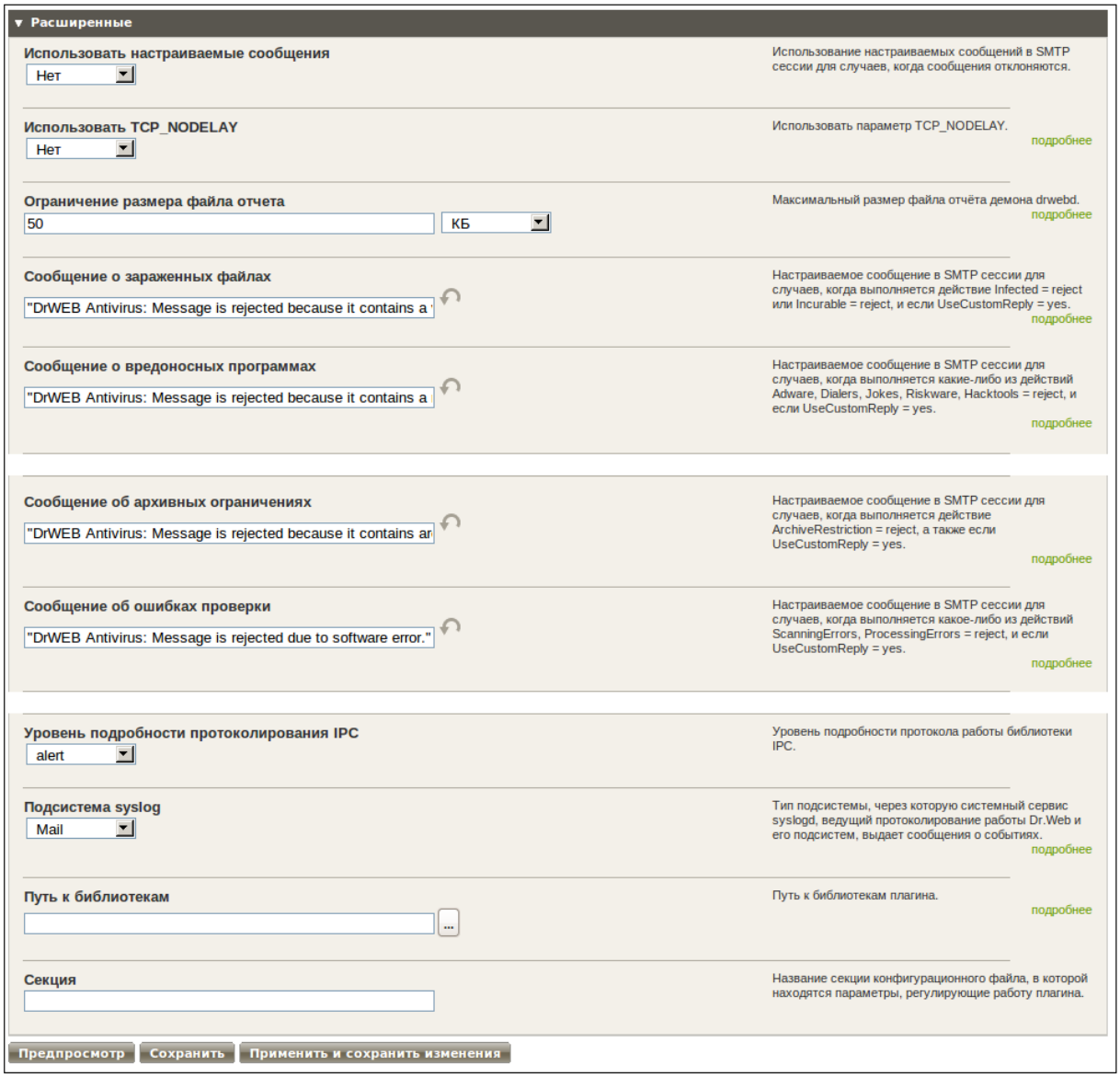


Рис. 43. Дополнительные настройки антивирусного модуля

Вкладка "Фильтрация по элементам письма"

На этой вкладке осуществляется настройка модуля фильтрации по элементам письма **Dr.Web Modifier**.

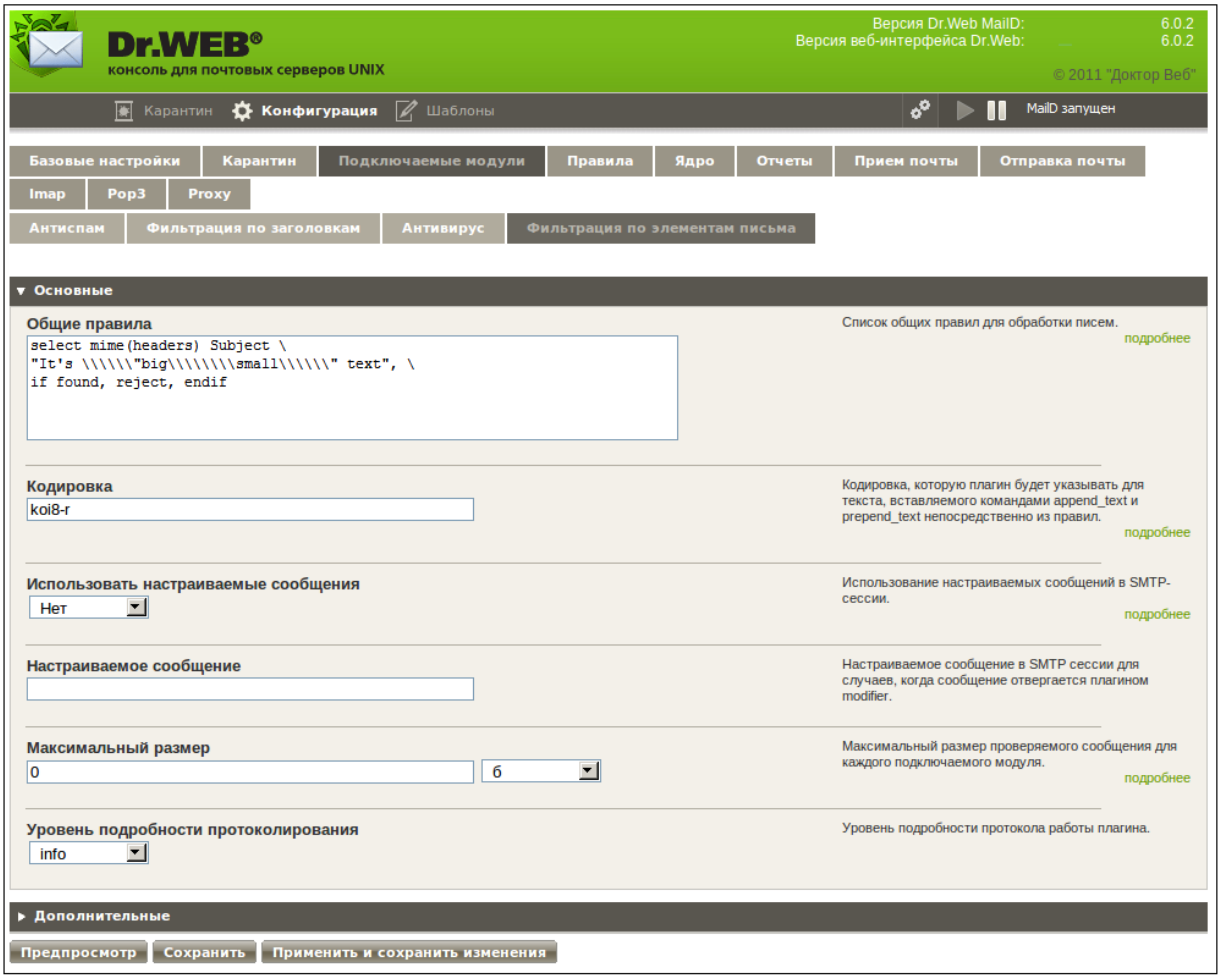


Рис. 44. Основные настройки модуля фильтрации по элементам письма

Обратите внимание, что правила модификации в поле **Общие правила** указываются так же, как и в конфигурационном файле подключаемого модуля **Dr.Web Modifier** (параметр `GlobalRules`), т.е. если внутри условий выборки будут кавычки или символы косой черты, их требуется экранировать:

- одинарная кавычка (') не экранируется;
- двойная кавычка (") экранируется 6-ю обратными косыми чертами;
- обратная косая черта экранируется 7-ю обратными обратными косыми чертами.

На картинке выше представлено правило модификации, срабатывающее, если тема письма соответствует строке `It's "big\small" text`.

Вкладка "Правила"

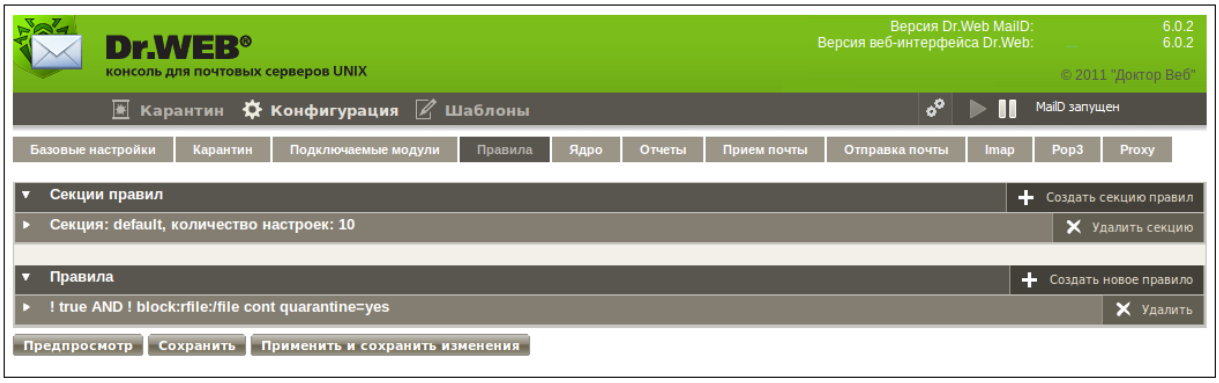


Рис. 45. Правила обработки писем

Данная вкладка содержит настройки Правил обработки писем из секции `[Rules]` конфигурационного файла **Dr.Web MailD**. С ее помощью можно создавать как Правила, так и пользовательские наборы настроек для последующего использования их в Правилах обработки писем.

Чтобы создать новое Правило, нажмите кнопку **Создать новое правило**. Чтобы отредактировать любое Правило (как только что созданное, так и старое), достаточно кликнуть на него левой кнопкой мыши.

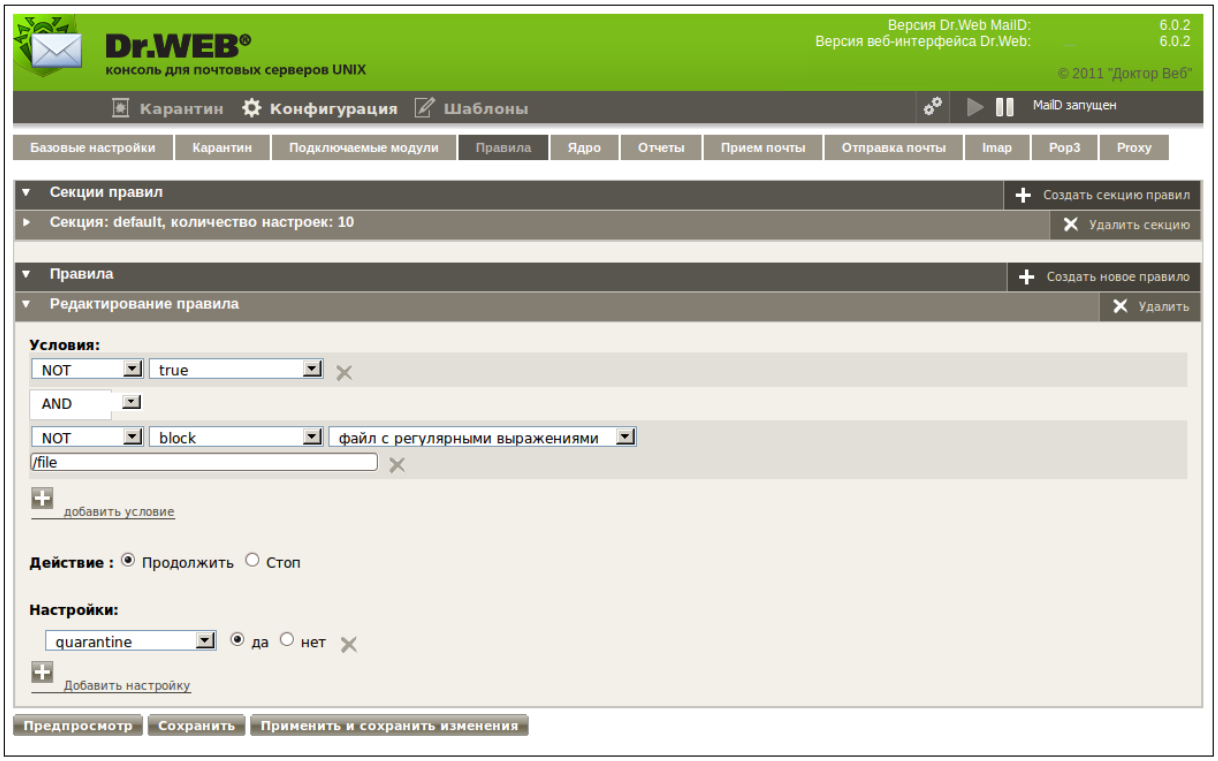


Рис. 46. Редактирование Правила обработки

При редактировании любого Правила должны быть заданы значения во всех трех разделах: **Условия**, **Действие** и **Настройки**. При составлении условий могут использоваться логические операторы.

Обратите внимание на следующие ограничения редактора Правил обработки писем, входящего в состав текущей версии **Консоли**:

1. Не поддерживаются сложные Правила, содержащие объединения.
2. Не поддерживаются Правила, локально переопределяющие для Lookup настройку параметров **SkipDomains**.

Работа с подобными Правилами возможна только посредством правки конфигурационного файла или через интерфес **Центра управления Dr.Web** (в Enterprise-режиме). Однако такие Правила обработки будут некорректно отображаться редактором Правил, входящим в состав **Консоли**.

Обратите внимание, что локальные правила фильтрации для подключаемого модуля **Dr.Web Modifier** указываются так же, как и в конфигурационном файле (параметр **modifier/LocalRules**), т.е. если внутри условий выборки будут кавычки или символы косой черты, их требуется экранировать:

- одинарная кавычка (') не экранируется;
- двойная кавычка (") экранируется 6-ю обратными косыми чертами;
- обратная косая черта экранируется 7-ю обратными обратными косыми чертами.

На картинке ниже представлен пример ввода локального правила фильтрации **modifier/LocalRules**, которое сработает, если тема письма соответствует строке **It's "big\small" text.**

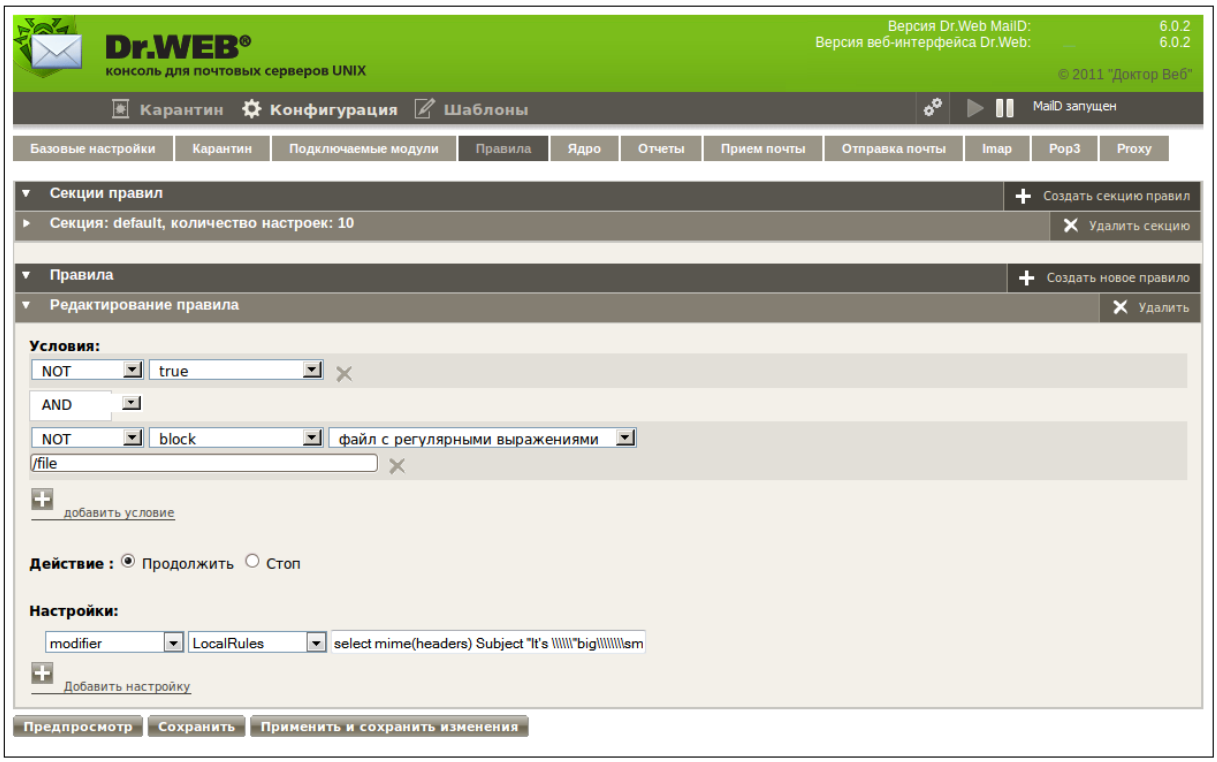


Рис. 47. Редактирование Правила обработки

Чтобы создать набор настроек для использования в Правилах обработки, нажмите кнопку **Создать секцию правил**. Чтобы отредактировать любую группу настроек (как только что созданную, так и старую), достаточно кликнуть на нее левой кнопкой мыши.

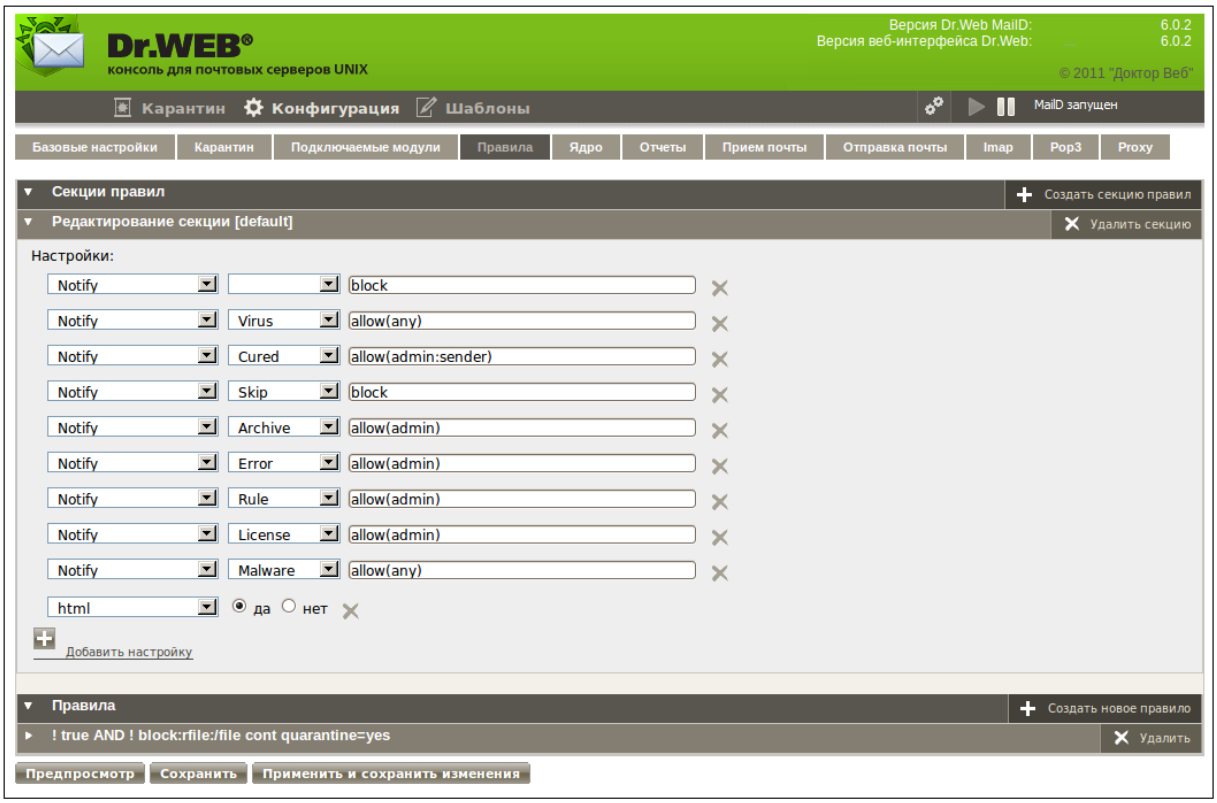


Рис. 48. Редактирование именованной секции параметров Правил

Вкладка "Ядро"

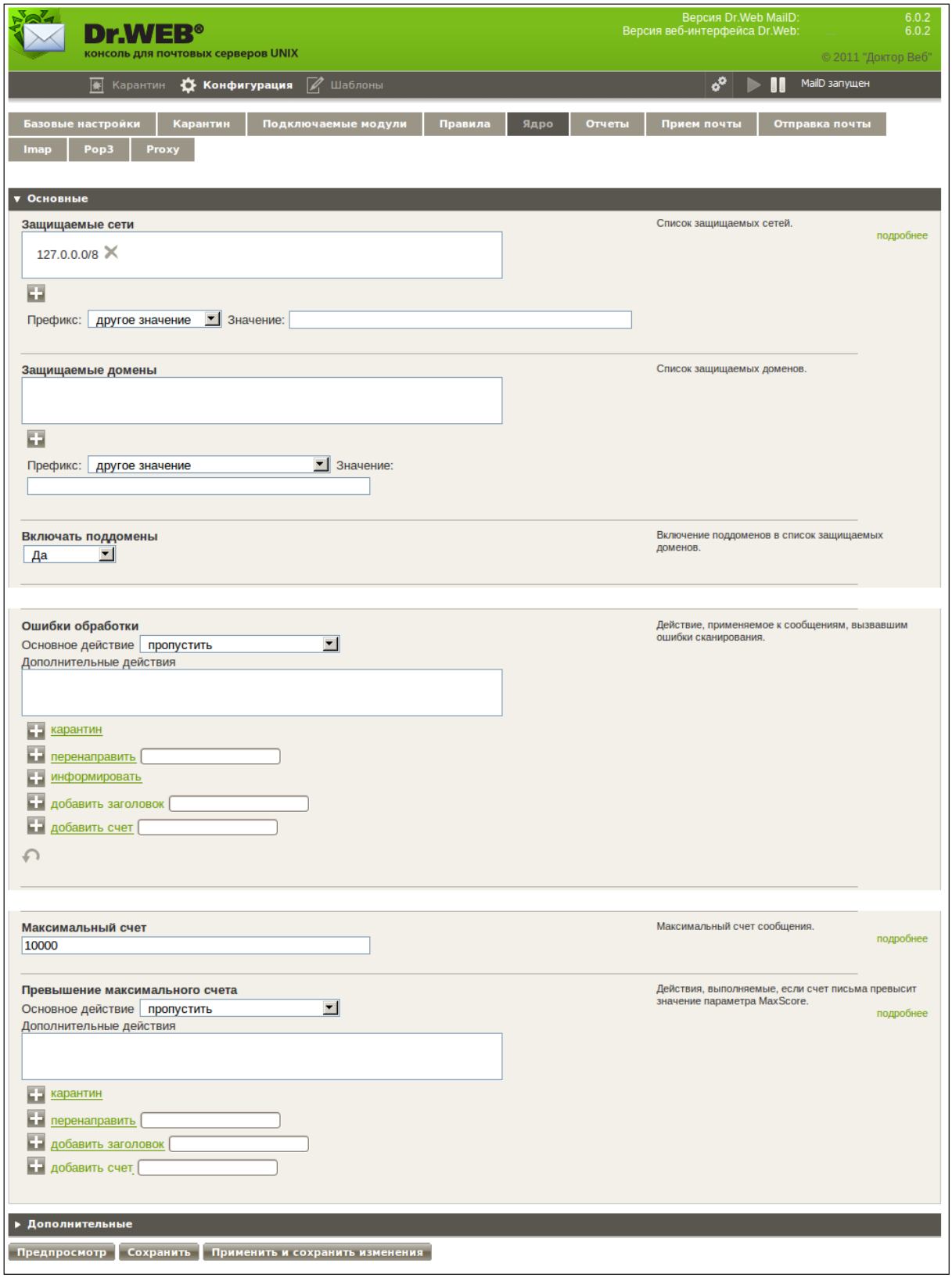


Рис. 49. Общие настройки Ядра

На этой вкладке вы можете указать адрес электронной почты, используемый по умолчанию для перенаправления писем, отфильтрованных каким-либо подключаемым модулем, при указании

значения `redirect` для соответствующих параметров. Также здесь осуществляется подключение удаленного управления модулем **`drweb-maild`** с помощью управляющих писем.

Настройки пулов потоков и сообщений, отправляемых пользователям при блокировании письма, могут быть выбраны из раскрывающихся списков или заданы вручную в соответствующих полях ввода.

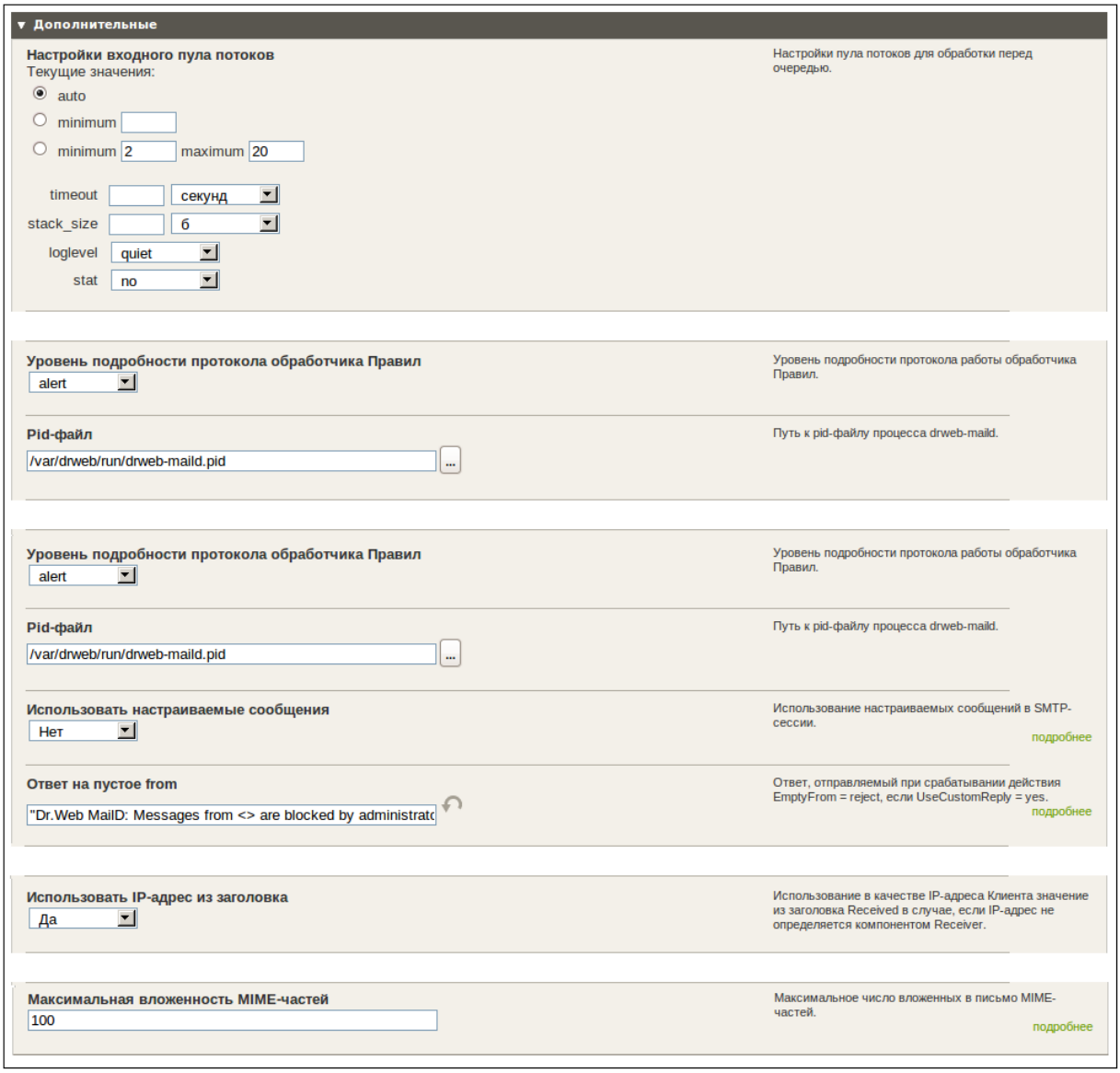


Рис. 50. Дополнительные настройки Ядра

В секции **Настройки базы писем** вы можете настроить работу с базой писем.

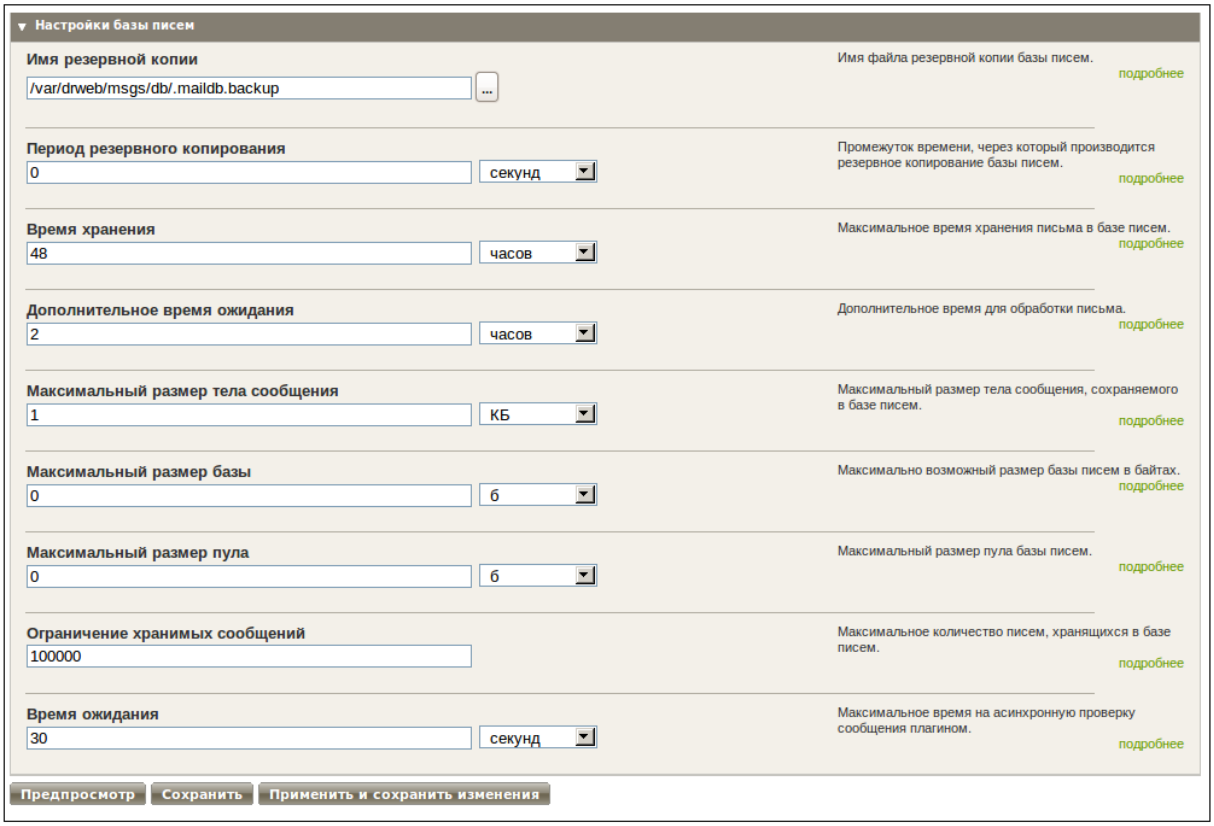


Рис. 51. Настройки секции MailBase

Вкладка "Отчеты"

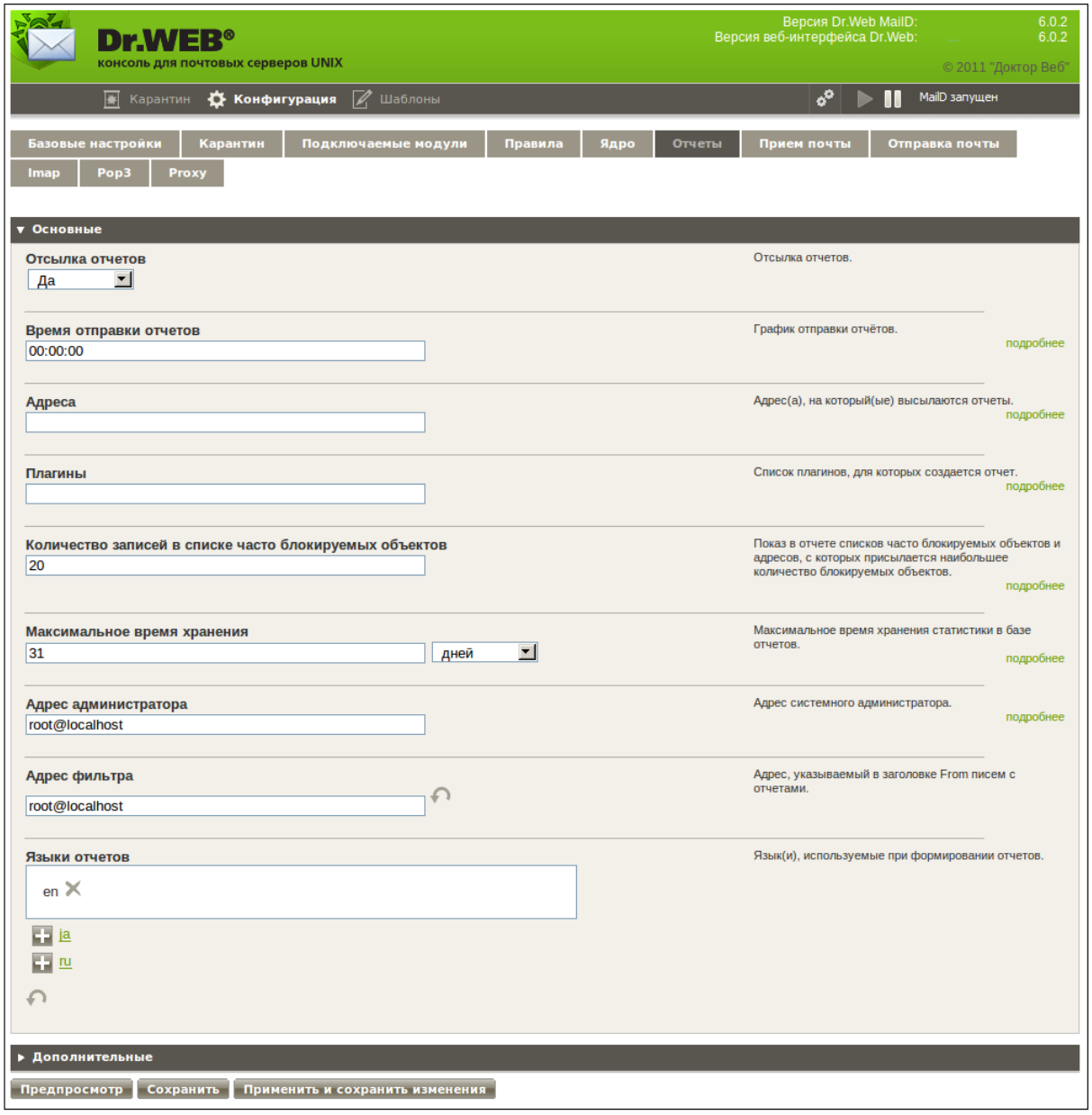


Рис. 52. Настройка отчетов

На этой вкладке вы можете настроить вид отчетов со статистикой, регулярность их отправки системному администратору и время хранения статистических данных в базе отчетов.

Вкладка "Прием почты"

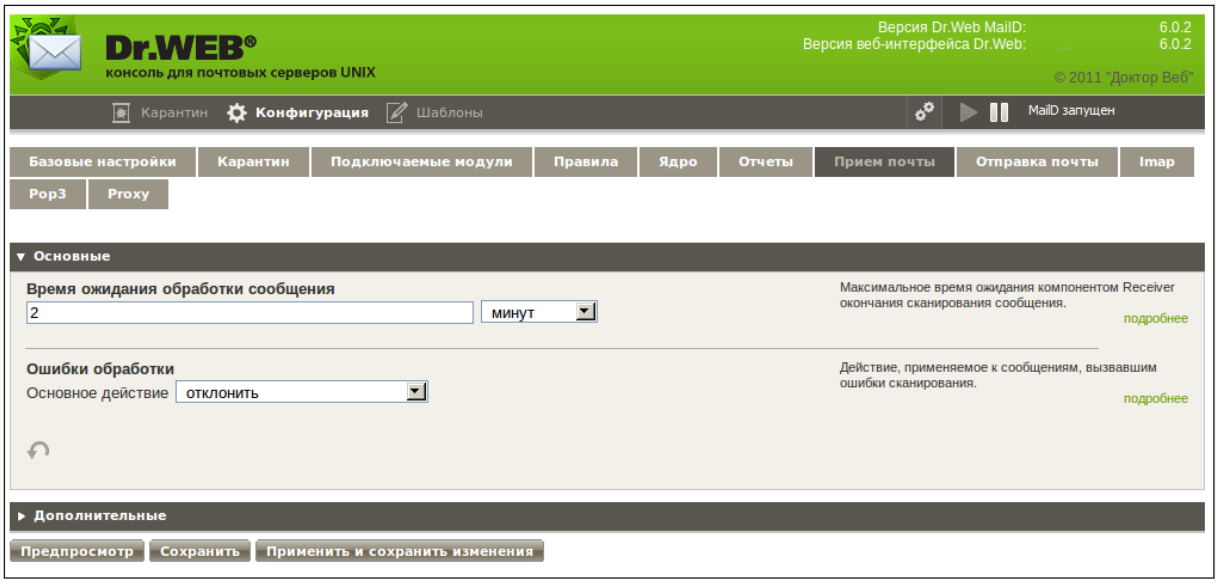


Рис. 53. Общие настройки приема почты

На данной вкладке вы можете указать один или несколько адресов для получения SMTP/LMTP-запросов, а также действие, применяемое к сообщениям, вызвавшим ошибки их обработки.

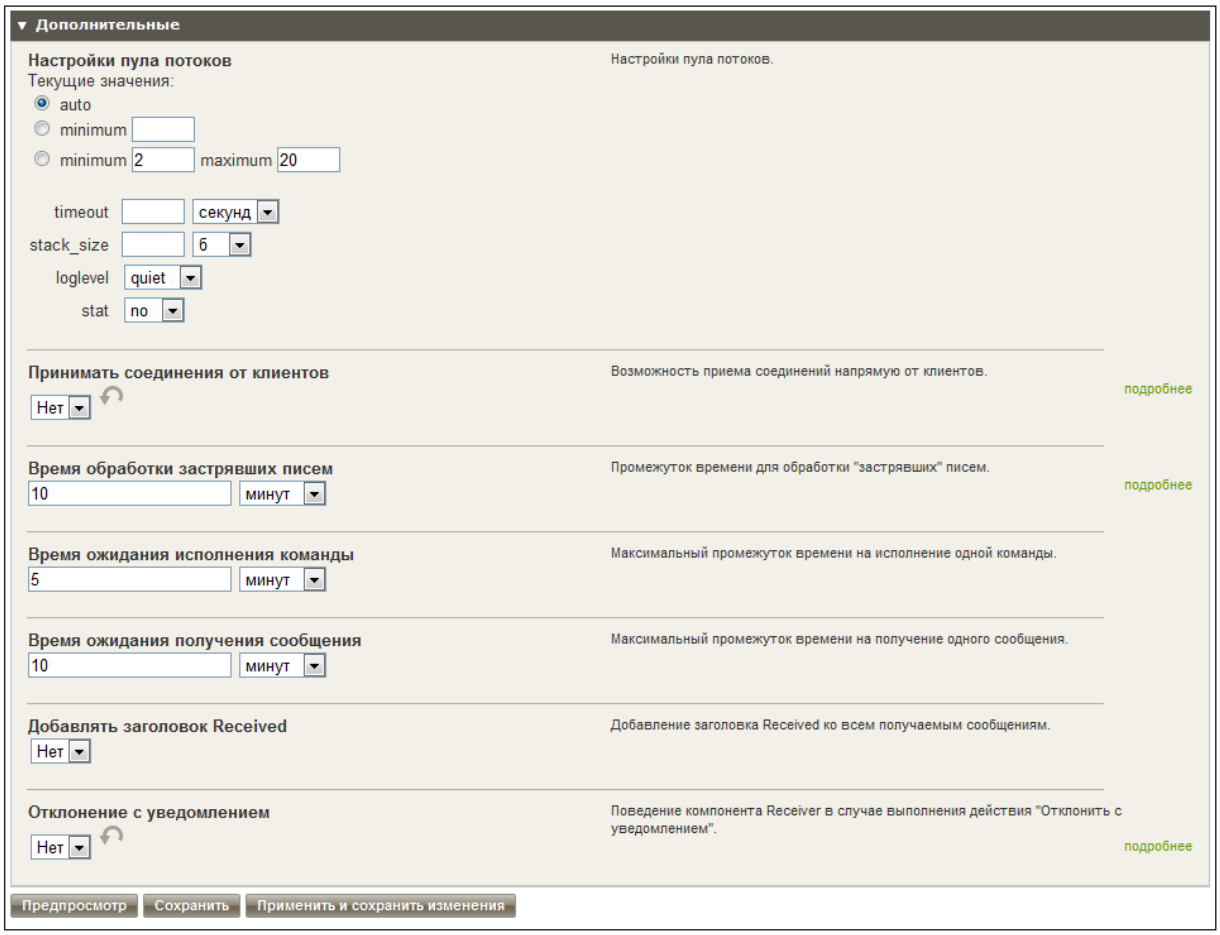


Рис. 54. Дополнительные настройки работы почтовой системы

На вкладке дополнительных настроек возможно настроить параметры взаимодействия **Dr.Web MailD** с используемой почтовой системой.

В текущей версии **Dr.Web для почтовых серверов UNIX** настройка единовременного использования нескольких компонентов **Receiver** и **Sender** посредством веб-интерфейса не поддерживается.

Вкладка "Отправка почты"

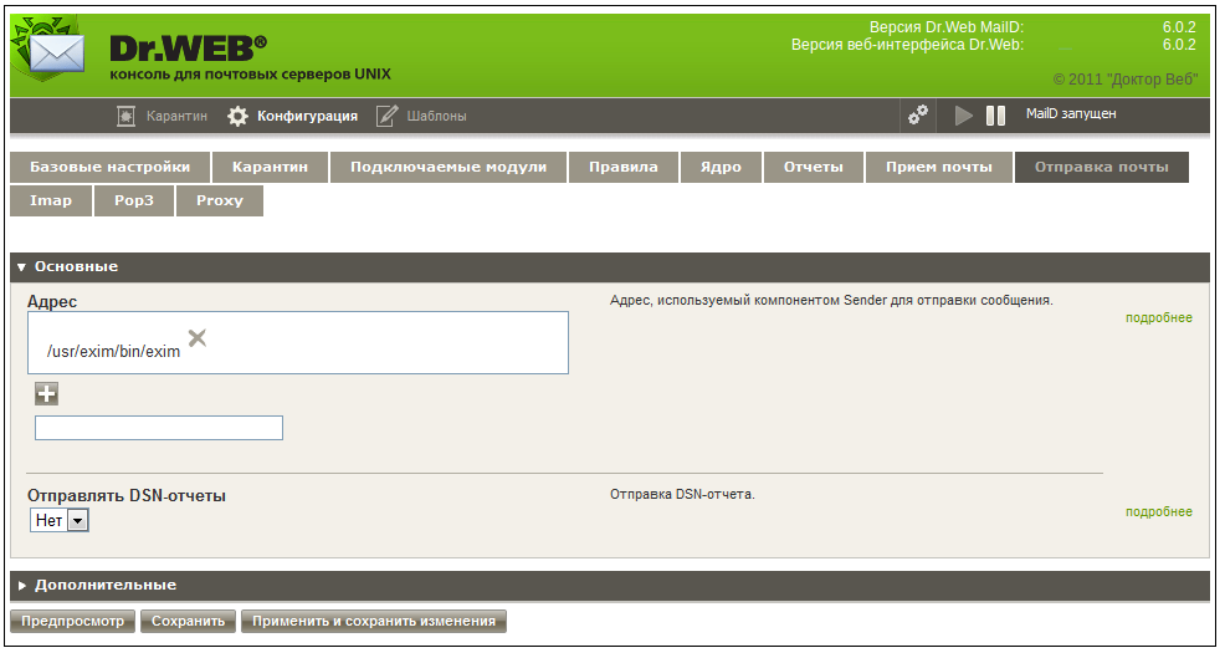


Рис. 55. Настройка отправки почты

На данной вкладке вы можете указать набор действий, которые должны быть предприняты при отправлении письма, а также установить максимальное время ожидания исполнения команд и обработки писем компонентом **Dr.Web Daemon** и подключаемыми модулями.

Обратите внимание, что веб-интерфейс **Dr.Web для почтовых серверов UNIX** не предусматривает управление "потерянными" письмами, находящимися в каталоге `/out/failed` хранилища писем (подробнее о "потерянных" письмах см. в описании параметров компонента **Sender**).

Для обнаружения таких писем необходимо проверять этот каталог периодически вручную, и при обнаружении "потерянных" писем, их можно либо явно отправить получателям при помощи утилиты **`drweb-inject`**, либо удалить средствами ОС.

Вкладка "IMAP"

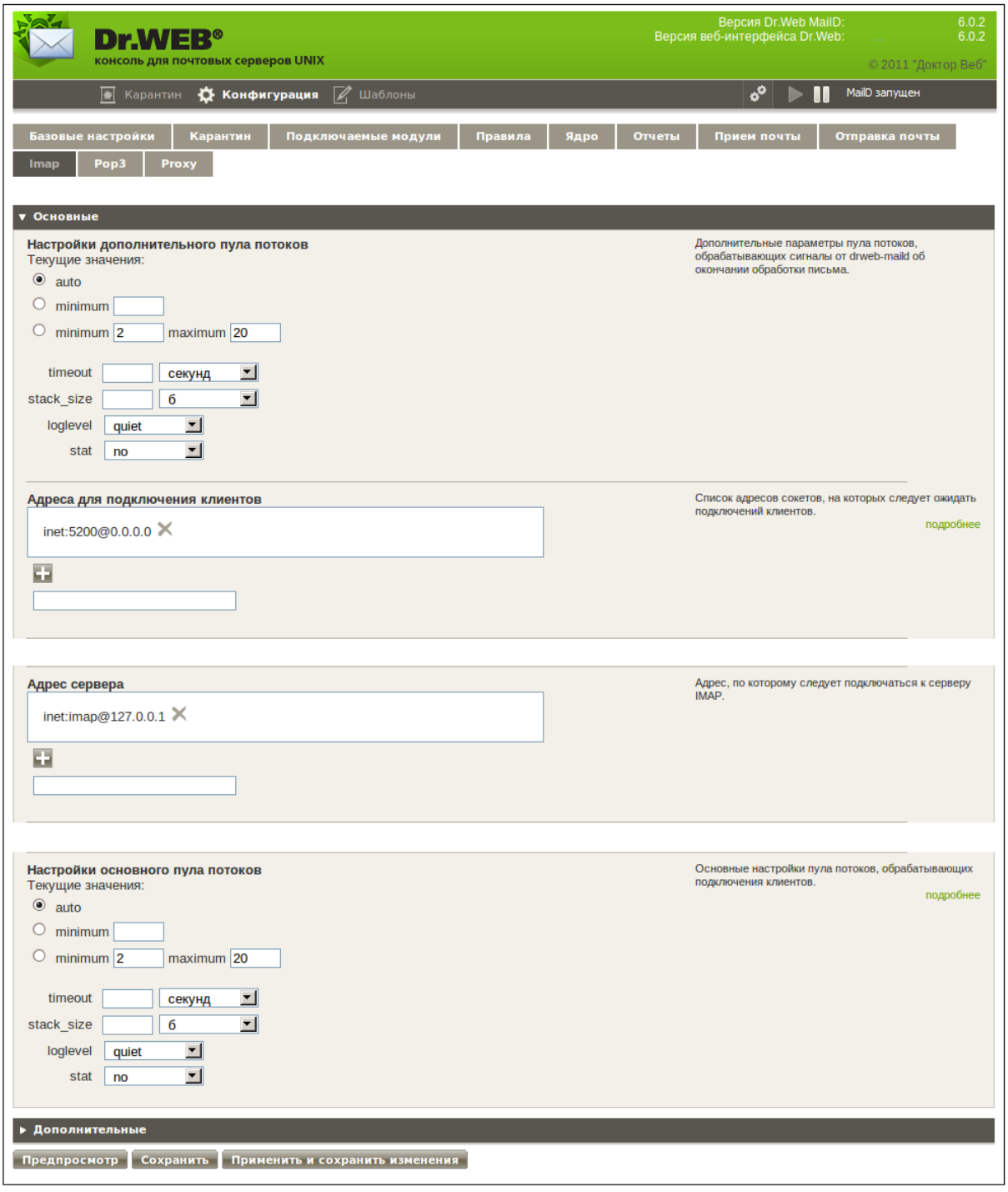


Рис. 56. Настройка фильтра IMAP

На данной вкладке вы можете указать настройки IMAP-фильтра для проверки почты по протоколу IMAP.

Вкладка "POP3"

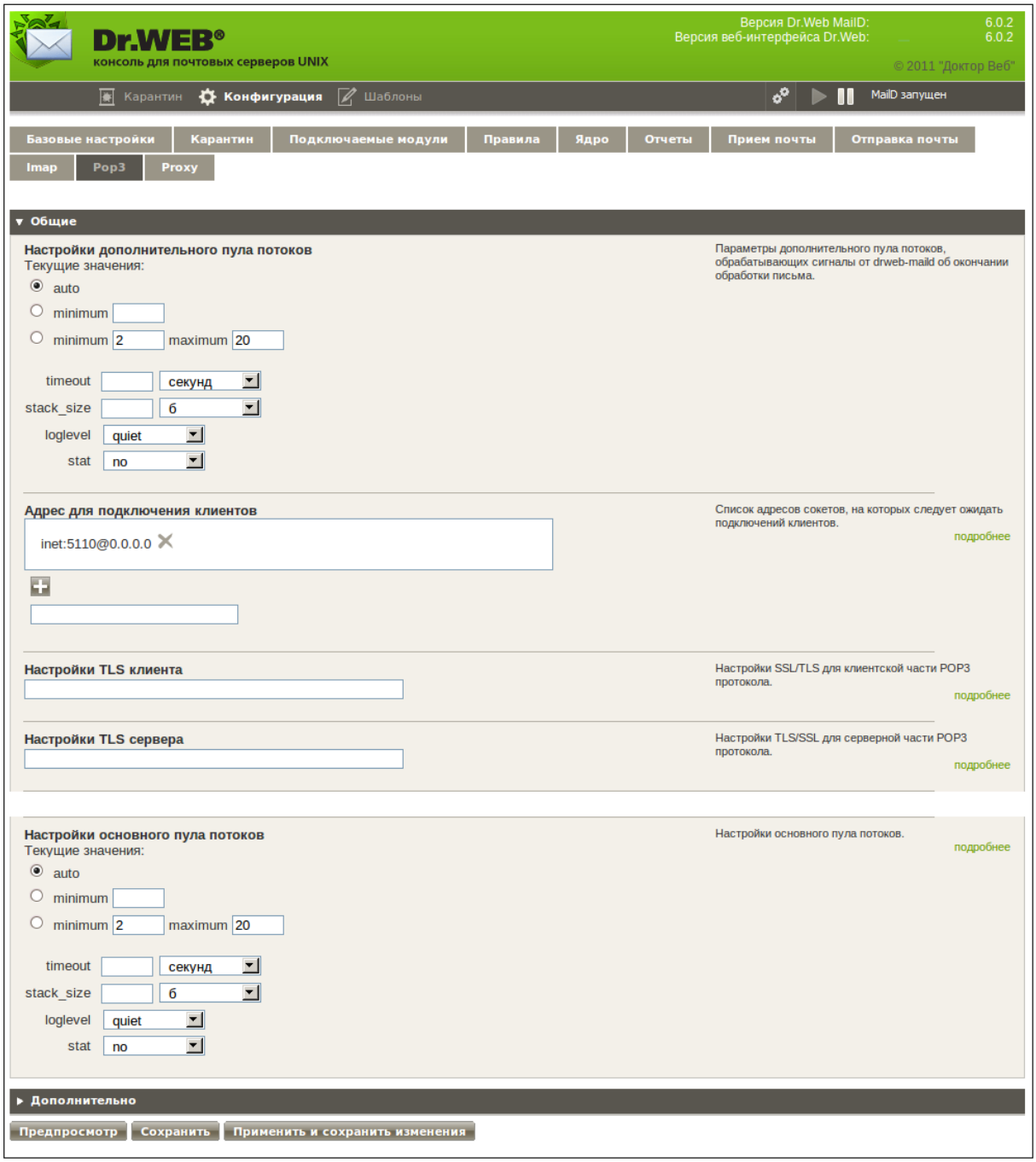


Рис. 57. Настройка фильтра POP3

На данной вкладке вы можете указать настройки POP3-фильтра для проверки почты по протоколу POP3.

Вкладка "Proxy"

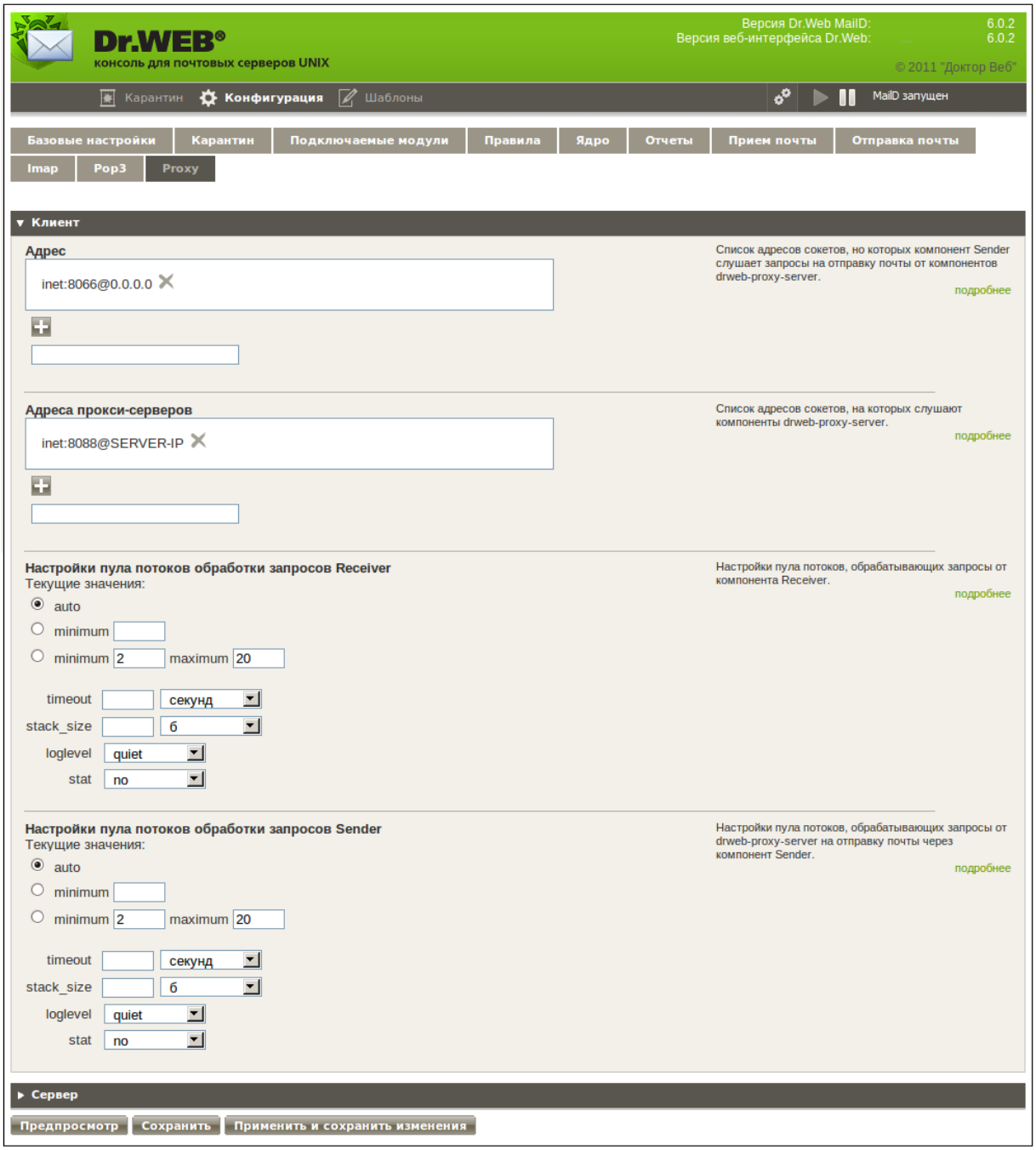


Рис. 58. Настройка прокси-сервера

На данной вкладке вы можете настроить работу прокси, позволяющего компонентами **Dr.Web для почтовых серверов UNIX**, расположенными на разных хостах, взаимодействовать между собой.

Шаблоны

В этой секции содержатся шаблоны уведомлений MailD, которые генерируются и высылаются различным типам получателей при обнаружении в письме вредоносных объектов, а также при возникновении ошибок в работе компонента **Dr.Web Daemon** или подключаемых модулей.

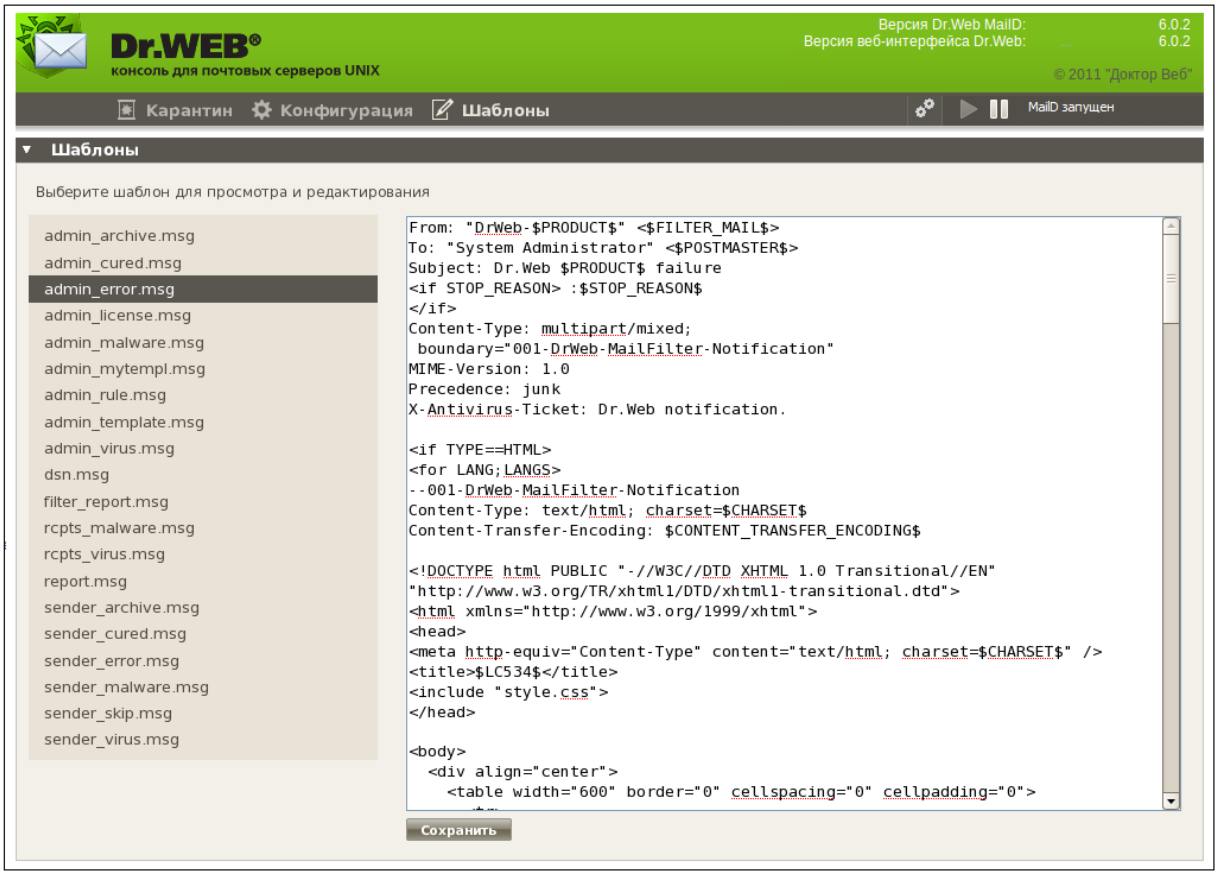


Рис. 59. Шаблоны уведомлений

Работа в Enterprise-режиме

Для начала работы **Консоли** в режиме централизованной защиты, необходимо произвести настройку **Dr.Web Agent**, описанную в соответствующем разделе. После внесения необходимых изменений откройте базовые настройки **Консоли**, нажав кнопку в верхнем меню навигации web-интерфейса. В открывшемся окне настроек установите `Yes` или `Auto` в качестве значения параметра `Central Protection Mode`.

Параметр `Central Protection Mode` может принимать 3 значения:

- `No` – в данном режиме **Консоль** работает с локальными конфигурационными файлами и не имеет доступа к конфигурации, получаемой **Dr.Web Agent** от **Dr.Web Enterprise Server**. Изменения конфигурации, внесенные в данном режиме, вступят в силу только после перевода **Dr.Web Agent** в режим `Standalone`.
- `Yes` – **Консоль** получает конфигурационные данные из сокета **Dr.Web Agent**. В случае, если при этом **Dr.Web Agent** работает в `Standalone` режиме, будет выведено предупреждение вида:
  `Ошибка соединения с Dr.Web Agent на local:%var_dir/ipc/.agent`
- `Auto` – режим работы **Консоли** выбирается в зависимости от установленного режима работы **Dr.Web Agent**.

При возникновении проблем подключения к серверу **Dr.Web Enterprise Server**, возможны следующие варианты поведения **Консоли**:

- Если при первом подключении (т.е. в случае, если вы ранее не работали с данным сервером) сервер недоступен, либо авторизация прошла неудачно, **Dr.Web Agent** завершит свою работу. В этом случае проверьте настройки и попробуйте перезапустить **Dr.Web**

Agent и **Консоль**.

- Если ранее вы уже подключались к серверу централизованной защиты, но в данный момент он недоступен (например, в случае проблем с соединением), **Dr.Web Agent** использует резервные копии конфигурационных файлов, полученных от сервера во время предыдущего подключения. Данные файлы зашифрованы и не предназначены для правки пользователем. Попытка изменить их вручную приведёт к их неработоспособности.

После перехода в режим `Enterprise` в верхнем меню навигации страницы будет отображена надпись **(CPM)** (аббревиатура от *Central Protection Mode*)**.**

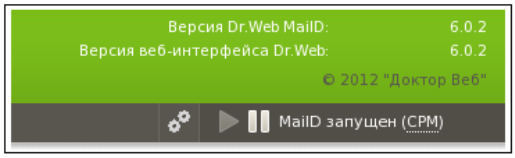


Рис. 60. Режим работы консоли Dr.Web для почтовых серверов Unix

Настройка прав доступа

При работе в режиме `Enterprise`, администратор **Центра Управления Dr.Web** может частично либо полностью заблокировать возможность настройки пользователем компонентов **Dr.Web**, установленных на рабочей станции.

Чтобы установить права пользователя рабочей станции:

- Войдите в **Центр Управления Dr.Web**. Обратите внимание, что для редактирования настроек антивирусного ПО **Dr.Web** на рабочей станции, а также редактирования прав доступа к настройкам, администратор должен обладать достаточными правами.
- Выберите пункт **Антивирусная сеть** главного меню, в открывшемся окне в иерархическом списке нажмите на название станции. В открывшемся управляющем меню (панель слева) выберите пункт **Права**. Откроется окно настройки прав.

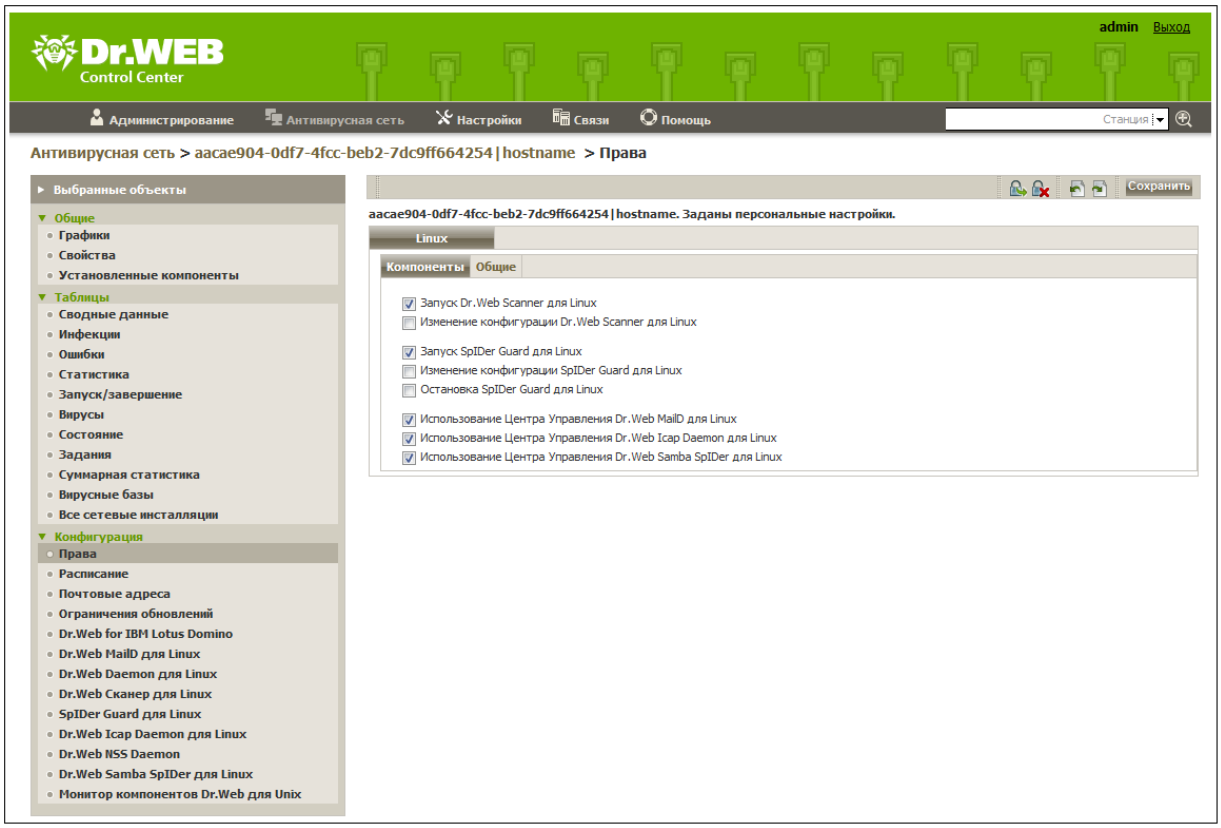


Рис. 61. Окно настройки прав пользователя рабочей станции

- В пункте **Компоненты** выберите компоненты, которые будут доступны для изменения пользователю рабочей станции. Например, чтобы разрешить изменение конфигурации **Dr.Web для почтовых серверов UNIX** пользователем рабочей станции, установите флажок **Использование Центра Управления Dr.Web MailD для Linux** и нажмите **Сохранить.**
- Чтобы отключить возможность изменения конфигурации **Dr.Web для почтовых серверов UNIX** пользователем рабочей станции, снимите флажок **Использование Центра Управления Dr.Web MailD для Linux** и нажмите кнопку **Сохранить.** При этом в окне **Консоли** пользователя рабочей станции будет выведено соответствующее предупреждение, а кнопки **Применить и сохранить изменения, Предпросмотр** и **Сохранить** блокируются.

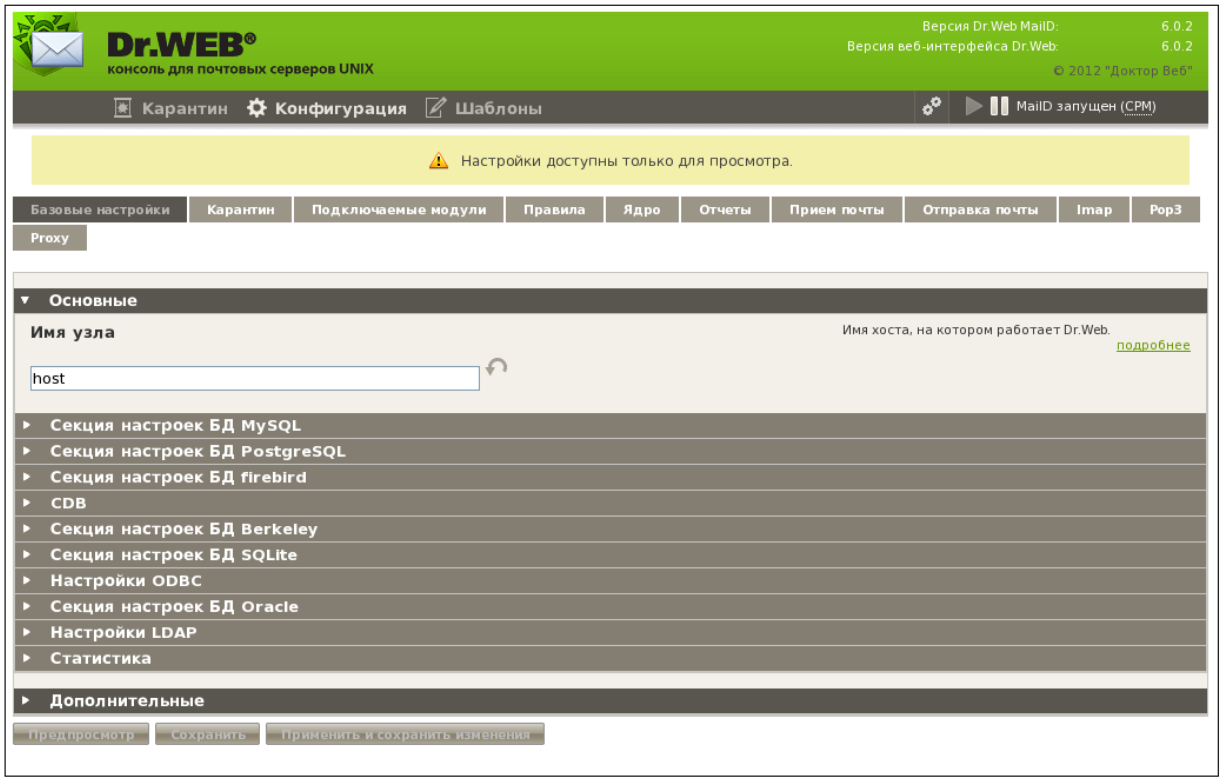


Рис. 62. Запрет на изменение конфигурации пользователем рабочей станции

Настройка конфигурации рабочей станции

При создании новой рабочей станции элементы ее конфигурации заимствуются от одной из групп, в которую она входит. Такая группа называется *первичной*. При изменениях в настройках первичной группы эти изменения наследуются входящими в группу станциями, за исключением случаев, когда станциям были заданы персональные настройки. При создании станции вы можете указать, какая из групп будет считаться первичной. По умолчанию это группа **Everyone**.

В условиях вложенных групп, если для станции не заданы персональные настройки, наследование элементов конфигурации осуществляется в соответствии со структурой вложенных групп. Поиск осуществляется вверх по иерархическому дереву, начиная с первичной группы станции, ее родительской группы и далее до корневого элемента дерева. Если при этом не были обнаружены персональные настройки, то наследуются элементы конфигурации группы **Everyone.**

Пример:

Структура иерархического списка представляет собой дерево следующего вида:

Группа `Group4` является первичной для станции `Station1`. При этом при наследовании настроек станцией `Station1` будет осуществляться поиск настроек в следующем порядке: `Station1 -> Group4 -> Group3 -> Group2 -> Group1 -> Everyone`.

Изменение конфигурации, унаследованной от первичной группы, возможно двумя способами:

- Через интерфейс **Центра Управления Dr.Web**. Для этого в интерфейсе **Центра Управления Dr.Web** выберите пункт **Антивирусная сеть** главного меню, в открывшемся окне в иерархическом списке нажмите на название станции. В открывшемся управляющем меню (панель слева) выберите компонент, который хотите настроить. Обратите внимание, что для редактирования настроек, вы должны обладать соответствующими правами. Процесс настройки аналогичен настройке посредством Консоли. После изменения настроек нажмите **Сохранить,** чтобы сохранить изменения.

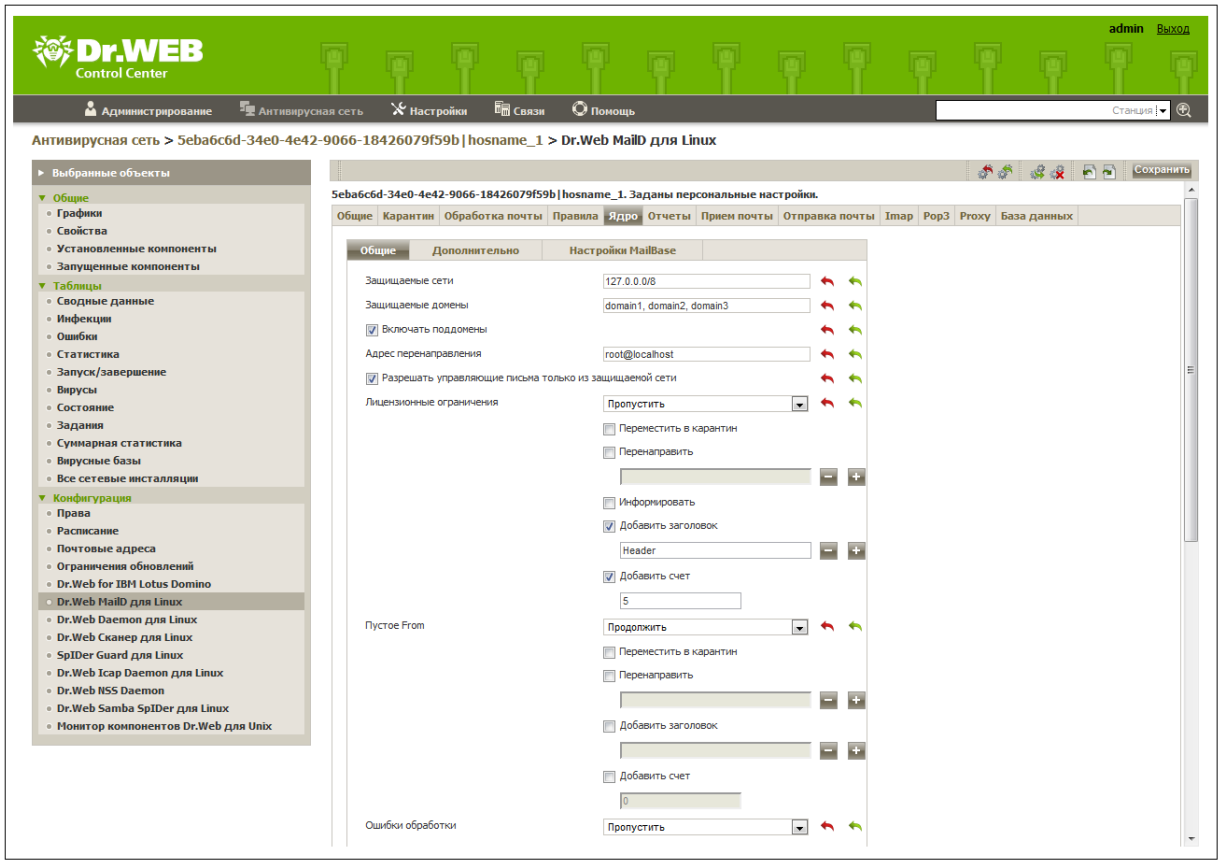


Рис. 63. Настройка Dr.Web MailD для Linux через интерфейс Центра Управления Dr.Web

При соответствующих настройках прав доступа параметры могут быть переопределены с помощью **Консоли**. Процесс настройки аналогичен работе в режиме Standalone. В случае недостатка прав у пользователя рабочей станции, **Консоль** предоставит доступ к настройкам в режиме «только для чтения».

Типы учетных записей администраторов

Учетные записи администраторов антивирусной сети делятся на 4 группы:

- *Администраторы с полными правами* имеют исключительные права на управление **Dr.Web Enterprise Server** и **Антивирусной сетью** в целом. Они могут просматривать и редактировать конфигурацию **Антивирусной сети**, а также создавать новые административные учетные записи. Администратор с такими правами также имеет полные права на управление антивирусным ПО на рабочей станции. При этом он может ограничить, вплоть до полного запрета, вмешательство пользователя рабочей станции в управление антивирусным ПО.

  Администратор с полными правами может просматривать и редактировать список имеющихся административных учетных записей.

- *Администраторы с правами "только для чтения"* могут только просматривать настройки **Антивирусной сети** в целом и отдельных ее элементов, но не менять их.
- *Администраторы групп с полными правами* имеют доступ ко всем системным группам и к тем пользовательским группам, управление которыми для них разрешено (включая вложенные). Возможно создание данных учетных записей только для пользовательских групп (подробнее см. Руководство администратора **Антивирусной сети Dr.Web® Enterprise Security Suite**). Для такого администратора в иерархическом дереве будут отображаться только те группы, к которым он имеет доступ.

  Администраторы групп не могут просматривать список имеющихся административных учетных записей.
- *Администраторы групп с правами "только для чтения"* могут обладать как полными правами для редактирования доступных им групп, так и правами "только для чтения".
- *Администраторы по умолчанию*. После установки **Dr.Web Enterprise Server** автоматически создается учетная запись **admin** - администратор с полными правами.

Таким образом, *Администраторы с полными правами* могут:

- Создавать новые и удалять имеющиеся учетные записи администраторов.
- Редактировать настройки всех администраторов **Антивирусной сети**.

Администраторы групп и *администраторы с правами "только для чтения"* могут:

- Редактировать часть настроек только своей учетной записи.

Контакты

Программный комплекс **Dr.Web для почтовых серверов UNIX** находится в постоянном развитии. Наиболее свежую информацию о его обновлениях, а также новости можно получить на сайте:

http://www.drweb.com/

Отдел продаж:

http://buy.drweb.com/

Техническая поддержка:

http://support.drweb.com/

В письме необходимо предоставить следующую дополнительную информацию, которая поможет лучше разобраться в ситуации:

- полное название и версию дистрибутива UNIX-системы;
- версии компонентов программного комплекса **Dr.Web для почтовых серверов UNIX**;
- конфигурационные файлы компонентов;
- файлы журнала компонентов и отчеты и статистику, если имеются.