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
02.03.2012

Доктор Веб, 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 9

Используемые обозначения и сокращения 12

Системные требования 13

Совместимость с дистрибутивами Linux 15

Расположение файлов пакета 16

Конфигурационные файлы 17

Установка и удаление Dr.Web для почтовых серверов UNIX 31

Установка универсального пакета для UNIX систем 32

Пользовательский интерфейс графического инсталлятора 38

Использование консольного инсталлятора 45

Удаление универсального пакета для UNIX систем 48

Пользовательский интерфейс графического деинсталлятора 49

Использование консольного деинсталлятора 52

Установка из нативных пакетов 54

Скрипты настройки 62

Запуск Dr.Web для почтовых серверов UNIX 64

Для Linux и Solaris 64

Для FreeBSD 67

Операционная система с SELinux 68

Регистрация продукта 72

Модуль обновления Dr.Web Updater 77
Обновление антивируса и вирусных баз 77
Настройка cron 80
Параметры командной строки 81
Блокирование обновлений для компонентов 82
Восстановление компонентов 83
Настройки 84
Процедура обновления 90
Консольный сканер Dr.Web Scanner 92
Параметры командной строки 92
Настройки 100
Запуск 114
Антивирусный модуль Dr.Web Daemon 117
Параметры командной строки 117
Запуск 118
Проверка работоспособности Dr.Web Daemon 120
Режимы проверки 124
Обрабатываемые сигналы 126
Файл отчета 126
Настройки 129
Dr.Web Control Agent 148
Режимы работы 149
Параметры командной строки 152
Конфигурационный файл 154
Секция [Logging] 154
Секция [Agent] 155

Секция [Server] 157
Секция [EnterpriseMode] 158
Секция [StandaloneMode] 160
Секция [Update] 161
Запуск 162
Взаимодействие с компонентами программного комплекса 164
Интеграция с Dr.Web Enterprise Security Suite 166
Настройка компонентов для работы в режиме Enterprise 166
Автоматическое создание учетной записи 167
Создание учетной записи на сервере вручную 168
Задание конфигурации компонентов через веб-интерфейс сервера 169
Экспорт существующей конфигурации на сервер 169
Запуск комплекса 169
Работа с вирусной статистикой 170
Dr.Web Monitor 176
Режимы работы 176
Параметры командной строки 179
Конфигурационный файл 179
Секция [Logging] 179
Секция [Monitor] 181
Запуск 185
Взаимодействие с компонентами программного комплекса 186
Dr.Web для почтовых серверов UNIX 189
Параметры командной строки 192

Обрабатываемые сигналы **193**

Внутренняя статистика работы **194**

Настройка и запуск **197**

Конфигурационный файл **197**

Lookups **342**

Статистика **353**

Карантин **357**

Интерактивное управление **366**

Использование drweb-inject для отправки писем **398**

Одновременное подключение нескольких компонентов Receiver/Sender **399**

Unified Score **404**

Reputation IP Filter **406**

Плагины **413**

Антивирусный плагин drweb **413**

Плагин headersfilter **428**

Антиспам плагин vaderetro **434**

Плагин Dr.Web Modifier **446**

Интеграция с почтовыми системами **467**

Работа в режиме SMTP-proxy **468**

Интеграция с почтовой системой CommuniGate Pro **469**

Интеграция с почтовой системой Sendmail **473**

Интеграция с почтовой системой Postfix **482**

Интеграция с почтовой системой Exim **488**

Интеграция с почтовой системой Qmail **497**

Интеграция с почтовой системой Courier **502**

Интеграция с почтовой системой ZMailer **504**

Использование прокси 509

Консоль Dr.Web для почтовых серверов UNIX 517

Установка 519

Настройка 522

Пользовательский интерфейс 523

Карантин 525

Конфигурация 532

Шаблоны 554

Работа в Enterprise режиме 556

Настройка прав доступа 558

Настройка конфигурации рабочей станции 559

Типы учетных записей администраторов 562

Контакты 564

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. Также доступны два SDK:

- SDK, предоставляющий инструментарий для разработки новых модулей, выполняющих функции компонентов **Receiver**/**Sender** (подробнее см. ниже), для осуществления поддержки новых MTA.
- SDK, предоставляющий средства для создания новых подключаемых модулей, обрабатывающих почтовые сообщения.

В программном комплексе **Dr.Web для почтовых серверов UNIX**:

- консольный сканер **Dr.Web Scanner** (**Сканер**) служит для обнаружения и лечения вирусов на локальной машине, в том числе и в директориях общего доступа;
- резидентный компонент **Dr.Web Daemon** (**Демон**) используется в качестве подключаемого внешнего антивирусного фильтра. Плагин vaderetro используется в качестве внешнего спам-фильтра;

- резидентный модуль **Dr.Web Monitor** (**Монитор**) используется для запуска и перезапуска прочих модулей **Dr.Web** в нужном порядке;
- резидентный модуль **Dr.Web Control Agent** (**Агент**) используется для управления конфигурацией модулей **Dr.Web**, сбора статистической информации и интеграции с **Dr.Web Enterprise Security Suite**;
- Perl-скрипт **Dr.Web Updater** используется для автоматического обновления вирусных баз данных;
- модуль **Dr.Web MailD** осуществляет анализ и обработку почтового трафика и позволяет интегрировать все компоненты программного комплекса с почтовыми системами Sendmail, Postfix, Courier, Qmail, CommuniGate Pro, ZMailer, Exim. **Dr.Web MailD** также может работать в составе антивирусной сети под управлением **Dr.Web Enterprise Security Suite**.

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 программного комплекса **Dr.Web для почтовых серверов UNIX**.

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

Необходимо отметить, что продукты **"Доктор Веб"** 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 **"Доктор Веб"**. 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 **Доктор Веб** 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.

Обозначение	Комментарий
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 директорий, 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 директории:

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/
```

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.

Пример:

Для поддержки 32-битных приложений в системах на основе Debian/Ubuntu Linux OS понадобится установить библиотеку `ia32-libs`, а для систем на основе ALT Linux - библиотеку `i586-glibc-core`.

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`

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 x86 и x86-64 дистрибутивы Linux.

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

- универсальный пакет для UNIX систем (Linux x86) - версия ядра 2.4.x, версия glibc2.2 (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 glibc версии 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:

- ALT Linux версий 4 - 6 (32-бит), версии 5-6 (64-бит);
- Arch Linux (64-бит);
- ASPLinux версий 12 - 14 (32-бит);
- Debian версий 3.1 - 6 (32-бит), версий 4-6 (64-бит);
- Fedora 14 (64-бит);
- Gentoo;
- Mandriva Linux старше версии 2009, CS4 (32-бит), 2010.x (64-бит);
- Mandrake 10;
- openSUSE версий 10.3-11 (32/64 бит);
- PCLinuxOS 2010;
- Red Hat Enterprise Linux (RHEL) версий 4 - 6 (32 - бит), версий 5 - 6 (64-бит);
- Suse Linux Enterprise Server версий 9-11 (32 - бит), версии 10-11 (64-бит);

- Ubuntu версий 7.04 - 11.04;

Совместимость с МСВС

Дистрибутив совместим со следующими версиями операционной системы МСВС:

- МСВС 3.0 80001-12 изм. 0;
- МСВС 3.0 80001-12 изм. 1;
- МСВС 3.0 80001-12 изм. 2;
- МСВС 3.0 80001-12 изм. 3;
- МСВС 3.0 80001-14 изм. 0;
- МСВС 3.0 80001-14 изм. 1;
- МСВС 3.0 80001-14 изм. 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 пакета

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

- `%bin_dir` — исполняемые модули программного комплекса и модуль обновления компонентов **Dr.Web Updater** (perl-скрипт `update.pl`);
- `%bin_dir/lib/` — антивирусное ядро в виде подгружаемой библиотеки (`drweb32.dll`). В той же поддиректории могут располагаться различные служебные библиотеки, необходимые для работы компонентов программного комплекса;

- %etc_dir/agent/ — дополнительные конфигурационные файлы модуля **Dr.Web Agent**;
- %etc_dir/monitor/ — дополнительные конфигурационные файлы модуля **Dr.Web Monitor**;
- %var_dir/bases/*.vdb — базы данных известных вирусов;
- %etc_dir — конфигурационные файлы программного комплекса: drweb32.ini, agent.conf, monitor.conf, drwebd.enable и drweb-monitor.enable (последние два - для настройки работы демонов);
- %bin_dir/lib/ru_scanner.dwl — файл языковых ресурсов модуля **Dr.Web Scanner**;
- %bin_dir/web/ — модуль веб-интерфейса **Dr.Web для почтовых серверов UNIX** для подключения к Webmin;
- %bin_dir/scripts/ — дополнительные скрипты, скрипт автоконфигурации **Dr.Web для почтовых серверов UNIX**, скрипт миграции для переноса конфигурационных параметров со старых версий продуктов **Dr.Web**;
- %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** или подключаемых модулей.
- %bin_dir/doc/ — документация. Вся документация представлена в виде текстовых файлов и присутствует в двух вариантах — англоязычном и русскоязычном (в кодировках KOI8-R и UTF-8);
- %var_dir/infected/ — карантин для перемещения в него зараженных файлов, если такая реакция компонентов программного комплекса на обнаружение зараженных или подозрительных файлов задана в настройках.

Конфигурационные файлы

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), построенными по следующему принципу:

```
--- начало файла ---
[Имя секции 1]
Параметр1 = значение1, ..., значениеK
...
ПараметрM = значение1, ..., значениеK
...
[Имя секции X]
Параметр1 = значение1, ..., значениеK
...
ПараметрY = значение1, ..., значениеK
--- конец файла ---
```

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.

Если какой-либо параметр не задан, это не означает, что у данного параметра нет значения. В таких случаях берется заданное в коде программы значение по умолчанию. Лишь некоторые параметры являются необязательными или не имеют значений по умолчанию, о чем, как правило, упоминается отдельно.

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 отчета

выводится соответствующее предупреждение.

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 описании параметра возможность существования нескольких значений указывается явно.

Примеры:

Перечисление нескольких значений через запятую:

```
Names = XXXXX, YYYYY
```

Задание тех же значений параметра в разных строках конфигурационного файла:

```
Names = XXXXX
Names = YYYYY
```

В данном руководстве параметры описываются следующим образом:

ИмяПараметра = {Тип параметра | возможные значения параметра}

Описание параметра.

{Может ли иметь несколько значений}.

Значение по умолчанию:

ИмяПараметра = {значение | отсутствует}

Описание параметров 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:

- численное значение (numerical value) — значение параметра является целым положительным 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 числом, обозначающим права доступа к файлам. Право чтения (r) обозначается числом 4, право записи (w) обозначается числом 2, право исполнения (x) обозначается числом 1 - при задании прав эти числа суммируются для каждой категории пользователей (владельца файла, группы владельцев файла и всех остальных, не являющихся ни владельцами, ни членами соответствующей группы).

  Примеры: 755 (-rwxr-xr-x), 644 (-rw-r--r--)
- путь к файлу/директории (path to file/directory) — параметр задает расположение файла или директории в файловой системе.
- действия (actions) — действия, совершаемые над объектами, вызвавшими какую-либо реакцию компонентов программного комплекса **Dr.Web для почтовых**

серверов UNIX. При настройке **Dr.Web MailD** и 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, и в этом случае он указывается отдельно для каждого параметра.

Возможные основные действия:

- `Cure` — попытаться вылечить зараженный объект;
- `Remove` — удалить зараженный объект;
- `Discard` — отклонить письмо, не уведомляя отправителя;
- `Pass` — пропустить письмо;
- `Reject` — отклонить письмо, уведомив отправителя;
- `Tempfail` — уведомить отправителя, что письмо временно не может быть доставлено;

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]. Можно указать несколько адресов, разделяя их символом "|";
- `Notify` — выслать 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. `ЗАГОЛОВОК` задается в виде `[ИМЯ:]ЗНАЧЕНИЕ`, где `ИМЯ` - название заголовка (`X-DrWeb-MailD` по умолчанию), а `ЗНАЧЕНИЕ` - значение заголовка. При использовании в заголовке символа ";", а также символов "(" и ")" их необходимо экранировать. В противном случае конфигурация может быть интерпретирована некорректно.

Экранирование 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 директорию карантина;
- `Delete` — удалить зараженный файл;
- `Rename` — переименовать файл;
- `Ignore` – пропустить файл;
- `Report` — только вывести информацию в отчет.
- `Cure` — попытаться вылечить зараженный объект.

- `адрес (address)` — адреса сокетов компонентов **Dr.Web для почтовых серверов UNIX** и внешних модулей. Такие параметры задаются в формате `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 значении параметра это указывается явно.
- `текст (text value)` — значение параметра задается в виде текстовой строки, текст в строке может быть заключен в кавычки (если в строке есть пробелы,

кавычки обязательны).

- настройки пула (pool options) - настройки пула потоков.

  Первым определяется количество потоков в пуле:

  - auto — количество потоков определяется автоматически в зависимости от загрузки системы;
  - N — целое неотрицательное число. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности;
  - N-M — целые положительное значения, и M ≥ N. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности, пока число потоков не достигнет значения M.

  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} — статистика по потокам в пуле. Статистика сохраняется при получении системного сигнала SIGUSR1 в директории, определенной значением параметра **BaseDir** секции [General]. Значение по умолчанию: no
  - log_level = {Quiet|Error|Alert|Info|Debug} — уровень подробности файла протокола 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, timeout=1m, stat = yes

В данном примере число потоков определяется автоматически, максимальное время ожидания активности потока составляет одну минуту, статистика по потокам в пуле ведется.

- lookups — разделенные запятыми объекты для поиска, заданные в форме [ПРЕФИКС1:]ЗНАЧЕНИЕ1, [ПРЕФИКС2:]ЗНАЧЕНИЕ2, ... Если префикс не задан, значение используется без префикса.

  Возможные значения префикса:

  - value — за ним указывается непосредственно искомое значение. Этот префикс используется, если, к примеру, в значении встречается символ ": ".
  - file — путь к файлу. Каждое значение в файле должно располагаться на новой строке. Это наиболее быстрый метод поиска, так как позволяет использовать сортировку и бинарный поиск в файлах.
  - regex — регулярное выражение (синтаксис Perl).
  - rfile — путь к файлу. Файл содержит набор регулярных выражений (синтаксис Perl), каждое из которых должно располагаться на новой строке.
  - ldap — путь поиска на LDAP-сервере.
  - odbc, oracle — SQL-запрос к хранилищу ODBC или Oracle.
  - postgres — SQL-запрос к хранилищу PostgreSQL.
  - cdb — текстовое имя ключа в базе

данных. Сама база данных CDB представляет собой доступное только для чтения хранилище пар `[текстовый ключ]:[текстовое значение]`.

- `berkeley` — обеспечивает взаимодействие с Berkeley DB.
- `firebird` — SQL-запрос к хранилищу Firebird.
- `sqlite` — SQL-запрос к хранилищу SQLite.
- `mysql` — SQL-запрос к хранилищу MySQL.

- `LookupsLite` — тип значений, аналогичный `lookups`, в котором можно указывать только либо непосредственное значение, либо `lookups` типа `file`.
- `хранилище (storage)` — объекты для хранения данных. Синтаксис аналогичен `lookups`, кроме другого списка префиксов и того, что в `storage` нельзя использовать макрос `$s`.

  Существуют следующие варианты префиксов:

  - `value` — за ним указывается непосредственно искомое значение. Этот префикс используется, если, к примеру, в значении встречается символ ": ".
  - `odbc` — синтаксис аналогичен тому же в `ldap`. 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.
  - `oracle` — синтаксис аналогичен тому же в `odbc`.

- postgres, mysql, sqlite, firebird — синтаксис аналогичен тому же в odbc, за исключением того, что тип char(length) не поддерживается, и для строковых данных следует использовать тип varchar_long.

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.

- настройки TLS/SSL (TLSSettings) — 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. 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 TSL 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).

- `строки и файлы (string)` — набор текстовых значений, разделенных запятыми. Если значение параметра соответствует шаблону `file:/path_to_file`, где `path_to_file` - это путь к файлу, то текстовые значения получаются из файла `path_to_file`. Каждое значение в файле `path_to_file` должно записываться в отдельной строке. Если при получении информации из файла `path_to_file` произошла ошибка, в файл отчета выводится соответствующее диагностическое сообщение и загрузка программы продолжается.
- `прочие значения (value)` — отдельные параметры могут иметь тип, не описанный в предыдущих пунктах данного списка.

Уровень подробности протоколирования работы различных

компонентов программного комплекса **Dr.Web для почтовых серверов UNIX** может быть очень высоким (например, при значении `Debug`, для использования в отладочных целях), а может отсутствовать вовсе (например, при значении `Quiet`, когда файл отчета не ведется). Значения параметров, отвечающих за протоколирование работы компонентов, в общем случае, могут задаваться из следующего набора: `Quiet`, `Error`, `Info`, `Alert`, `Notice`, `Warning`, `Verbose`, `Debug`.

Компоненты **Dr.Web Daemon** и **Dr.Web Scanner** имеют следующие уровни подробности протоколирования работы: `Error`, `Info`, `Notice`, `Warning`, `Alert`. Компонент **Dr.Web Updater** работает с уровнями: `Quiet`, `Error`, `Alert`, `Info`, `Debug`, `Verbose`.

Компонент **Dr.Web Updater** при протоколировании работы использует уровни: `Quiet`, `Error`, `Alert`, `Info`, `Debug`, `Verbose`.

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 администратора (root).

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 **Доктор Веб**, то при каждом использовании графического деинсталлятора вам будет предложено удалить абсолютно все модули **Доктор Веб**, 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 директорий. В процессе установки данного компонента будут созданы пользователь `drweb` и группа `drweb`;
- `drweb-bases`: пакет содержит антивирусное ядро 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 и вирусных баз. 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 центральной защиты. Для установки требует пакеты drweb-agent, drweb-updater и drweb-scanner;
- 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 плагина drweb, его конфигурационный файл, документацию и скрипт конфигурации. Для установки требует пакет drweb-maild;

- drweb-maild-plugin-headersfilter: пакет содержит библиотеку плагина headersfilter, его конфигурационный файл, документацию и скрипт конфигурации. Для установки требует пакет drweb-maild;
- drweb-maild-plugin-modifier: пакет содержит библиотеку плагина modifier, его конфигурационный файл, документацию и скрипт конфигурации. Для установки требует пакет drweb-maild;
- drweb-maild-plugin-vaderetro: пакет содержит конфигурационный файл плагина vaderetro, документацию и скрипт конфигурации. Для установки требует пакеты drweb-maild и drweb-libvaderetro;
- drweb-libvaderetro: пакет содержит библиотеку плагина 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 создана директория 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 директорию `%etc_dir/software/conf/` записываются оригиналы дистрибутивных конфигурационных файлов с названиями в формате `[имя_конфигурационного_файла].N`;
- конфигурационные файлы устанавливаются в соответствующие директории 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 директории 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**.

Пользовательский интерфейс графического инсталлятора

1. При запуске графического инсталлятора командой:

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

открывается окно программы установки.

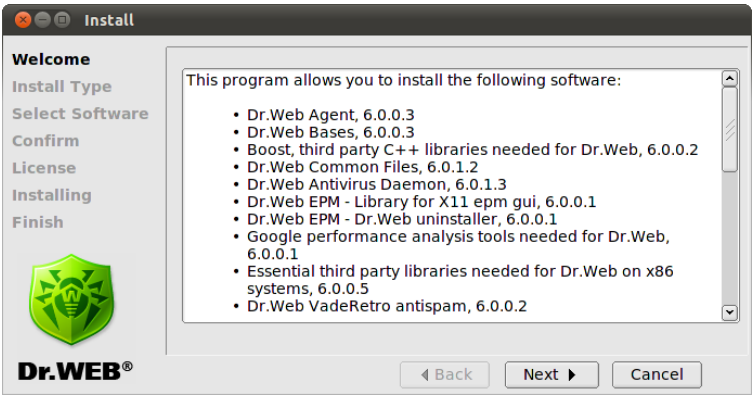


Рис. 1. Окно начала установки программы

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.

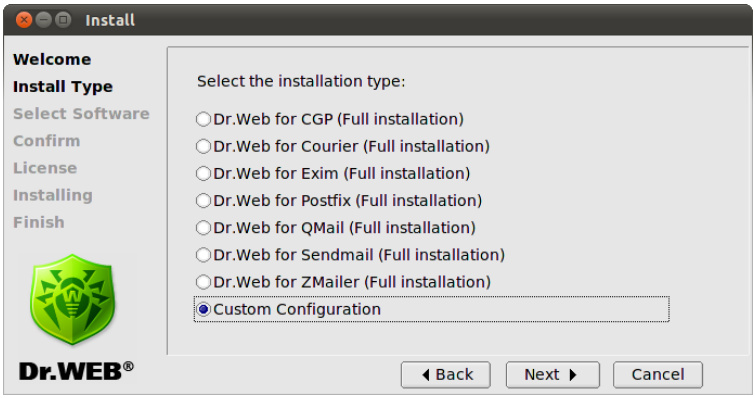


Рис. 2. Окно Install Type для Dr.Web для почтовых серверов UNIX

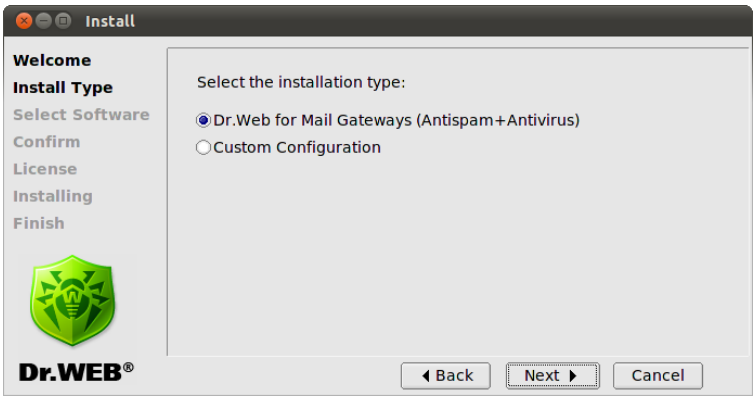


Рис. 3. Окно Install Type для Dr.Web для почтовых шлюзов UNIX

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

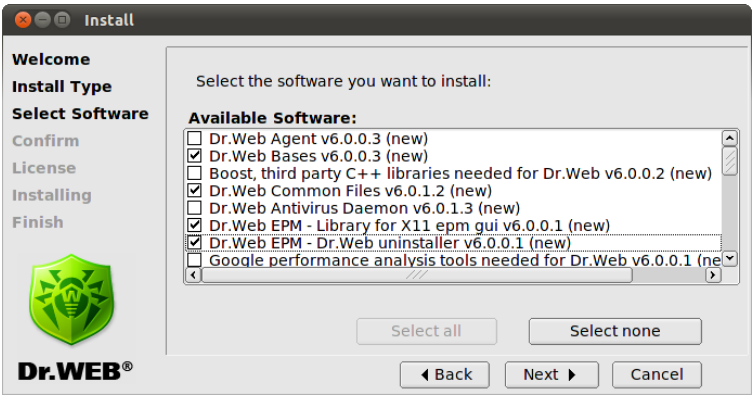


Рис. 4. Окно выбора компонентов для установки

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.

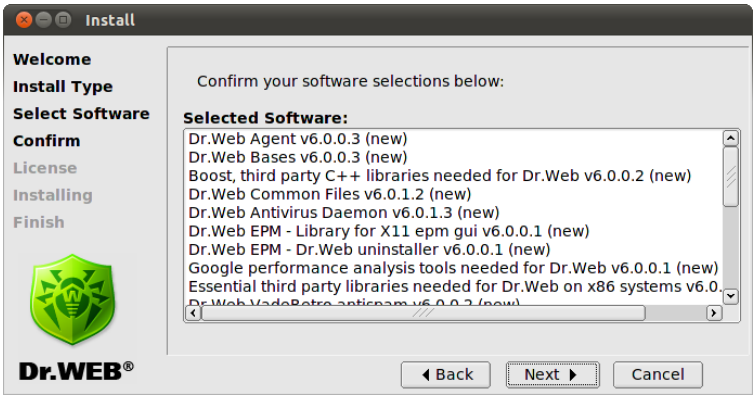


Рис. 5. Окно подтверждения установки компонентов

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**.

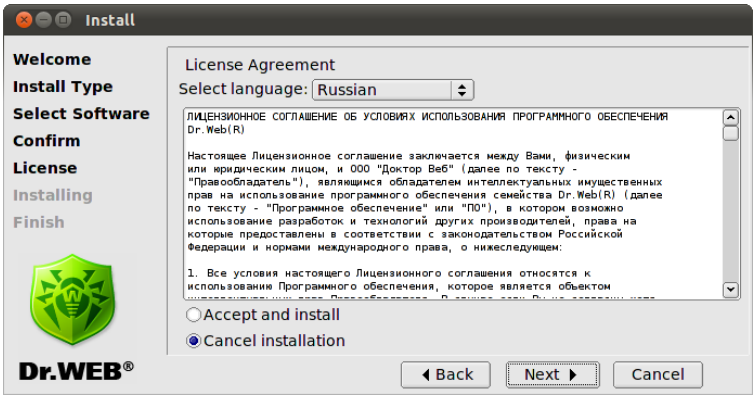


Рис. 6. Окно ознакомления с лицензионным соглашением

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

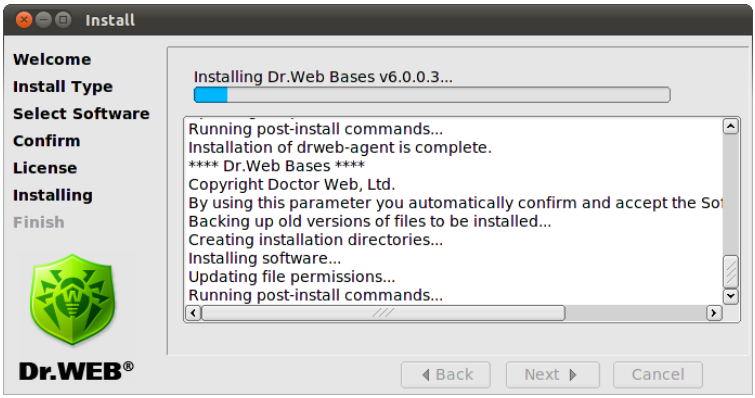


Рис. 7. Окно установки компонентов программы

Одновременно данный отчет копируется в файл `install.log`, расположенный в директории `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**.

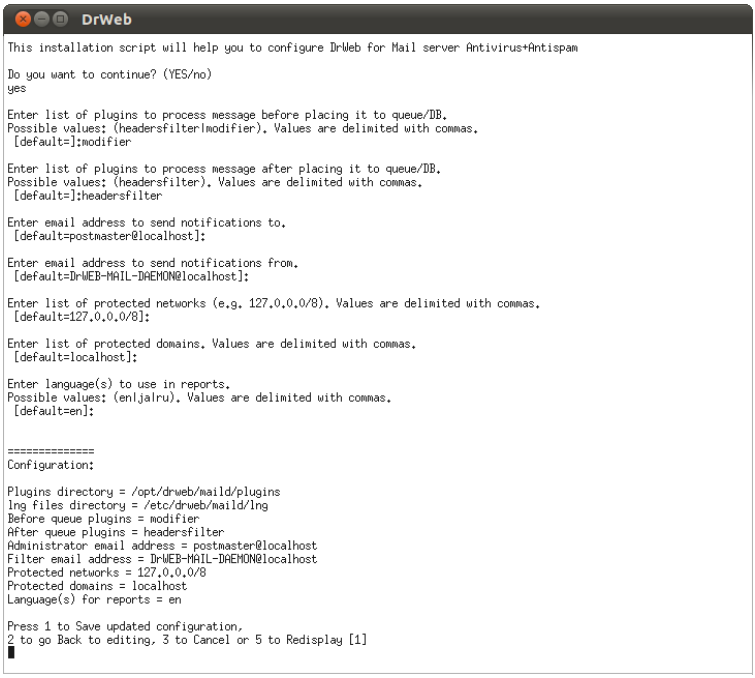


Рис. 8. Интерактивный установочный скрипт

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**).

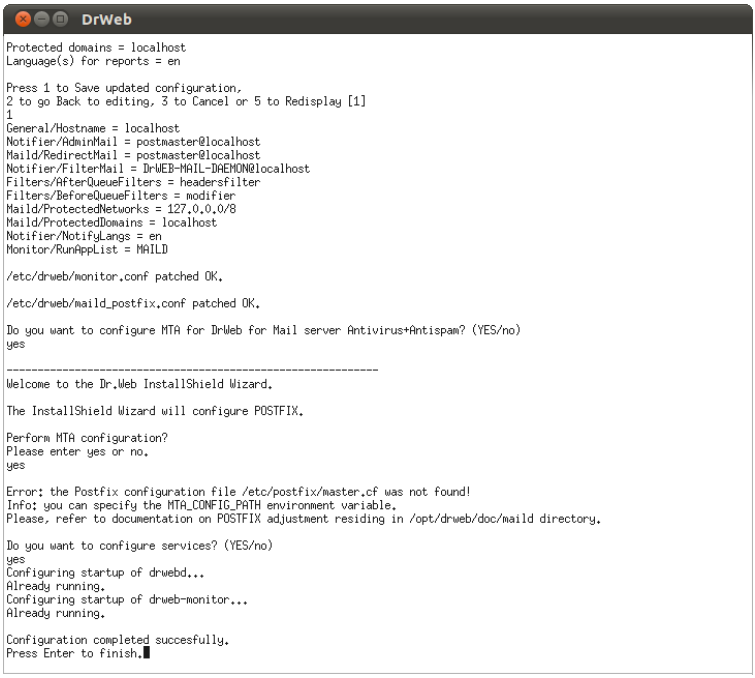


Рис. 9. Настройка MTA и автоматического запуска сервисов

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

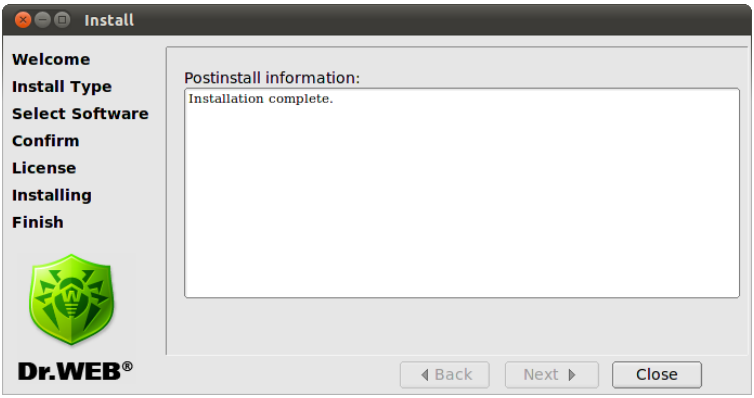


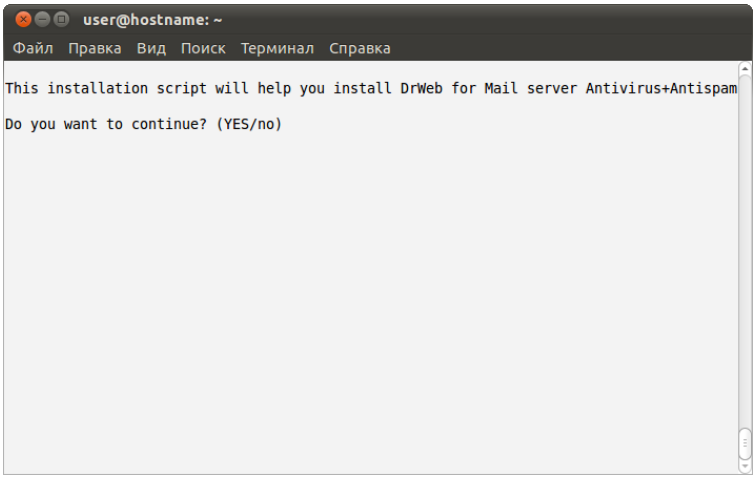
Рис. 10. Окно завершения установки программы

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:

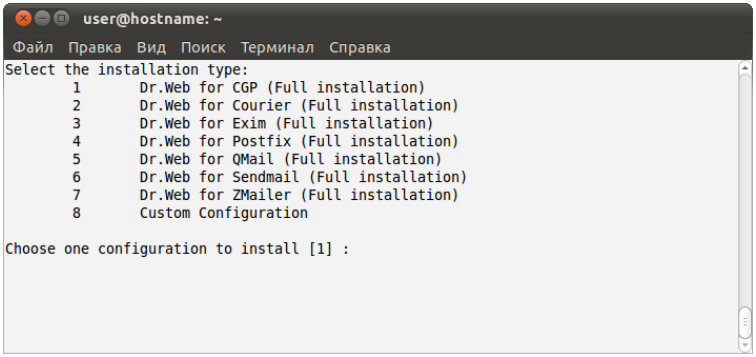
```
# drweb-mail-[название-продукта]_[номер версии]
~[название ОС]/setup.sh
```

Откроется диалоговое окно консольного инсталлятора.



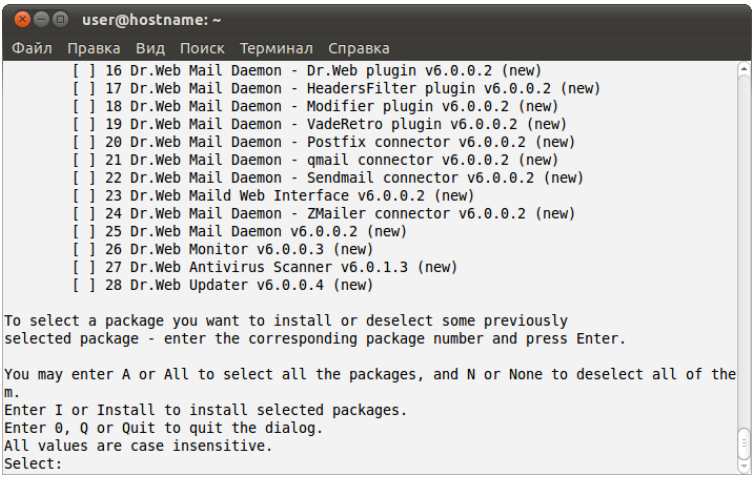
Если вы хотите установить **Dr.Web для почтовых серверов UNIX**, укажите **Y** или **Yes** в строке ввода (значения регистронезависимы) и нажмите клавишу ENTER. В противном случае введите **N** или **No**.

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

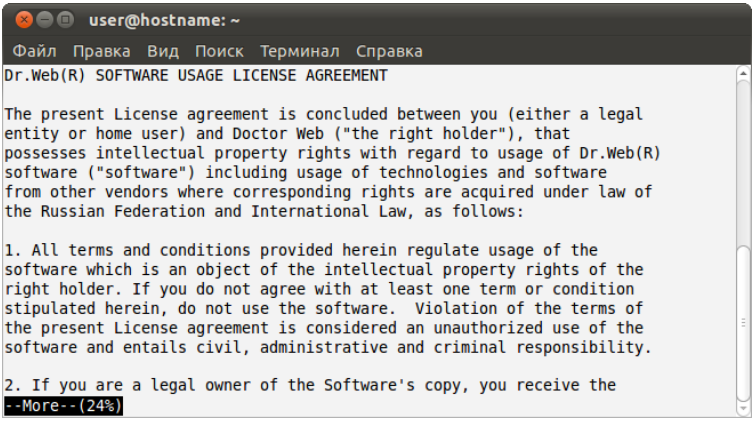


Если вы выбрали пункт **Custom Configuration**, то на следующем этапе вам будет предложено указать необходимые компоненты для установки. Укажите номер соответствующего

компонента в меню и нажмите ENTER.

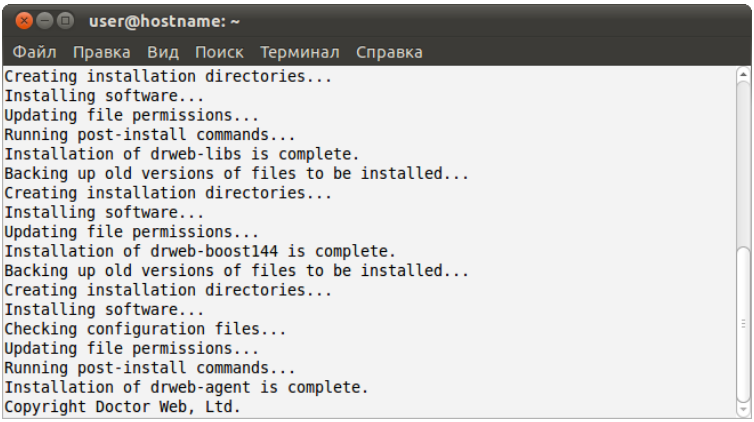


На следующем этапе вам будет предложено ознакомиться с текстом **Лицензионного Договора**. Для пролистывания текста договора нажимайте клавишу ПРОБЕЛ.



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, 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**).

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 директории `%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 директориях сохраняются.

Пользовательский интерфейс графического деинсталлятора

1. При запуске графического деинсталлятора командой:

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

   открывается окно программы удаления компонентов.

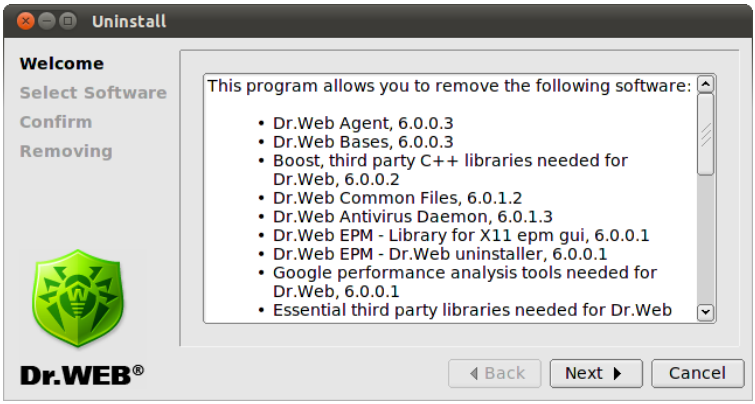


Рис. 11. Окно начала удаления программы

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.

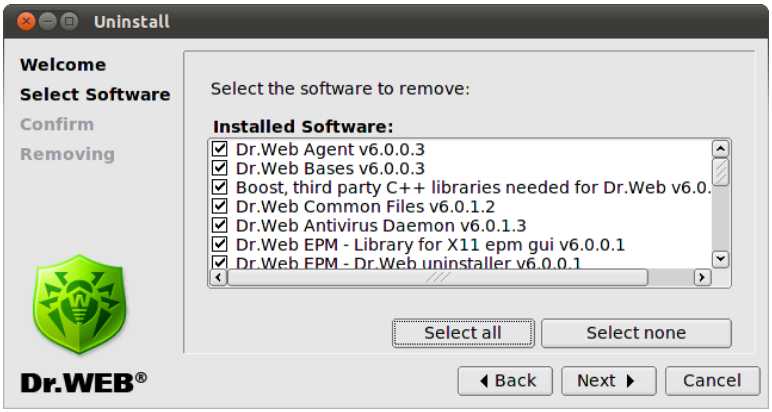


Рис. 12. Окно выбора компонентов для удаления

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.

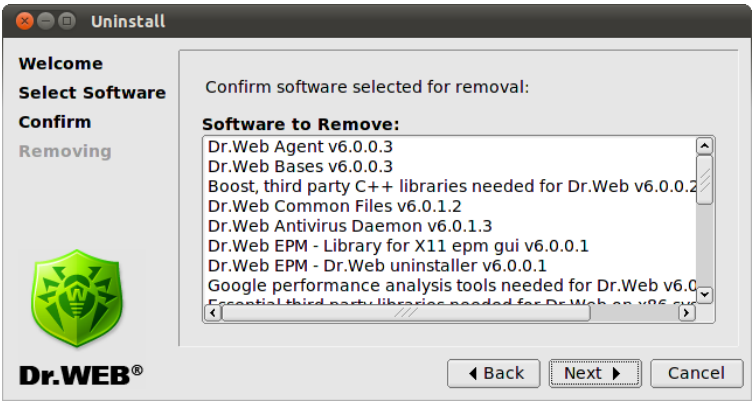


Рис. 13. Окно подтверждения удаления компонентов

4. В последнем окне **Removing** выводится отчет о процессе

удаления компонентов программного комплекса в режиме реального времени.

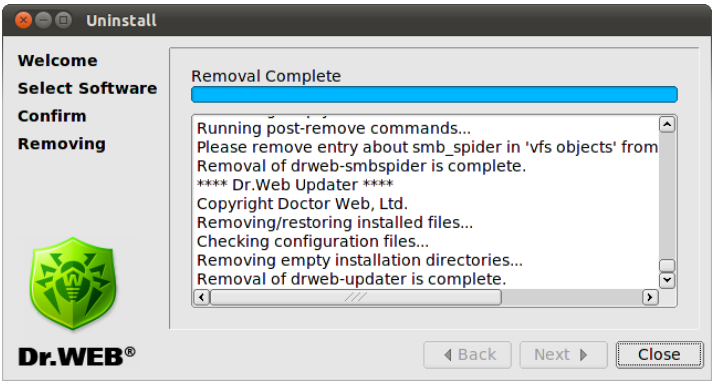


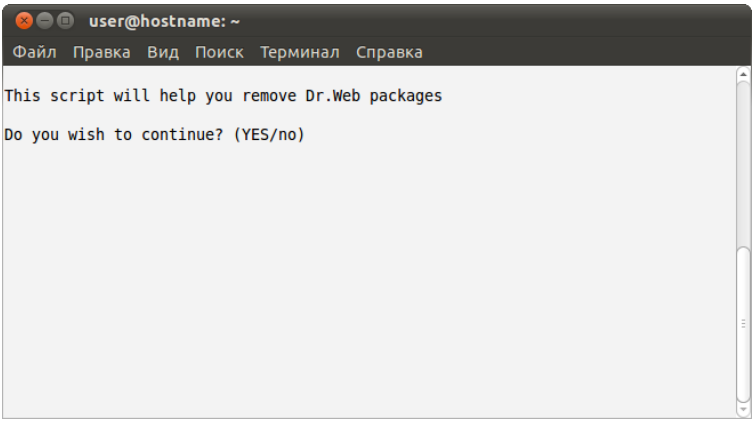
Рис. 14. Окно удаления компонентов программы

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

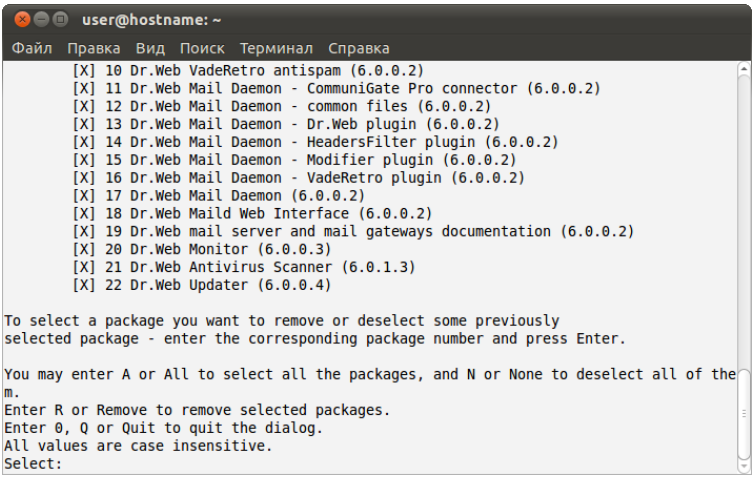
Использование консольного деинсталлятора

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

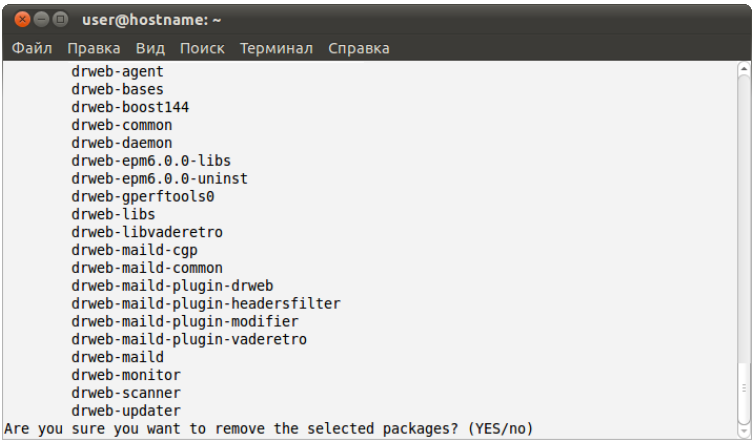
Откроется диалоговое окно консольного деинсталлятора.



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 операционных систем Solaris и 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 не будет запущен пост-инсталляционный скрипт для автоматической установки лицензионного ключевого файла. Ключевой файл необходимо вручную скопировать в директорию `%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 может быть установлен 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).

Debian, Ubuntu (apt)

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>
```

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

```
apt-get remove <имя пакета>
```

Установка и удаление пакетов также может осуществляться с помощью графического менеджера (например, Synaptic).

Обратите внимание, что при установке из нативных пакетов, enable-файл `drwebd` будет располагаться следующим образом:

- `/etc/defaults` — для deb пакетов;
- `/etc/sysconfig` — для rpm пакетов.

ALT Linux, PCLinuxOS (apt-rpm)

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>
```

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

```
apt-get remove <имя пакета>
```

Установка и удаление пакетов также может осуществляться с

помощью графического менеджера (например, Synaptic).

Mandriva (urpmi)

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>
```

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

```
urpme <имя пакета>
```

Установка и удаление пакетов также может осуществляться с

помощью графического менеджера (например, rpmdrake).

Red Hat Enterprise Linux, Fedora, CentOS (yum)

Добавьте файл со следующим содержимым в директорию /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>
```

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

```
yum remove <имя пакета>
```

Установка и удаление пакетов также может осуществляться с помощью графического менеджера (например, PackageKit, Yumex).

SUSE Linux (Zypper)

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>
```

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

```
zypper remove <имя пакета>
```

Установка и удаление пакетов также может осуществляться с помощью графического менеджера (например, YaST).

FreeBSD

Загрузите архив `<имя пакета>_current-current~freebsd_all.tar.gz` с http://officeshield.drweb.com/drweb/freebsd/ports/, распакуйте в отдельную директорию 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 директории `/usr/ports/Mk` с помощью параметра командной строки `-I` . В данной директории располагается дерево портов.

Пример:

```
tar -xzvf <имя пакета>-meta_current-current~freebsd_all.tar.gz
make install -I /usr/ports/Mk/
```

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

Solaris

Нативные пакеты для Solaris могут быть загружены с публичного FTP-сервера:

ftp://ftp.drweb.com/pub/drweb/unix/release/Solaris/packages

и установлены с помощью утилиты `pkgadd`.

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 директории `%bin_dir/maild/scripts/`. При запуске `configure.pl`, в общем случае, предложит указать порядок обработки писем конкретным плагином (до или после помещения письма в базу данных), предпочитаемый язык уведомлений и адрес для их отсылки, а также путь к спискам защищаемых сетей и доменов. 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 плагинов читайте в соответствующих разделах данного руководства (главы Настройка и запуск, Плагины, Интеграция с почтовыми

системами).

Присутствующие в директории `%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 или FreeBSD.

Для Linux и Solaris

Для запуска комплекса необходимо:

1. Зарегистрировать продукт.
2. Скопировать или переместить полученный после регистрации лицензионный ключевой файл с расширением `.key` в директорию 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 директорию `%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 (`enterprise.key`) и ключевой файл продукта (`agent.key`). Переименуйте `agent.key` как `drweb32.key` и скопируйте его в директорию `%bin_dir`.

Если вы хотите использовать ключевой файл, расположенный в какой-либо другой директории, 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 **Агента** `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_dir/drwebd.enable`, присвоив переменной `ENABLE` значение `1`. Это позволит запустить **Dr.Web Daemon**. Если запускать **Dr.Web Daemon** не нужно (используется **Демон**, запущенный на другом компьютере в локальной сети), то для переменной `ENABLE` нужно оставить присвоенное по умолчанию значение `0`.
5. Вручную исправить файл `%etc_dir/drweb-monitor.enable`, присвоив переменной `ENABLE` значение `1`. Это позволит запустить **Dr.Web Monitor**.
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 **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 **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 файле .enable равным нулю.

Для FreeBSD

Для запуска комплекса необходимо:

1. Зарегистрировать продукт.
2. Скопировать или переместить полученный после регистрации лицензионный ключевой файл с расширением `.key` в директорию 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 директорию `%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 (`enterprise.key`) и ключевой файл продукта (`agent.key`). Переименуйте `agent.key` как `drweb32.key` и скопируйте его в директорию `%bin_dir`.

   Если вы хотите использовать ключевой файл, расположенный в какой-либо другой директории, 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 **Агента** `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.

Операционная система с SELinux

Чтобы при работающем SELinux компоненты **Dr.Web Scanner** и **Dr.Web Daemon** могли успешно функционировать, необходимо скомпилировать политики для работы с соответствующими модулями `drweb-scanner` и `drweb-daemon` или установить значение переменной allow_execheap равным `1`.

Пожалуйста, обратите внимание, что во время компиляции модули политик используют шаблоны, большинство из которых разные в зависимости от дистрибутива Linux, его версии, набора политик SELinux и пользовательских настроек. Соответственно, для получения более подробной информации о компиляции модулей политик вы можете обратиться к документации вашего дистрибутива Linux.

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:
   - 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` в Red Hat Enterprise Linux и CentOS Linux, может работать некорректно. В таком случае воспользуйтесь `audit2allow`.

     Пример:

     ```
     # policygentool drweb-scanner /opt/drweb/drweb.real
     ```
     – для **Сканера**.

     ```
     # policygentool drweb-daemon /opt/drweb/drwebd.real
     ```
     – для **Демона**.

     Вам будет предложено указать несколько общих характеристик домена, после чего для каждого модуля будут созданы три файла: `[module_name].te`, `[module_name].fc` и `[module_name].if`

   - С помощью утилиты `audit2allow`. 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.

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

Пример:

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

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

Пример:

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

В данном примере `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`. Если вы хотите внести изменения в разграничения для работы компонентов **Dr.Web для почтовых серверов UNIX**, отредактируйте файл `drweb.te` соответствующим образом и перейдите к пункту 2. Если вносить изменения в файл политики не требуется, перейдите к пункту 4 для установки модуля политики `drweb.pp`.

2. 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:

```
# 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
   ```

Также для разрешения работы компонентов **Dr.Web Scanner** и **Dr.Web Daemon** возможно, но не рекомендуется, установить значение переменной окружения `allow_execheap` равной 1. Переменная окружения `allow_execheap` позволяет или запрещает исполнение данных в *куче (memory heap)* для всех приложений, запущенных в *неограниченном (unconfined)* домене. Чтобы установить значение переменной `allow_execheap`, выполните в командной строке:

```
# setsebool -P allow_execheap = 1
```

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 плагинов программного комплекса **Dr.Web для почтовых серверов UNIX**, которые разрешено использовать данному пользователю (работа некоторых плагинов, например, плагина `headersfilter`, не требует их упоминания в ключевом файле);
- другие ограничения (например, количество писем, которое могут проверять плагины **Dr.Web для почтовых серверов UNIX** за сутки).

Ключевой файл имеет расширение `key` и при работе комплекса по умолчанию должен находиться в одной директории 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 (`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 директорию 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 вышеуказанную директорию.

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, **Сканер** предупредит об этом при запуске.

Демон в такой ситуации может извещать пользователя по электронной почте. Сообщения отправляются для каждого установленного ключевого файла при каждом запуске, перезапуске или перезагрузке **Демона**, 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 директории, отличной от стандартной, 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 плагинов, разрешенных к использованию, составляется из всех плагинов, упомянутых в ключевых файлах (хотя бы в одном из них). Ограничения на работу определенного плагина складываются из ограничений, установленных для этого плагина в разных ключевых файлах.

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

Пример:

Используются три ключевых файла. В одном указан антивирусный плагин drweb, а также ограничение на проверку 10.000 писем ежесуточно. Во втором указан антиспам плагин vaderetro и ограничение на проверку 15.000 писем ежесуточно. В третьем снова указан антивирусный плагин 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 плагинами `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 плагина `vaderetro`, несмотря на то, что плагин `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 **"Доктор Веб"** используется модуль обновления **Dr.Web Updater**. Модуль обновления представляет собой написанный на Perl скрипт `update.pl` и находится в директории, содержащей исполняемые файлы программного комплекса **Dr.Web для почтовых серверов UNIX**.

Настройки модуля обновления хранятся в в секции `[Updater]` главного конфигурационного файла (`drweb32.ini` по умолчанию), который находится в директории `%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]
```

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 обновлений 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 директорию программного комплекса **Dr.Web для почтовых серверов UNIX** (по умолчанию в `%var_dir/bases/`) или иную директорию, определенную 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. 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** (`/opt/drweb/update.pl`). После установки создаётся файл `/etc/cron.d/drweb-update` для запуска **Dr.Web Updater**

каждые 30 минут. Это обеспечивает регулярное обновление и наилучшую защиту. Вы можете исправить файл для изменения периода обновления.

Настройка cron

Для Linux: при установке компонентов программного комплекса в `/etc/cron.d/` будет создан пользовательский файл 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
```

Соответственно, при работе с Solaris можно использовать следующий набор команд:

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

При значениях по умолчанию демон `cron` запускает модуль **Dr.Web Updater** в 0 и 30 минут каждого часа. Это может вызывать повышенную нагрузку на сервера обновления компании **"Доктор Веб"** и приводить к задержке обновления. Чтобы избежать подобной ситуации, рекомендуется изменять значения 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 модуля 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>` - это 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>` - это список имен компонентов, разделенных запятыми.

Пример:

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

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 рабочей директории их резервные копии. Это позволяет вам вернуть

компонент к предыдущему состоянию в случае каких-либо проблем с обновлением.

Чтобы восстановить компонент к предыдущему состоянию, запустите **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 директории `%etc_dir`. Устройство конфигурационного файла и краткое описание его параметров приведены в разделе Конфигурационные файлы.

Секция [`Updater`]

`UpdatePluginsOnly` = {`Yes` | `No`}	Значение Yes предписывает модулю не производить обновление **Демона** и **Сканера**, а ограничиться только обновлением плагинов. 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 файлу}	Путь к файлу программы 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 = {Yes | No}	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. Модуль обновления выбирает сервера обновления из этого списка случайным образом. Данный файл подписан **Доктор Веб**, не подлежит редактированию пользователем и обновляется автоматически. Значение по умолчанию:

	DrlFile = %var_dir/bases/update.drl
CustomDrlFile = {путь к файлу}	Путь к альтернативному файлу, содержащему список серверов обновления. Модуль обновления выбирает сервера обновления из этого списка случайным образом. Данный файл подписан компанией **Доктор Веб**, не подлежит редактированию пользователем и обновляется автоматически. Значение по умолчанию: **CustomDrlFile** = %var_dir/bases/custom.drl
FallbackToDrl = {Yes | No}	Определяет поведение модуля обновления в случае, когда заданы значения обоих параметров **DrlFile** и **CustomDrlFile** одновременно. При указании значения Yes модуль обновления сперва попытается использовать путь из значения параметра **CustomDrlFile**, а в случае неудачи использует путь из значения **DrlFile**. Значение по умолчанию: **FallbackToDrl** = Yes
DrlDir = {путь к директории}	Путь к директории, содержащей подписанные **Доктор Веб** drl-файлы со списками серверов обновления для каждого из плагинов. Значение по умолчанию: **DrlDir** = %var_dir/drl/
Timeout = {время в секундах}	Максимальное время ожидания для загрузки обновлений.

	Значение по умолчанию: **Timeout** = 90
Tries = {числовое значение}	Количество попыток установки соединения модулем обновления. Значение по умолчанию: **Tries** = 3
ProxyServer = {имя или IP-адрес прокси-сервера}	Имя или IP-адрес используемого прокси-сервера. Значение по умолчанию: **ProxyServer** =
ProxyLogin = {имя пользователя прокси-сервера}	Имя пользователя прокси-сервера. Значение по умолчанию: **ProxyLogin** =
ProxyPassword = {пароль пользователя прокси-сервера}	Пароль пользователя прокси-сервера. Значение по умолчанию: **ProxyPassword** =
LogFileName = {имя файла}	Имя файла отчета. В качестве имени можно указать значение syslog, тогда отчет будет вестись средствами системного сервиса syslogd. При использовании syslog нужно обратить внимание на параметры **SyslogFacility** и **SyslogPriority** (см. ниже). Поскольку syslogd имеет несколько файлов для протоколирования разных событий и разных степеней их важности, то, основываясь на этих двух параметрах и содержимом конфигурационного файла syslogd (обычно /etc/syslogd.conf), можно определить, куда будет записываться отчет программы.

	Значение по умолчанию: **LogFileName** = syslog
SyslogFacility = {Daemon | Local0 .. Local7 | Kern | User | Mail}	Тип записи при использовании системного сервиса syslogd. Значение по умолчанию: **SyslogFacility** = Daemon
LogLevel = {Debug | Verbose | Info | Warning | Error | Quiet}	Уровень подробности ведения файла отчета Значение по умолчанию: **LogLevel** = Info
LotusdPidFile = {путь к файлу}	Путь к PID-файлу для демона **Lotus**. Значение по умолчанию: **LotusdPidFile** = %var_dir/run/drweblotusd.pid
MaildPidFile = {путь к файлу}	Путь к PID-файлу для drweb-maild. Значение по умолчанию: **MaildPidFile** = %var_dir/run/drweb-maild.pid
IcapdPidFile = {путь к файлу}	Путь к PID-файлу для drweb-icapd. Значение по умолчанию: **IcapdPidFile** = %var_dir/run/drweb_icapd.pid
BlacklistPath = {путь к директории}	Путь к директории с .dws файлами. Значение по умолчанию: **BlacklistPath** = %var_dir/dws
AgentConfPath = {путь к файлу}	Путь к конфигурационному файлу **Агента**.

	Значение по умолчанию: **AgentConfPath** = %var_dir/agent.conf
PathToVadeRetro = {путь к файлу}	Путь к библиотеке libvaderetro.so. Значение по умолчанию: **PathToVadeRetro** = %var_dir/lib/libvaderetro.so
ExpiredTimeLimit = {number}	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, **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 читает конфигурационный файл.
2. Из конфигурационного файла используются параметры, находящиеся в секции [Updater] (описание параметров см. выше), а также параметры **EnginePath**, **VirusBase**, **UpdatePath** и **PidFile**.
3. Модуль запрашивает с сервера список обновлений, 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 задается значением параметра **`LzmaDecoderPath`** в секции `[Updater]`.

4. Обновления раскладываются по директориям, 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.

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>`, то будет проверен загрузочный сектор соответствующего устройства и при необходимости произведено его лечение. Запущенный без параметров, только с указанием пути в качестве аргумента, консольный сканер **Dr.Web Scanner** (далее **Сканер**) осуществляет проверку указанной директории, используя набор параметров по умолчанию. В следующем примере проверяется домашняя директория пользователя:

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

По окончании проверки, в случае обнаружения зараженных или подозрительных файлов, **Сканер** 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, **Сканер** 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.

Сканер "Доктор Веб" может быть настроен с помощью многочисленных параметров командной строки. Они отделяются от указания пути пробелом и начинаются с символа "-" (дефис). Полный список параметров командной строки можно получить, запустив программу `drweb` с параметрами `-?`, `-h` или `-help`.

Основные параметры программы могут быть сгруппированы следующим образом:

- параметры области проверки;
- параметры диагностики;
- параметры действий;

- параметры интерфейса.

Параметры области проверки указывают, где следует проводить проверку:

- `path` – необязательный параметр для задания пути для сканирования. В одном параметре может быть задано несколько путей;
- `@[+]<файл>` – проверка 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 директориям или просто список подлежащих регулярной проверке файлов;
- `sd` – рекурсивный поиск и проверка файлов в поддиректориях, начиная с текущего;
- `fl` – указание следовать символическим ссылкам, как для файлов, так и для директорий. Ссылки, приводящие к "зацикливанию", игнорируются;
- `mask` – указание игнорировать маски имен файлов.

Параметры диагностики, определяющие, какие типы объектов должны проверяться на вирусы:

- `al` – диагностика всех файлов на заданном устройстве или в указанной в качестве аргумента директории;
- `ar[d|m|r][n]` – проверка файлов в архивах (ARJ, CAB, GZIP, RAR, TAR, ZIP и др.). `d` – удаление, `m` – перемещение, `r` – переименование архивов, содержащих зараженные объекты, `n` – отключение вывода имен архиваторов. Под архивом в данном случае понимаются не только собственно архивы (например, вида `*.tar`), но и их сжатые формы (в частности, сжатые tar-архивы вида `*.tar.bz2` и `*.tbz`);
- `cn[d|m|r][n]` – проверка файлов в контейнерах (HTML, RTF, PowerPoint и др.). `d` – удаление, `m` – перемещение, `r` – переименование контейнеров, содержащих зараженные объекты, `n` – отключение вывода типа контейнера;
- `ml[d|m|r][n]` – проверка файлов почтовых программ.

d – удаление, m – перемещение, r – переименование файлов почтовых программ, содержащих зараженные объекты, n – отключение вывода типа файлов почтовых программ;

- upn – проверка исполняемых файлов, упакованных LZEXE, DIET, PKLITE, EXEPACK, с отключенным выводом имен утилит упаковки;
- ex – диагностика файлов, имена которых соответствуют заданным маскам (см. параметр конфигурационного файла **FilesTypes**);
- ha – эвристический анализ файлов, поиск неизвестных вирусов.

Параметры действия определяют, какие манипуляции должны быть выполнены в отношении зараженных (или подозрительных) файлов:

- cu[d| m| r] – лечение зараженных файлов. Дополнительные параметры: d – удаление, m – перемещение, r – переименование зараженных файлов;
- ic[d| m| r] – действия для неизлечимых файлов: d – удаление, m – перемещение, r – переименование неизлечимых файлов;
- sp[d| m| r] – действия для подозрительных файлов: d – удаление, m – перемещение, r – переименование подозрительных файлов;
- adw[d| m| r| i] – действия для файлов, содержащих рекламные программы: d – удаление, m – перемещение, r – переименование, i – игнорирование;
- dls[d| m| r| i] – действия для файлов, содержащих программы дозвона: d – удаление, m – перемещение, r – переименование, i – игнорирование;
- jok[d| m| r| i] – действия для файлов, содержащих программы-шутки : d – удаление, m – перемещение, r – переименование, i – игнорирование;
- rsk[d| m| r| i] – действия для файлов, содержащих потенциально опасные программы: d – удаление, m –

перемещение, r – переименование, i – игнорирование;

- hck[d| m| r| i] – действия для файлов, содержащих программы, используемые для взлома: d – удаление, m – перемещение, r – переименование, i – игнорирование.

Параметры интерфейса определяют условия вывода результатов работы программы:

- v, version – вывод информации о версии продукта и версии антивирусного ядра;
- ki – вывод информации о ключе и его владельце (только в кодировке UTF8);
- foreground[yes| no] – запуск **Сканера** в приоритетном или в фоновом режиме;
- ot – вывод информации на stdout, т.е стандартный вывод;
- oq – отключение вывода информации;
- ok – вывод сообщения Ok для не зараженных вирусами файлов;
- log=<путь к файлу> – запись отчета о работе в указанный файл;
- ini=<путь к файлу> – использование альтернативного конфигурационного файла;
- lng=<путь к файлу> – использование альтернативного языкового файла. Если во время установки был выбран английский язык интерфейса, то для вывода сообщений на русском языке в качестве такого файла следует указать ru_scanner. dwl;
- -a=<адрес Агента> – запуск **Сканера** в режиме центральной защиты;
- --only-key – при запуске **Сканер** получает от **Агента** только лицензионный ключевой файл.

Некоторые из параметров отменяют соответствующее им действие, если оканчиваются символом "-" (дефис). К ним принадлежат параметры:

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

Например, при запуске **Сканера** командой вида:

```
$ drweb <путь> -ha-
```

проверка будет производиться без эвристического анализа файлов, который обычно по умолчанию включен.

Если не производились действия по перенастройке программы, то по умолчанию (т.е. без отдельного указания параметров), **Сканер** запускается с параметрами:

```
-ar -ha -fl- -ml -sd
```

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 символ "-" (дефис), как это было показано выше на примере параметра `-ha` (эвристический анализ).

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 **Сканера** 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;
- `spm` – перемещение подозрительных файлов;
- `spr` – переименование подозрительных файлов.

Запуск **Сканера** с параметром лечения 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 отдельную директорию, которая и будет указана как аргумент при повторном запуске **Сканера**.

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

Параметр переименования 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, выявленных при эвристическом анализе как подозрительные. Переименование сделает невозможным случайный запуск исполняемых модулей в этих системах или загрузку зараженных документов приложений без дальнейшей проверки и таким образом предотвратит заражение возможным вирусом и дальнейшее его распространение.

Параметр перемещения переместит зараженный (или

подозрительный) файл в предназначенную для этого директорию 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 **Сканера** 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. Однако набор параметров по умолчанию может быть изменен и при настройке **Сканера**, о чем говорится в следующем разделе.

Настройки

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

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

Устройство конфигурационного файла и краткое описание его параметров приведены в разделе Конфигурационные файлы.

Секция [`Scanner`]

`EnginePath` = {путь к файлу, обычное расширение dll}	Расположение модуля `drweb32.dll` (антивирусное ядро). Этот параметр также используется модулем обновления. Значение по умолчанию: **`EnginePath`** = `%bin_dir/lib/drweb32.dll`
`VirusBase` = {список путей (масок) к файлам, обычное расширение vdb}	Маски для подключаемых вирусных баз. Этот параметр также используется модулем обновления. Допустимо перечисление нескольких масок. Значение по умолчанию: **`VirusBase`** = `%var_dir/bases/*.vdb, %var_dir/bases/*.VDB`
`UpdatePath` = {путь к директории}	Этот параметр используется модулем обновления (`update.pl`) и должен быть задан обязательно.

	Значение по умолчанию: **UpdatePath** = %var_dir/updates/
TempPath = {путь к директории}	Эта директория используется антивирусным ядром для создания временных файлов. При нормальной работе директория практически не используется, она нужна для распаковки некоторых видов архивов, или когда в системе не хватает памяти. Значение по умолчанию: **TempPath** = /tmp/
LngFileName = {путь к файлу языковых ресурсов, обычное расширение dwl}	Расположение файла языковых ресурсов. Значение по умолчанию: **LngFileName** = %bin_dir/lib/ru_scanner.dwl
Key = {путь к ключевому файлу, обычное расширение key}	Расположение ключевого файла (лицензионного или демонстрационного). 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 = {Yes | No}	Включение использования эвристического анализатора. 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 анализатора. Кроме того, данный режим может незначительно увеличить время проверки. Данные обстоятельства могут быть доводами в пользу отключения эвристического анализа. Вместе с тем, включение этого типа анализа увеличивает надежность антивирусной защиты. Все файлы, обнаруженные эвристическим анализатором, 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 **Сканера**.
	Значение по умолчанию: **HeuristicAnalysis** = Yes
ScanPriority = {значение}	Приоритет работы **Сканера**. 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}	Типы файлов, которые будут проверяться при сканировании по типу, т.е. когда параметр **ScanFiles** (см. ниже) имеет значение ByType. Допускаются символы "*" и "?". Допускается несколько строк с этим параметром; в этом случае задаваемые списки суммируются. 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 = {Yes | No}	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 режимах локального сканирования. Значение по умолчанию:

	ScanFiles = All
ScanSubDirectories = {Yes | No}	Проверка содержимого поддиректорий. Значение по умолчанию: **ScanSubDirectories** = Yes
CheckArchives = {Yes | No}	Распаковка архивов форматов ZIP (WinZip, InfoZIP и др.), RAR, ARJ, TAR, GZIP, CAB и др. Значение по умолчанию: **CheckArchives** = Yes
CheckEMailFiles = {Yes | No}	Проверка файлов в почтовых (e-mail) форматах. Значение по умолчанию: **CheckEMailFiles** = Yes
ExcludePaths = {список путей (масок) для исключения из проверки}	Маски для тех файлов, которые не должны проверяться. Значение по умолчанию: **ExcludePaths** = /proc,/sys,/dev
FollowLinks = {Yes | No}	Следование символическим ссылкам при сканировании. Значение по умолчанию: **FollowLinks** = No

RenameFilesTo = {маска}	Маска для переименования файлов, если для данной ситуации (зараженный или подозрительный файл) задано действие Rename. К примеру, если задана маска #??, то первая буква расширения файла будет заменена на символ #, а остальные буквы будут сохранены. Если файл не имел расширения, оно будет состоять из одного символа #. Значение по умолчанию: **RenameFilesTo** = #??
MoveFilesTo = {путь к директории}	Путь к директории карантина. Значение по умолчанию: **MoveFilesTo** = %var_dir/infected/
EnableDeleteArchiveAction = {Yes | No}	Применение действия Delete (Удалить) для составных объектов (архивов, почтовых ящиков, html-страниц), если они содержат зараженные объекты. Важно понимать, что будет удален весь составной объект, т.е. весь архив или весь почтовый ящик, а не только зараженное письмо или элемент архива. Значение по умолчанию: **EnableDeleteArchiveAction** = No
InfectedFiles = {Report | Cure | Delete | Move | Rename | Ignore}	Задает реакцию на обнаружение файла, зараженного известным вирусом. Допустимые значения параметра: • Report - только вывести информацию в отчет; • Cure - попытаться вылечить объект (только для параметра **InfectedFiles**); • Delete - удалить зараженный файл;

	• `Move` - переместить файл в директорию, заданную параметром **`MoveFilesTo`**; • `Rename` - переименовать файл, используя маску, заданную параметром **`RenameFilesTo`**; • `Ignore` – пропустить файл. Удаление и перемещение, заданное в связи с обнаружением зараженных объектов в архивах, контейнерах и почтовых ящиках, применяется к соответствующему архиву, контейнеру или почтовому ящику 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` = {`Report` | `Delete` | `Move` | `Rename` | `Ignore`}	Возможно, файл заражен неизвестным вирусом
	Значение по умолчанию: **`SuspiciousFiles`** = `Report`
`IncurableFiles` = {`Report` | `Delete` | `Move` | `Rename` | `Ignore`}	Файл заражен и не может быть вылечен (имеет смысл, только если **`InfectedFiles`** = `Cure`)
	Значение по умолчанию: **`IncurableFiles`** = `Report`
`ActionAdware` = {`Report` | `Delete` | `Move` | `Rename` | `Ignore`}	Файл содержит программу для показа рекламы (adware).
	Значение по умолчанию: **`ActionAdware`** = `Report`

ActionDialers = {Report | Delete | Move | Rename | Ignore}	Файл содержит программу автоматического дозвона. Значение по умолчанию: **ActionDialers** = Report
ActionJokes = {Report | Delete | Move | Rename | Ignore}	Файл содержит программу-шутку, которая может пугать или раздражать пользователя Значение по умолчанию: **ActionJokes** = Report
ActionRiskware = {Report | Delete | Move | Rename | Ignore}	Файл содержит потенциально опасную программу, которая может быть использована не только ее владельцем, но и злоумышленниками. Значение по умолчанию: **ActionRiskware** = Report
ActionHacktools = {Report | Delete | Move | Rename | Ignore}	Файл содержит программу, которая используется для взлома компьютеров. Значение по умолчанию: **ActionHacktools** = Report
ActionInfectedMail = {Report | Delete | Move | Rename | Ignore}	Сообщение или почтовый ящик содержат зараженный объект. Значение по умолчанию: **ActionInfectedMail** = Report
ActionInfectedArchive = {Report | Delete | Move | Rename | Ignore}	Архив (ZIP, TAR, RAR и др.) содержит зараженный файл. Значение по умолчанию: **ActionInfectedArchive** = Report
ActionInfectedConta	Контейнер (OLE, HTML, PowerPoint и др.) содержит зараженный объект.

iner = {Report | Delete | Move | Rename | Ignore}	Значение по умолчанию: **ActionInfectedContainer** = Report

Параметры регистрации событий:

LogFileName = {имя файла}	Имя файла отчета. В качестве имени можно указать syslog, тогда отчет будет вестись средствами системного сервиса syslogd. При использовании syslogd нужно обратить внимание на параметры **SyslogFacility** и **SyslogPriority** (см. ниже). Поскольку syslogd имеет несколько файлов для протоколирования разных событий и разных степеней их важности, то, основываясь на этих двух параметрах и содержимом конфигурационного файла syslogd (обычно /etc/syslogd.conf), можно определить, куда будет писаться отчет программы.
	Значение по умолчанию: **LogFileName** = syslog
SyslogFacility = {Daemon | Local0 .. Local7 | Kern | User | Mail}	Тип записи при использовании системного сервиса syslogd.
	Значение по умолчанию: **SyslogFacility** = Daemon
SyslogPriority = {Alert | Warning | Notice | Info | Error}	Приоритет записи при использовании системного сервиса syslogd.
	Значение по умолчанию: **SyslogPriority** = Info

LimitLog = { Yes | No}	Ограничение размера файла отчета. Параметр не влияет на работу программы при значении **LogFileName** = syslog. Ограничение размера файла отчета реализуется следующим образом: при запуске **Сканер** проверяет размер файла отчета, и если он превышает значение, заданное в параметре **MaxLogSize**, файл отчета стирается и ведение отчета начинается с нуля. Значение по умолчанию: **LimitLog** = No
MaxLogSize = {значение в КБайтах}	Максимальный размер файла отчета. Имеет смысл только если **LimitLog** = Yes. Если указано значение 0, размер файла отчета проверяться не будет. Значение по умолчанию: **MaxLogSize** = 512
LogScanned = { Yes | No}	Вывод в файл отчета информации обо всех проверяемых объектах, независимо от того, обнаружены вирусы или нет. Значение по умолчанию: **LogScanned** = Yes
LogPacked = { Yes | No}	Вывод в файл отчета дополнительной информации о файлах, упакованных утилитами DIET, PKLITE и др. Значение по умолчанию: **LogPacked** = Yes
LogArchived = { Yes | No}	Вывод в файл отчета дополнительной информации об архиваторах. Значение по умолчанию:

	LogArchived = Yes
LogTime = {Yes | No}	Вывод в файл отчета времени каждой записи. Параметр не имеет смысла, если **LogFileName** = syslog. Значение по умолчанию: **LogTime** = Yes
LogStatistics = {Yes | No}	Запись в отчет суммарной статистики задания для сканирования. Значение по умолчанию: **LogStatistics** = Yes
RecodeNonprintable = {Yes | No}	Перекодировка при выводе в файл отчета 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 = {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 = {значение в КБайтах}	Минимальный размер файла внутри архива, начиная с которого будет производиться проверка коэффициента сжатия (если это предписано параметром **MaxCompressionRatio**).
	Значение по умолчанию: **CompressionCheckThreshold** = 1024
MaxFileSizeToExtract = {значение в КБайтах}	Максимальный размер файла, извлекаемого из архива. 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 = {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 = {значение в МБайтах}	Максимальный размер памяти, выделяемой **Сканером** при сканировании одного файла (в мегабайтах). Если установлено значение 0, размер выделяемой памяти не ограничен.
	Значение по умолчанию: **MaximumMemoryAllocationSize** = 0
ScannerScanTimeout = {время в секундах}	Максимальное время сканирования одного файла (в секундах). Если установлено значение 0, время сканирования одного файла не ограничено.
	Значение по умолчанию: **ScannerScanTimeout** = 0

MaxBasesObsolescencePeriod = {время в часах}	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 сокета Агента}	Адрес сокета **Агента** в формате ТИП:АДРЕС, где ТИП может принимать значения inet (для TCP-сокетов), local и unix (для UNIX сокетов). **Пример**: **ControlAgent** = inet:4040@127.0.0.1,local:/var/drweb/ipc/.agent **Сканер** получает от **Агента** ключ и конфигурационный файл (если в качестве значения параметра **OnlyKey** задано No). Значение по умолчанию: **ControlAgent** = local:%var_dir/ipc/.agent
OnlyKey = {Yes | No}	Подключение возможности запросить только ключевой файл от **Агента**, не запрашивая конфигурацию. При этом будет использоваться локальный конфигурационный файл.

	Если указан адрес сокета **Агента** и значение параметра **OnlyKey** установлено в `No`, **Агенту** будет отправляться статистика работы **Сканера** (после сканирования каждого файла **Сканер** будет отправлять информацию **Агенту**). Значение по умолчанию: **OnlyKey** = No

Запуск

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

```
$ %bin_dir/drweb
```

В том случае, если директория `%bin_dir` внесена в переменную окружения командной оболочки `PATH`, запуск осуществляется из произвольной директории. Следует учесть, что последний вариант не рекомендуется из соображений безопасности, равно как и создание символической ссылки на исполняемый файл drweb в какой-либо из директорий типа `/bin/`, `/usr/bin/` и т.д.

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

После запуска **Сканера** 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.

Все иные действия по обнаружению и обезвреживанию вирусов требуют применения параметров командной строки.

Антивирусный модуль Dr.Web Daemon

Dr.Web Daemon - постоянно загруженный антивирусный модуль, который позволяет по запросу от других компонентов комплекса проверять файлы на диске или данные, переданные по сокету. Запросы осуществляются по специальному протоколу через UNIX сокеты или TCP-сокеты. **Dr.Web Daemon** использует то же ядро и вирусные базы, что и **Dr.Web Scanner**, и способен обнаруживать и лечить все известные вирусы.

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.

Параметры командной строки

Как и для любой UNIX программы, для **Демона Dr.Web** предусматриваются параметры командной строки. Они отделяются от указания пути пробелом и предваряются символом "-" (дефис). Полный список можно получить, запустив программу `drwebd` с параметрами `-?`, `-h` или `-help`.

Параметры командной строки **Демона Dr.Web**:

- `-ini=<путь к файлу>` — использование альтернативного конфигурационного файла;
- `--foreground=<yes|no>` — задание режима работы **Демона** при запуске. Если выбрано значение `Yes`, то

Демон будет работать как приоритетная задача; при значении `No` **Демон** будет работать в фоновом режиме;

- `--check-only <параметры командной строки для проверки>` — проверка правильности конфигурации **Демона** при запуске. Если указаны какие-либо параметры командной строки, то правильность задаваемых с их помощью значений также будет проверена;
- `-a=<адрес Агента>` — запуск **Демона** в режиме центральной защиты;
- `--only-key` — при запуске **Демон** получает от **Агента** только лицензионный ключевой файл.

Запуск

В процессе загрузки **Демона** осуществляются следующие действия:

- поиск и загрузка конфигурационного файла. Если конфигурационный файл не найден, загрузка **Демона** прекращается. Путь к конфигурационному файлу может быть задан при запуске параметром командной строки `-ini: {путь/к/drweb32.ini}`, иначе будет использовано значение по умолчанию (`%etc_dir/drweb32.ini`). При загрузке проверяется допустимость некоторых параметров и, если значение параметра недопустимо, берется значение по умолчанию;
- создается файл отчета. Директория с файлом отчета должна быть доступна на запись пользователю, с чьими правами работает **Демон**. Директория по умолчанию `/var/log/` недоступна пользователям на запись. Поэтому, если задано значение параметра **User**, необходимо также указать путь к альтернативной директории для хранения отчётов в значении параметра **LogFileName**;
- производится загрузка ключевого файла по пути, указанному в конфигурационном файле. Если ключевой файл не найден, загрузка **Демона** прекращается;

- если задан параметр **User**, **Демон** пытается изменить свои права;
- производится загрузка антивирусного ядра (drweb32.dll). Если антивирусное ядро не найдено (ошибки в конфигурационном файле) или повреждено, загрузка **Демона** прекращается;
- 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 **Демона** продолжается;
- **Демон** отключается от терминала, поэтому сообщения о дальнейших проблемах не могут быть выведены на терминал и выводятся только в файл отчета;
- создается сокет, в случае использования TCP-сокетов, возможно, не один. Если какой-либо TCP-сокет создать не удалось, загрузка **Демона** продолжается. В случае использования UNIX сокета следует убедиться, что директория, его содержащая, доступна на запись и чтение пользователю, с чьими правами работает **Демон**. Для пользователей, с правами которых будут работать интеграционные модули, директория должна быть доступна на выполнение, а сам файл сокета — на запись и чтение. Директория по умолчанию /var/run/ недоступна пользователям на запись и выполнение. Поэтому, если задано значение параметра **User**, необходимо также указать путь к альтернативной директории для сокетов в значении параметра **Socket**. Если UNIX сокет создать не удалось, загрузка **Демона** прекращается;

- после этого создается PID-файл, в котором хранится информация об идентификаторе процесса **Демона** и о транспортных адресах, по которым доступен **Демон**. Директория с PID-файлом должна быть доступна на запись пользователю, с чьими правами работает **Демон**. Директория по умолчанию `/var/run/` недоступна пользователям на запись и выполнение. Поэтому, если задано значение параметра **`User`**, необходимо также указать путь к альтернативной директории для PID-файла в значении параметра **`PidFile`**. Если создать PID-файл не удалось, загрузка **Демона** прекращается.

Проверка работоспособности Dr.Web Daemon

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

```
$ netstat -a
```

В случае TCP-сокетов:

```
--- cut ---
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address Foreign Address State
tcp 0 0 localhost:3000 *:* LISTEN
raw 0 0 *:icmp *:* 7
raw 0 0 *:tcp *:* 7
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags Type State I-Node Path
unix 0 [ ACC ] STREAM LISTENING 384 /dev/gpmctl
```

```
unix  0  [ ]  STREAM  CONNECTED  190  @0000001b
unix  1  [ ]  STREAM  CONNECTED  1091  @00000031
unix  0  [ ACC ]  STREAM  LISTENING  403 /tmp/.font-unix/fs7100
unix  4  [ ]  DGRAM  293  /dev/log
unix  1  [ ]  STREAM  CONNECTED  1092  /dev/gpmctl
unix  0  [ ]  DGRAM  450
unix  0  [ ]  DGRAM  433
unix  0  [ ]  DGRAM  416
unix  0  [ ]  DGRAM  308
--- cut ---
```

В случае UNIX сокетов:

```
--- cut ---
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address Foreign Address State
raw  0  0 *:icmp  *:*  7
raw  0  0 *:tcp  *:*  7
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags Type State I-Node Path
unix  0  [ ACC ]  STREAM  LISTENING  384 /dev/gpmctl
unix  0  [ ]  STREAM  CONNECTED  190 @0000001b
unix  1  [ ]  STREAM  CONNECTED  1091 @00000031
unix  0  [ ACC ]  STREAM  LISTENING  1127 %var_dir/.daemon
```

```
unix  0  [ ACC ]  STREAM  LISTENING  403 /tmp/.
font-unix/fs7100
unix  4  [ ]  DGRAM  293  /dev/log
unix  1  [ ]  STREAM  CONNECTED  1092 /dev/
gpmctl
unix  0  [ ]  DGRAM  450
unix  0  [ ]  DGRAM  433
unix  0  [ ]  DGRAM  416
unix  0  [ ]  DGRAM  308
--- cut ---
```

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

Для проверки работоспособности **Демона** можно использовать консольный клиент **Демона** (`drwebdc`), запустив его для получения служебной информации о **Демоне**. 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:

```
--- cut ---
- 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.
--- cut ---
```

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 **Демона** 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).

В случае лицензии для файловых серверов:

Для TCP-сокета:

```
$ drwebdc -nИМЯ_УЗЛА -pНОМЕР_ПОРТА eicar.com
```

Для UNIX сокета:

```
$ drwebdc -uФАЙЛ_СОКЕТА eicar.com
```

Результатом команды должно быть сообщение:

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

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

Если проверка файла прошла успешно, **Демон** находится в рабочем состоянии.

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

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

Демон Dr.Web имеет два основных режима проверки:

- проверка фрагмента памяти, полученного из сокета;

- проверка файла на диске (локальное сканирование).

При использовании первого режима **Демон** 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 файле отчета **Демона**. Пример работы **Демона** в этом режиме приведен в предыдущем пункте: клиент читает файл и отсылает его **Демону** для проверки. **Демон** может проверять любой фрагмент данных, не обязательно файл.

Более эффективен режим, в котором **Демон** проверяет указанный файл на диске — локальное сканирование. Клиент (консольный клиент или фильтр для почты) сообщает **Демону** лишь путь к файлу, а не передает весь файл. Путь к проверяемому файлу задается относительно **Демона** (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. **Демону** 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 **Демона** 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 **Демону** о его местоположении. На этом этапе нужно правильно распределить права на директорию, в которой фильтрами будут создаваться файлы. Можно порекомендовать либо включить в группу почтовой подсистемы пользователя, чьи права используются **Демоном**, либо сразу запускать **Демона** с правами пользователя, с которыми запускается почтовая система.

В корректно настроенной системе **Демону** в большинстве

случаев не требуется прав администратора.

Обрабатываемые сигналы

Демон Dr.Web может принимать и обрабатывать следующие сигналы:

- `SIGHUP` — перезагрузка конфигурационного файла;
- `SIGTERM` — корректное завершение работы **Демона**;
- `SIGKILL` — принудительное завершение работы **Демона** (в случае проблем).

Файл отчета

Поскольку **Демон Dr.Web** является резидентной программой, информация о его работе может быть получена только из файла отчета. Файл отчета содержит подробности обработки каждого запроса на сканирование, полученного **Демоном**. Имя файла отчета указывается в значении параметра конфигурационного файла `LogFileName`.

Демон 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 отчета (или назначить службу протоколирования 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 **Демоном**, результаты сканирования будут отмечены специальным префиксом при выводе в файл отчета. Возможны следующие префиксы:

- `<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**.

В операционной системе FreeBSD вывод на консоль **Демона** может быть перехвачен системной службой syslog и выведен в файл отчета посимвольно. Эта проблема проявляется если в конфигурационном файле `syslog.conf` установлен уровень протоколирования `*.info`.

Статистика пула процессов

Статистика пула процессов, который используется для обработки запросов на сканирование может быть выведена в файл отчета по сигналу SIGUSR1 (сигнал должен посылаться только родительскому процессу, для дочерних процессов SIGUSR1 приведет к завершению процесса) и при завершении работы **Демона**.

Пример вывода статистики 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` - минимальное 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 **Демон** с настройками по умолчанию, но предпочтительнее настроить его в соответствии с требованиями и условиям эксплуатации. Конфигурационный файл drweb32.ini читается **Демоном** из директории %etc_dir. Для использования другого конфигурационного файла полный путь к нему необходимо указать параметром командной строки при запуске **Демона**.

Устройство конфигурационного файла и краткое описание его параметров приведены в разделе Конфигурационные файлы.

Секция [Daemon]

EnginePath = {путь к файлу, обычное расширение dll}	Расположение модуля drweb32.dll (антивирусное ядро). Этот параметр также используется модулем обновления. Значение по умолчанию: **EnginePath** = %bin_dir/lib/drweb32.dll
VirusBase = {список путей (масок) к файлам, обычное расширение vdb}	Маски для подключаемых вирусных баз. Этот параметр также используется модулем обновления. Допустимо перечисление нескольких масок. Значение по умолчанию: **VirusBase** = %var_dir/bases/*.vdb, %var_dir/bases/*.VDB
UpdatePath = {путь к директории}	Этот параметр используется модулем обновления (update.pl) и должен быть задан обязательно.

	Значение по умолчанию: **UpdatePath** = %var_dir/updates/
TempPath = {путь к директории}	Эта директория используется антивирусным ядром для создания временных файлов. При нормальной работе директория практически не используется, она нужна для распаковки некоторых видов архивов или когда в системе не хватает памяти. Значение по умолчанию: **TempPath** = /tmp/
Key = {путь к ключевому файлу, обычное расширение key}	Расположение ключевого файла (лицензионного или демонстрационного). Ключевой файл может быть различным для **Демона** и для **Сканера**. Соответственно, при необходимости нужно изменить настройки данного параметра. Параметр может задаваться несколько раз, указывая несколько лицензионных ключевых файлов. В таком случае **Демон** 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 15 или 30 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 = {Yes | No}	Значение Yes запрещает **Демону** переходить в режим демона, т.е. становиться фоновым процессом без управляющего терминала. Эта возможность может быть использована некоторыми средствами мониторинга (например, **Монитором**). Значение по умолчанию: **RunForeground** = No
User = {имя пользователя}	Пользователь, с правами которого работает **Демон**. Рекомендуется завести в системе специального пользователя drweb, который будет использоваться **Демоном** и некоторыми фильтрами. Использовать **Демон** 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 **Демона** 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 **Демона**: он создается сканирующей "копией" **Демона** при получении команды и уничтожается после передачи результата ее выполнения. Имя файла, создаваемого каждой "копией" **Демона**, 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} — статистика по процессам в пуле. Статистика сохраняется при получении системного сигнала SIGUSR1 в директории, определенной значением параметра BaseDir секции General. • **stop_timeout** = {время в секундах} — время ожидания остановки работающего процесса.
	Значение по умолчанию: **ProcessesPool** = auto, **timeout** = 120, **stat** = no, **stop_timeout** = 1
OnlyKey = {Yes | No}	Подключение возможности запросить только ключевой файл от **Агента**, не запрашивая конфигурацию. При этом будет использоваться локальный конфигурационный файл.

	Если указан адрес сокета **Агента** и значение параметра **OnlyKey** установлено в `No`, то **Агенту** будет отправляться статистика работы **Демона** (после сканирования каждого файла **Демон** будет отправлять информацию **Агенту**). Значение по умолчанию: **OnlyKey** = `No`
ControlAgent = {адрес сокета Агента}	Адрес сокета **Агента** в формате `ТИП:АДРЕС`, где `ТИП` может принимать значения `inet` (для TCP-сокетов), `local` и `unix` (для UNIX сокетов). **Пример**: **ControlAgent** = `inet:4040@127.0.0.1,local:/var/drweb/ipc/.agent` **Демон** получает от **Агента** лицензионный ключ и конфигурационный файл (если в качестве значения параметра **OnlyKey** задано `No`). Значение по умолчанию: **ControlAgent** = `local:%var_dir/ipc/.agent`

MailCommand = {команда}	Команда, используемая **Демоном** и модулем обновления для отсылки уведомлений пользователю (администратору) по электронной почте. **Демон** использует этот механизм при каждом запуске (перезапуске, перезагрузке), если до истечения срока действия ключевого файла (одного из ключевых файлов) осталось менее 14 дней. Модуль обновления использует этот механизм для рассылки пользователям информационных материалов, подготовленных компанией **Доктор Веб**, в том числе по вопросам, связанным с обновлениями файлов программы. Значение по умолчанию: **MailCommand** = "/usr/sbin/sendmail -i -bm -f drweb -- root"
NotifyPeriod = {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. Этот файл отсылается администратору по истечении срока действия лицензионного ключа. 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 **Демона** или обновлении баз Значение по умолчанию: **NotifyType** = Ever
FileTimeout = {значение в секундах}	Максимальное время проверки одного файла. Если указано значение 0, время проверки файла не ограничивается. Значение по умолчанию: **FileTimeout** = 30
StopOnFirstInfected = {Yes | No}	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 = {значение}	Приоритет сканирующих процессов **Демона**. 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}	Типы файлов, которые будут проверяться при сканировании по типу, т.е. когда параметр **ScanFiles** (см. ниже) имеет значение ByType. Допускаются символы "*" и "?". Допускается несколько строк с этим параметром, в этом случае задаваемые списки суммируются. 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 = {Yes | No}	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 режимах локального сканирования. Значение по умолчанию: **ScanFiles** = All

CheckArchives = {Yes | No}	Распаковка архивов форматов ZIP (WinZip, InfoZIP и др.), RAR, ARJ, TAR, GZIP, CAB и др. Значение по умолчанию: **CheckArchives** = Yes
CheckEMailFiles = {Yes | No}	Проверка файлов в почтовых (e-mail) форматах. Значение по умолчанию: **CheckEMailFiles** = Yes
ExcludePaths = {список путей (масок) для исключения из проверки}	Маски для тех файлов, которые не должны проверяться. Значение по умолчанию: **ExcludePaths** = /proc,/sys,/dev
FollowLinks = {Yes | No}	Следование символическим ссылкам при сканировании. Значение по умолчанию: **FollowLinks** = No
RenameFilesTo = {маска}	Маска для переименования файлов, если для данной ситуации (зараженный или подозрительный файл) задано действие Rename. К примеру, если задана маска #??, то первая буква расширения файла будет заменена на символ #, а остальные буквы будут сохранены. Если файл не имел расширения, оно будет состоять из одного символа #. Значение по умолчанию: **RenameFilesTo** = #??
MoveFilesTo = {путь к директории}	Путь к директории карантина. Значение по умолчанию:

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

Параметры регистрации событий:

LogFileName = {имя файла}	Имя файла отчета. В качестве имени можно указать syslog, тогда отчет будет вестись средствами системного сервиса syslogd. При использовании syslogd нужно обратить внимание на параметры **SyslogFacility** и **SyslogPriority** (см. ниже). Поскольку syslogd имеет несколько файлов для протоколирования разных событий и разных степеней их важности, то, основываясь на этих двух параметрах и содержимом конфигурационного файла syslogd (обычно /etc/syslogd.conf), можно определить, куда будет писаться отчет программы. Значение по умолчанию: **LogFileName** = syslog
SyslogFacility = {Daemon | Local0 .. Local7 | Kern | User | Mail}	Тип записи при использовании системного сервиса syslogd. Значение по умолчанию: **SyslogFacility** = Daemon
SyslogPriority = {Alert | Warning | Notice | Info |	Приоритет записи при использовании системного сервиса syslogd. Значение по умолчанию:

Error}	**SyslogPriority** = Info
LimitLog = {Yes | No}	Ограничение размера файла отчета. Параметр не влияет на работу программы при значении **LogFileName** = syslog. Ограничение размера файла отчета реализуется следующим образом: при запуске или получении сигнала HUP **Демон** проверяет размер файла отчета, и если он превышает значение, заданное в параметре **MaxLogSize**, файл отчета стирается и ведение отчета начинается с нуля. Значение по умолчанию: **LimitLog** = No
MaxLogSize = {значение в КБайтах}	Максимальный размер файла отчета. Имеет смысл только если **LimitLog** = Yes. Если указано значение 0, размер файла отчета проверяться не будет. Значение по умолчанию: **MaxLogSize** = 512
LogScanned = {Yes | No}	Вывод в файл отчета информации обо всех проверяемых объектах, независимо от того, обнаружены вирусы или нет. Значение по умолчанию: **LogScanned** = Yes
LogPacked = {Yes | No}	Вывод в файл отчета дополнительной информации о файлах, упакованных утилитами DIET, PKLITE и др. Значение по умолчанию: **LogPacked** = Yes
LogArchived = {Yes | No}	Вывод в файл отчета дополнительной информации об архиваторах.

	Значение по умолчанию: **LogArchived** = Yes
LogTime = { Yes | No}	Вывод в файл отчета времени каждой записи. Параметр не имеет смысла, если **LogFileName** = syslog. Значение по умолчанию: **LogTime** = Yes
LogProcessInfo = { Yes | No}	Вывод в файл отчета 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 = { Yes | No}	Перекодировка при выводе в файл отчета 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 = {адрес сокета}	Описание сокета, который будет использован для связи с **Демоном**. Существует несколько вариантов задания сокетов для связи с **Демоном**. Если необходимо указать несколько сокетов в одной строке, то можно использовать формат записи ТИП: АДРЕС, где ТИП может принимать значения inet (для TCP-сокетов), local и unix (для UNIX сокетов). **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 **Демон** 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 параметров **Socket** не ограничено, **Демон** будет работать со всеми из описанных сокетов. Чтобы **Демон** 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 = {значение в секундах}	Время, 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 = {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. Если указано значение 0, проверка коэффициента сжатия проводиться не будет. Значение по умолчанию: **MaxCompressionRatio** = 5000

CompressionCheckThreshold = {значение в КБайтах}	Минимальный размер файла внутри архива, начиная с которого будет производиться проверка коэффициента сжатия (если это предписано параметром **MaxCompressionRatio**). Значение по умолчанию: **CompressionCheckThreshold** = 1024
MaxFileSizeToExtract = {значение в КБайтах}	Максимальный размер файла, извлекаемого из архива. Если размер файла внутри архива превышает это значение, он будет пропущен. Значение по умолчанию: **MaxFileSizeToExtract** = 40960
MaxArchiveLevel = {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). Если указано значение 0, уровень вложенности архивов не будет ограничиваться. Значение по умолчанию: **MaxArchiveLevel** = 8

ClientsLogs = {список}	Параметр разделения файлов отчета. Если при обращении к **Демону** клиент передает в расширенных опциях свой идентификатор, файл отчета клиента заменяется на тот, который указан в параметре ClientsLogs. Описания логов разделяются запятыми или пробелами. В случае задания в параметре больше шести файлов отчета 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 = {значение в часах}	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 Control Agent

Компонент **Dr.Web Control Agent** (далее **Агент**) 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 антивирусную статистику.

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

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

Пожалуйста, обратите внимание, что если в конфигурационном файле компонента указано несколько параметров с одним именем, то **Агент** их объединяет через запятую.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:

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

Пример:

```
to: user@host cont \
```

```
modifier/LocalRules=select mime.headers "X-
Spam-Level" "\\*\\*\\*", \
# 3 и более звездочек
if found,\
select mime.headers Subject ".*",\
replace "[SPAM]" "^",\
endif
```

При задании значений параметров и правил в конфигурационных файлах можно использовать обратный слэш "\ ". В этом случае **Агент** объединит в одну строку все строки, разделённые с помощью обратного слэша. Использование пробела после слэша не допускается.

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 таком режиме центральной защиты не требует установки дополнительного программного обеспечения или удаления **Dr.Web для почтовых серверов UNIX**.

Для обеспечения этой возможности, **Агент** 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 **Агент** полностью управляется с защищаемого компьютера.

- Режиме центральной защиты (enterprise mode), когда защитой компьютера управляет сервер центральной 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 центрального сервера защиты. Персональный лицензионный ключевой файл на локальном компьютере не используется.

Чтобы использовать режим центральной защиты

1. Свяжитесь с системным администратором вашей сети чтобы получить файл с открытым ключом и параметры соединения с центральным сервером защиты.
2. В конфигурационном файле **Агента** (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 Enterprise Server** (далее - **Enterprise Сервер**). Если вы - администратор сети, вы можете найти этот файл в соответствующей директории на **Enterprise Сервере**.
   - Укажите IP-адрес или имя узла сервера **Enterprise Сервера** в параметре `ServerHost`.
   - Укажите номер порта для связи с **Enterprise Сервером** параметре `ServerPort`.
3. Чтобы включить режим центральной защиты, установите **Yes** в качестве значения параметра `UseEnterpriseMode`.

В режиме центральной 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 центрального сервера защиты. Персональный лицензионный ключевой файл на локальном компьютере не используется.

Для работы **Агента** в режиме центральной защиты должен быть установлен пакет `drweb-agent-es`.

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

Чтобы настройки модулей интеграции с MTA drweb-courier, drweb-cgp-receiver вступили в силу после обновления на сервере центральной 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 **Агента** (по умолчанию, `%etc_dir/agent.conf`) установлены корректно.
2. Установите **No** в качестве значения параметра `UseEnterpriseMode` секции `[EnterpriseMode]` конфигурационного файла **Агента**.

   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 центральной защиты, не могут быть использованы в этом режиме.

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

Ввиду особенностей реализации, единовременное 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 другую директорию файлы `%etc_dir/agent/drweb-cc.amc` и `%etc_dir/agent/drweb-spider.amc`.

Рекомендуется сохранить эти файлы в качестве резервной копии в директории, отличной от `%etc_dir/agent`, если в дальнейшем вы планируете перевести 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 директорию `%etc_dir/agent/` и следуйте инструкциям, представленным в руководстве пользователя **Антивируса Dr.Web для Linux**.

Параметры командной строки

Dr.Web Control Agent, в дополнение к параметрам командной строки, поддерживаемым всеми модулями программного комплекса, допускает использование следующих параметров:

- `-h, --help` – краткая справка по параметрам командной строки;
- `-v, --version` – вывод информации о текущей версии **Агента**;
- `-u, --update-all` – запуск процесса обновления для всех компонентов;
- `-f, --update-failed` – запуск процесса обновления для компонентов, которые не удалось обновить в штатном режиме;
- `-C, --check-only` – проверка конфигурации **Агента**. Данный параметр командной строки не может быть использован при запущенном **Агенте**.
- `-c <путь к файлу>, --conf <путь к файлу>` – использование альтернативного файла конфигурации;
- `-d, --droppwd` – сбросить регистрационную информацию (имя пользователя и пароль) **Enterprise Сервера**. При следующей попытке соединения с **Enterprise Сервером**, будет запущен процесс регистрации новой станции;
- `-p, --newpwd` – смена пользовательского имени и пароля на сервере центральной защиты;
- `-s <путь к файлу>, --socket <путь к файлу>` – использование альтернативного сокета;
- `-P <путь к файлу>, --pid-file <путь к файлу>` – PID-файл **Агента**;
- `-e <название приложения>, --export-config <название приложения>` – экспорт конфигурации приложения, указанного в аргументе, на **Enterprise Сервер**. В качестве аргумента используется имя приложения, указанное в заголовке секции `Application "имя приложения"` соответствующего amc-файла. Данный параметр командной строки не может быть использован при запущенном Агенте. Также данный 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`. Устройство конфигурационного файла и краткое описание его параметров даны в разделе Конфигурационные файлы.

Секция [Logging]

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

Секция [Logging]

Level = {Quiet | Error | Alert | Info | Debug}	Устанавливает уровень подробности ведения протокола работы компонента. Значение по умолчанию: **Level** = Info
IPCLevel = {Quiet | Error | Alert | Info | Debug}	Устанавливает уровень подробности протокола работы библиотеки IPC. Значение по умолчанию: **IPCLevel** = Error
SyslogFacility = {Daemon | Local0 .. Local7 | Kern | User | Mail}	Тип подсистемы, через которую системный сервис `syslogd`, ведущий протоколирование, выдает сообщения о событиях (более подробную информацию вы можете получить в документации по `syslog`). Значение по умолчанию: **SyslogFacility** = Daemon

FileName = {строка}	Имя файла отчета. В качестве имени можно указать syslog, тогда отчет будет вестись средствами системного сервиса syslogd. При использовании syslogd нужно обратить внимание на параметры **SyslogFacility**, **IPCLevel**, и **Level**. Поскольку syslogd имеет несколько файлов для протоколирования разных событий и разных степеней их важности, то, основываясь на этих трех параметрах и содержимом конфигурационного файла syslogd (обычно /etc/syslogd.conf), можно определить, куда будет писаться отчет программы.
	Значение по умолчанию: **FileName** = syslog

Секция [Agent]

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

Секция [Agent]

MetaConfigDir = {путь к директории}	Расположение файлов мета-конфигурации drweb-agent. В файлах мета-конфигурации описываются особенности взаимодействия **Агента** 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 = {Yes | No}	Значение Yes данного параметра, указывает модулю drweb-agent, что в составе программного комплекса используется **Монитор**. Значение по умолчанию: **UseMonitor** = Yes
MonitorAddress = {адрес}	Сокет, через который **Агент** взаимодействует с **Монитором** (значение параметра должно совпадать со значением параметра **Address** конфигурационного файла **Монитора**). Значение по умолчанию: **MonitorAddress** = local:%var_dir/ipc/.monitor
MonitorResponseTime = {время в секундах}	Максимальное время отклика модуля drweb-monitor. Если в течение этого времени от него не поступает реакции, то предполагается, что он не запущен, и **Агент** больше не предпринимает попыток взаимодействия с **Монитором**. Значение по умолчанию: **MonitorResponseTime** = 5

PidFile = {путь к файлу}	Путь к файлу, в который записывается PID 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 модуль drweb-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 = {численное значение}	Количество одновременных потоков drweb-agent. Параметр управляет максимальным количеством одновременных подключений к модулям, передающим **Агенту** 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 = {время в секундах}	Максимальное время установления соединения между **Агентом** 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 = {Yes | No}	При значении Yes данного параметра drweb-agent работает в режиме Enterprise, при значении No – в режиме Standalone. Значение по умолчанию: **UseEnterpriseMode** = No
ComputerName = {имя компьютера}	Название компьютера в антивирусной сети. Значение по умолчанию: **ComputerName** =
VirusbaseDir = {путь к директории}	Путь к вирусным базам. Значение по умолчанию: **VirusbaseDir** = %var_dir/bases
PublicKeyFile = {путь к файлу}	Путь к файлу открытого ключа для доступа к **Enterprise Серверу**. Значение по умолчанию:

	PublicKeyFile = %bin_dir/drwcsd.pub
ServerHost = {IP-адрес}	IP-адрес **Enterprise Сервера**. Значение по умолчанию: **ServerHost** = 127.0.0.1
ServerPort = {номер порта}	Порт доступа к **Enterprise Серверу**. Значение по умолчанию: **ServerPort** = 2193
CryptTraffic = {Yes | Possible | No}	Шифрование трафика между сервером **Enterprise Сервером** и модулем drweb-agent. Значение по умолчанию: **CryptTraffic** = possible
CompressTraffic = {Yes | Possible | No}	Сжатие трафика между **Enterprise Сервером** и модулем drweb-agent. Значение по умолчанию: **CompressTraffic** = possible
CacheDir = {путь к директории}	Путь к директории, в которой хранятся служебные файлы: конфигурационные файлы компонентов и файлы, содержащие информацию о правах каждого из приложений, на случай, если **Enterprise Сервер** по какой-либо причине окажется недоступен, файлы с регистрационной информацией на **Enterprise Сервере** и т.п. Значение по умолчанию: **CacheDir** = %var_dir/agent

Секция [StandaloneMode]

Настройки `drweb-agent` для одиночного режима работы.

Секция [`StandaloneMode`]

`StatisticsServer` = {адрес сервера}	URL сервера вирусной статистики. Если URL сервера не указан, то статистика не будет отправляться. Значение по умолчанию: **`StatisticsServer`** = `stat.drweb.com:80/update`
`StatisticsUpdatePeriod` = {время в минутах}	Период обновления статистической информации. Не может быть меньше 5 минут Значение по умолчанию: **`StatisticsUpdatePeriod`** = `10`
`StatisticsProxy` = {адрес прокси-сервера}	IP-адрес или имя хоста прокси-сервера для вирусной статистики. Обратите внимание, что если значение параметра не задано, используется значение переменной окружения `http_proxy`. **Пример**: **`StatisticsProxy`** = `localhost:3128` Значение по умолчанию: **`StatisticsProxy`** =
`StatisticsProxyAuth` = {параметры аутентификации}	Имя пользователя и пароль для доступа к прокси-серверу. **Пример**: **`StatisticsProxyAuth`** = `test:`

	testpwd Значение по умолчанию: **StatisticsProxyAuth** =
UUID = {идентификатор}	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). Значение по умолчанию: **LicenseFile** = %bin_dir/drweb32.key
ProtectedEmails = {lookups}	Список защищаемых почтовых адресов. Их можно задать непосредственно, либо указать путь к файлу, в котором они перечислены. Значение по умолчанию: **ProtectedEmails** = file:%etc_dir/email.ini

Секция [Update]

В этой секции собраны параметры, относящиеся к процессу

обновления компонентов программного комплекса **Dr.Web для почтовых серверов UNIX** через **Enterprise Сервер** (подробнее см. в руководстве администратора **Dr.Web ESS**):

Секция [Update]

CacheDir = {путь к директории}	Директория, в которой **Агент** временно сохраняет загруженные файлы обновлений. Значение по умолчанию: **CacheDir** = %var_dir/updates/cache
Timeout = {время в секундах}	Максимальное время обработки **Агентом** полученных обновлений. Если указано значение 0, время обработки не ограничивается. Значение по умолчанию: **Timeout** = 120
RootDir = {путь к директории}	Путь к корневой директории. Значение по умолчанию: **RootDir** = /

Запуск

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

В процессе запуска **Агента** 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), **Агент** запускается в режиме Enterprise. В противном случае, если в файле настроек заданы параметры секции [Standalone], **Агент** запускается в одиночном режиме. Если параметры секции [Standalone] также не заданы, то загрузка **Агента** 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, он может быть создан только когда директория, его содержащая, доступна на запись и чтение пользователю, с чьими правами работает модуль drweb-agent. Если ни один сокет не может быть создан, загрузка **Агента** прекращается.

Дальнейший процесс загрузки **Агента** зависит от того, в каком режиме он работает.

Если **Агент** работает в режиме Enterprise:

- производится соединение с сервером централизованной защиты **Dr.Web**. Если при первом подключении сервер недоступен, либо **Агенту** не удалось авторизоваться, **Агент** завершает свою работу. Если ранее **Агент** уже работал с данным сервером, но в данный момент он недоступен (например, в случае проблем с соединением), **Агент** 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 **Агент** готов к работе.

Если **Агент** работает в режиме `Standalone`:

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

Взаимодействие с компонентами программного комплекса

Взаимодействие с компонентами программного комплекса осуществляется с помощью amc-файлов. В этих файлах описывается конфигурация компонентов и параметры, значения которых **Агент** выдает компонентам.

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 **Enterprise Сервере**;
- **`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`).

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 **Dr.Web Enterprise Security Suite**:

- первоначальная установка и настройка почтового сервера UNIX в уже работающей системе **Dr.Web ESS**;
- встраивание работающего почтового сервера UNIX с установленным и настроенным программным комплексом **Dr.Web для почтовых серверов UNIX** в систему **Dr.Web ESS**.

Для того, чтобы программный комплекс **Dr.Web для почтовых серверов UNIX** мог работать в составе **Dr.Web Enterprise Security Suite**, необходимо настроить компоненты **Агент** и **Монитор** для работы в режиме `Enterprise` и зарегистрировать комплекс на сервере **Dr.Web Enterprise Server** (далее - **Enterprise Сервер**).

В соответствии с политикой подключения новых станций (подробнее см. руководство администратора **Dr.Web Enterprise Security Suite**), подключить **Dr.Web для почтовых серверов UNIX** к **Enterprise Серверу** 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 **Агента** и **Монитора**.

Для Агента

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

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

Для Монитора

В секции `[Monitor]` конфигурационного файла **Монитора** `%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` **Агент** запрашивает регистрационные данные (идентификатор станции и пароль) у **Enterprise Сервера**;
- если на **Enterprise Сервере** установлен режим "**Ручное подтверждение доступа**" (режим по умолчанию, см. руководство администратора **Dr.Web Enterprise Security Suite**), то администратору в течение одной минуты с момента запроса необходимо подтвердить регистрацию новой станции через веб-интерфейс **Центра управления Dr.Web**;

- после первого подключения **Агент** записывает хэш идентификатора станции и пароля пользователя в файл с названием `pwd`. Данный файл создаётся в директории, заданной 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 **Enterprise Серверу**;
- удаление файла с паролем приведет к повторному запросу регистрационных данных у ESS-сервера при следующем запуске **Агента**.

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 **Dr.Web Enterprise Security Suite**);
- Запустите **Агент** 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 директории, путь к которой 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 **Enterprise Серверу**;
- Удаление файла с паролем приведет к необходимости повторить процедуру регистрации при следующем запуске **Агента**.

Задание конфигурации компонентов через веб-интерфейс сервера

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 Enterprise Security Suite** 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 **Агент** запрашивает и получает конфигурацию от сервера централизованной защиты.

Экспорт существующей конфигурации на сервер

При помощи работающего в режиме **Enterprise** **Агента** возможно автоматически экспортировать конфигурацию компонентов на ESS-сервер. Для этого необходимо экспортировать конфигурацию параметром командной строки `--export-config` (или `-e`) с обязательным 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 **Монитора** установите флаги `Daemon` и `Maild` для запуска соответствующих компонентов комплекса;

- Запустите **Монитор** на локальной станции:

  `# /etc/init.d/drweb-monitor start` – для Linux и Solaris;

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

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**, если **Агент** работает в режиме `Enterprise`. Для соединения **Агента** с сервером статистики **"Доктор Веб"** необходим идентификатор пользователя – `UUID`. По умолчанию в качестве `UUID` используется `md5` ключевого файла. Также вы можете получить персональный `UUID`, обратившись в службу поддержки. Такой `UUID` прописывается в файле конфигурации **Агента**.

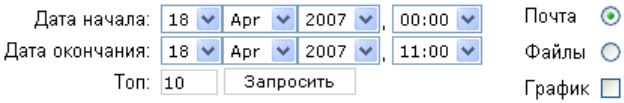
По адресу http://stat.drweb.com/ можно ознакомиться как с результатами обработки статистических данных по вашему серверу, так и с обобщенной статистической информацией по всем серверам, обслуживаемым антивирусом **Dr.Web для UNIX** либо программным комплексом **Dr.Web для почтовых серверов UNIX** с подключенным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

Результаты обработки содержат сведения о наиболее часто обнаруживаемых вирусах (количество обнаружений и процент от общей суммы) 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 абсолютной и процентной форме. Внешний вид страницы может различаться в зависимости от используемого веб-браузера.



18.04.2007 00:00 - 18.04.2007 11:00		
1	Win32.HLLM.Netsky.35328	5351 (20.61%)
2	Win32.HLLM.Beagle	4566 (17.58%)
3	Win32.HLLP.Sector	3256 (12.54%)
4	Win32.HLLM.Netsky.based	2298 (8.85%)
5	Win32.Hazafi.30720	1660 (6.39%)
6	Win32.HLLM.MyDoom.based	1652 (6.36%)
7	Win32.HLLM.Limar.based	903 (3.48%)
8	Trojan.Spambot	808 (3.11%)
9	Win32.HLLM.Graz	614 (2.36%)
10	Win32.HLLM.Beagle.pswzip	565 (2.18%)

Рис. 15. 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**.
- Установите флажок **График**, если вы хотите получить статистическую информацию в графическом виде.

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/<UID>, где `<UID>` - это md5 ключевого файла пользователя. Страница персональной

статистики имеет формат, полностью аналогичный формату страницы обобщенной статистики.

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

Ниже приводится сокращенный пример такого файла:

```
<drwebvirustop period="24" top="2" user="<UID>"
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** (далее **Монитор**) 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, **Монитор** повторяет попытку. Количество попыток и время между ними определяются настройками компонента.

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

Режимы работы

При необходимости продукты компании **"Доктор Веб"** могут быть подключены к корпоративной или частной антивирусной сети, управляемой комплексом **Dr.Web Enterprise Security Suite**. Работа в таком режиме центральной защиты не требует установки дополнительного программного обеспечения или удаления **Dr.Web для почтовых серверов UNIX**.

Для обеспечения этой возможности, **Монитор** 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, **Монитор** полностью управляется с защищаемого компьютера, а все необходимые модули **Dr.Web** запускаются в соответствии с локальными настройками **Монитора**.
- Режиме центральной защиты (enterprise mode), когда защитой компьютера управляет сервер центральной 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 центрального сервера защиты. Персональный лицензионный ключевой файл на локальном компьютере не используется.

Чтобы использовать режим центральной защиты

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

В режиме центральной 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 центрального сервера защиты. Персональный лицензионный ключевой файл на локальном компьютере не используется.

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

Чтобы использовать одиночный (standalone) режим

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

   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 центральной защиты, не могут быть использованы в этом режиме.

Параметры командной строки

Dr.Web Monitor допускает использование следующих параметров:

- `-h, --help` – вывод краткой справки по параметрам командной строки **Монитора**;
- `-v, --version` – вывод информации о текущей версии **Монитора**;
- `-u, --update` – запуск процесса обновления;
- `-C, --check-only` – проверка конфигурации **Монитора**;
- `-A, --check-all` – проверка конфигурации всех компонентов;
- `-c <путь к файлу>, --conf <путь к файлу>` – использование альтернативного конфигурационного файла;
- `-r, --run приложение1[,приложение2]` – запуск приложений.

Пример:

```
-r AGENT, MAILD
```

Конфигурационный файл

Настройки компонента **Dr.Web Monitor** задаются отдельным конфигурационным файлом `%etc_dir/monitor.conf`. Устройство конфигурационного файла и краткое описание его параметров даны в разделе Конфигурационные файлы.

Секция [Logging]

В секции `[Logging]` собраны параметры, управляющие ведением протоколов работы компонента **Dr.Web Monitor**

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

Секция [Logging]

Level = {Quiet | Error | Alert | Info | Debug}	Устанавливает уровень подробности ведения протокола работы компонента. Значение по умолчанию: **Level** = Info
IPCLevel = {Quiet | Error | Alert | Info | Debug}	Устанавливает уровень подробности протокола работы библиотеки IPC. Значение по умолчанию: **IPCLevel** = Error
SyslogFacility = {Daemon | Local0 .. Local7 | Kern | User | Mail}	Тип подсистемы, через которую системный сервис syslogd, ведущий протоколирование, выдает сообщения о событиях (более подробную информацию вы можете получить в документации по syslog). Значение по умолчанию: **SyslogFacility** = Daemon
FileName = {строка}	Имя файла отчета. В качестве имени можно указать syslog, тогда отчет будет вестись средствами системного сервиса syslogd. При использовании syslogd нужно обратить внимание на параметры **SyslogFacility**, **IPCLevel**, и **Level**. Поскольку syslogd имеет несколько файлов для протоколирования разных событий и разных степеней их важности, то, основываясь на этих трех параметрах и содержимом конфигурационного файла syslogd (обычно /etc/syslogd.conf), можно определить, куда будет писаться отчет программы.

	Значение по умолчанию: **FileName** = syslog

Секция [Monitor]

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

Секция [Monitor]

RunForeground = {Yes | No}	Значение Yes запрещает **Монитору** 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 = {имя пользователя}	Имя пользователя, с правами которого запускается **Монитор**. Пожалуйста, обратите внимание, что при работе программного комплекса в режиме SMTP-прокси значение данного параметра должно быть установлено в root.
	Значение по умолчанию: **User** = drweb
Group = {название группы}	Имя пользовательской группы, с правами которой запускается **Монитор**. Пожалуйста, обратите внимание, что при работе программного комплекса в режиме SMTP-прокси значение данного параметра должно быть установлено в root.
	Значение по умолчанию:

	Group = drweb
PidFileDir = {путь к директории}	Имя директории, где содержится файл, в который при запуске **Монитора** записывается информация об идентификаторе его процесса (PID). Значение по умолчанию: **PidFileDir** = %var_dir/run/
ChDir = {путь к директории}	Смена активной директории при запуске **Монитора**. Если значение параметра задано, то при запуске **Монитор** делает активной директорию, указанную в значении этого параметра. Если значение параметра не задано, то смена активной директории не происходит. Значение по умолчанию: **ChDir** = /
MetaConfigDir = {путь к директории}	Путь к директории с файлами мета-конфигурации. В этих файлах задаются параметры работы **Монитора** 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 **Монитор** взаимодействует с другими модулями антивируса. Значение по умолчанию: **Address** = local:%var_dir/ipc/.monitor

Timeout = {время в секундах}	Максимальное время установления соединения между **Монитором** и другими компонентами программного комплекса. Значение по умолчанию: **Timeout** = 5
TmpFileFmt = {текст}	Шаблон имени временных файлов **Монитора**. Формат шаблона: путь_к_файлу.XXXXXX, где X - произвольные буквы и цифры в именах создаваемых временных файлов. Значение по умолчанию: **TmpFileFmt** = %var_dir/msgs/tmp/monitor.XXXXXX
RunAppList = {текст}	Список модулей, запускаемых **Монитором**. 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 **Монитор** не сможет запуститься сам и запустить остальные компоненты. Значение по умолчанию: **RunAppList** = AGENT
UseEnterpriseMode = {Yes | No}	При значении Yes данного параметра список модулей, запускаемых **Монитором**, берется не из параметра **RunAppList**, а от модуля drweb-agent. Значение по умолчанию: **UseEnterpriseMode** = No

RecoveryTimeList = {время в секундах}	Временные промежутки между попытками перезапуска "зависших" приложений. 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 = {строка}	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 = {lookups}	Сокет, через который **Монитор** взаимодействует с **Агентом** (значение параметра должно совпадать со значением параметра **Address** конфигурационного файла **Агента**). Значение по умолчанию: **AgentAddress** = local:%var_dir/ipc/.agent
AgentResponseTime = {время в секундах}	Максимальное время отклика модуля drweb-agent. Если в течение этого времени от модуля не поступает ответа, то **Монитор** перезапускает его. Если указано значение 0, время отклика не ограничивается. Значение по умолчанию: **AgentResponseTime** = 5

Запуск

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

В процессе запуска **Монитора** (при установках по умолчанию) осуществляются следующие действия:

- производится поиск и загрузка конфигурационного файла; если файл не найден, то загрузка прекращается;
- **Монитор** переходит в режим демона, поэтому сообщения о дальнейших проблемах не могут быть выведены на терминал и выводятся только в файл отчета;
- 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 когда директория, его содержащая, доступна 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 **Монитора**. 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, **Монитор** пытается запустить его повторно. Если все попытки **Монитора** загрузить модуль окончились неудачей, **Монитор** выгружает все уже загруженные модули и завершает свою работу. Обо всех проблемах с запуском модулей программного комплекса **Монитор** сообщает одним из доступных ему способов (записью в файл протокола, сообщением электронной почты, запуском произвольной программы). Способы оповещения, используемые для разных модулей, задаются в файле мета-конфигурации **Монитора**.

Для успешного запуска Монитора в автоматическом режиме:

- либо в файле `%etc_dir/drweb-monitor.enable` 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).

Взаимодействие с компонентами программного комплекса

Взаимодействие с компонентами программного комплекса осуществляется с помощью mmc-файлов. В этих файлах описывается состав компонентов, расположение бинарных файлов, порядок их запуска и параметры запуска.

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 (могут быть заключены в кавычки), максимальное время, отводимое на запуск компонента, максимальное время для остановки, 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 лог). Права для запуска - указывают группу и пользователя, с чьими правами будет запускаться компонент.

Пример mmc-файла Dr.Web Daemon для Linux:

```
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 для почтовых серверов UNIX

Программный комплекс **Dr.Web для почтовых серверов UNIX** – это группа совместно работающих программных модулей, которые в совокупности могут исполнять функции прокси-сервера для протоколов SMTP и LMTP и фильтра для большого набора поддерживаемых почтовых систем (Sendmail, Postfix, Exim, CommuniGate Pro, Courier, Zmailer и Qmail) со своими собственными настройками, статистикой, отчетами и карантином. Компоненты комплекса совместно обрабатывают почтовые сообщения.

Обработка происходит по следующему алгоритму:

1. Сообщения, поступающие от почтовых систем либо по протоколам SMTP/LMTP, принимаются компонентом **Receiver**, который передает их компоненту `drweb-maild`, отвечающему за проверку почтовых сообщений.

   В зависимости от используемых почтовых систем и протоколов функции компонента **Receiver** выполняют разные модули (`drweb-receiver`, `drweb-milter`, `drweb-cgp-receiver` и т.п.), причем поддерживается одновременная работа нескольких модулей компонента **Receiver**, что позволяет получать и обрабатывать почту сразу из нескольких источников. Некоторые модули компонента **Receiver** поддерживают возможность модификации/отправления полученных сообщений, принимая результаты проверки писем от компонента `drweb-maild`. (Например, такой возможностью обладает модуль `drweb-milter`, что позволяет ему возвращать почтовой системе Sendmail результат проверки писем до окончания SMTP-сессии.);

2. Модуль `drweb-maild` – основной компонент системы обработки почты. Он производит MIME-разбор сообщений, передает письма на обработку подключаемым модулям и отвечает за хранение писем в базе данных. Результаты проверки отправляются либо компоненту **Receiver** (если

существует такая возможность - например, еще не истекло время ожидания результата проверки), либо компоненту **Sender**;

3. Обработка писем производится подключаемыми к `drweb-maild` модулями антивирусной проверки и антиспам-защиты. Эти плагины могут запускаться и выгружаться пользователями в произвольные моменты времени, без остановки модуля `drweb-maild`. Обработка писем осуществляется в порядке, определяемом пользователем системы. Некоторым плагинам для работы необходима поддержка базы данных;
4. Компонент **Sender** отвечает за отправление писем либо напрямую в различные почтовые системы, 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.). Компонент **Sender** может получать запросы на отправление писем от компонентов `drweb-maild`, `drweb-notifier` и `drweb-monitor`;
5. Модуль `drweb-notifier` отвечает за создание и отправку всех отчетов, формируемых в процессе работы комплекса. Запрос на отправку отчетов могут отправлять как плагины (например, при обнаружении вируса), так и другие компоненты системы. Например, модуль `drweb-maild` может посылать запрос на создание общего отчета со статистикой работы всех подключенных плагинов, а компонент **Sender** может посылать запрос на формирование отчета DSN о невозможности отправить письмо. Отчеты могут рассылаться отправителям писем, их получателям и администратору системы.
6. Модуль `drweb-agent` обеспечивает возможность работы системы обработки почты как в автономном режиме, так и в режиме интеграции с программным комплексом **Dr.Web Enterprise Security Suite**. В последнем случае все компоненты системы (кроме `drweb-monitor`) получают свои конфигурационные данные через модуль `drweb-agent`, и поэтому он должен запускаться перед другими компонентами.

   Кроме того, модуль `drweb-agent` отвечает за получение

лицензионных ключевых файлов и сбор статистической информации о работе компонентов системы: имена обнаруженных блокируемых объектов, общий объем проверенной информации и т.д.

7. Модуль `drweb-monitor` является вспомогательным компонентом, запускающим и останавливающим модули системы в заданном порядке, а также контролирующим их работу. В случае сбоя в работе какого-либо модуля системы, `drweb-monitor` осуществляет его перезапуск, а также (если это задано настройками) может уведомлять об этом администратора системы.

В настоящий момент в состав программного комплекса **Dr.Web для почтовых серверов UNIX** входят следующие плагины для фильтрации почты:

- `drweb` – плагин, осуществляющий антивирусную проверку почты с помощью компонента **Dr.Web Daemon** (**Демон**). Сообщения передаются на проверку **Демону** уже разобранными на части, поэтому поддержка MIME-разбора не требуется;
- `headersfilter` – плагин, осуществляющий фильтрацию сообщений по заголовкам. При задании правил фильтрации можно использовать регулярные выражения (синтаксис Perl);
- `vaderetro` – плагин, осуществляющий спам-фильтрацию почты через свою собственную библиотеку **VadeRetro**. Эта библиотека динамически обновляется, что позволяет сохранять стабильно высокое качество фильтрации. **VadeRetro** отличается очень высокой скоростью проверки писем;
- `modifier` – плагин, 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.

Параметры командной строки

Как и для любых UNIX программ, для компонентов программного комплекса **Dr.Web для почтовых серверов UNIX** предусмотрены параметры командной строки. Формат командной строки для компонентов программного комплекса следующий:

```
название_компонента [параметры] сокет_агента
```

где:

- `название_компонента` – название компонента программного комплекса **Dr.Web для почтовых серверов UNIX**;
- `параметры` – необязательные параметры командной строки;
- `сокет_агента` – сокет, через который компоненты программного комплекса **Dr.Web для почтовых серверов UNIX** получают от **Dr.Web Agent** конфигурационную информацию.

Полный список параметров можно получить, запустив компонент с параметром `-h` либо `--help`. В текущей версии **Dr.Web для почтовых серверов UNIX** компоненты поддерживают следующие параметры командной строки:

- `-v, --version` – информация о текущей версии компонента **Dr.Web для почтовых серверов UNIX**;
- `-l <уровень>, --level <уровень>` – уровень детализации протокола запуска компонента **Dr.Web для почтовых серверов UNIX** (значение по умолчанию: `info`);
- `-t <значение в секундах>, --timeout <значение в секундах>` – максимальное время ожидания получения конфигурационных данных;
- `--component arg` – имя компонента, под которым он будет обращаться к **Dr.Web Agent** для получения конфигурации;

- `--log-name arg` – имя, под которым компонент будет выводить сообщения в лог;
- `--unique-id arg` – уникальный идентификатор для компонентов **Receiver**/**Sender**. Данная настройка позволяет осуществлять работу с несколькими **Receiver**'ами и **Sender**'ами. Для этого каждый новый **Receiver**/**Sender** должен запускаться со своим уникальным идентификатором. Для отправки письма будет выбираться **Sender** с тем же идентификатором, что и у **Receiver**'а, или **Sender** по умолчанию, если соответствующий **Receiver** не был найден. Список доступных **Sender**'ов переинициализируется через сигнал `SIGHUP`;
- `--check-only` – компонент запускается в режиме проверки конфигурации. Для корректной функциональности должен быть запущен **Dr.Web Agent**. При успешной проверке на консоль выводится сообщение:

  ```
  Options OK
  ```

  при неудаче выводится описание проблемы и сообщение:

  ```
  Options ERROR
  ```

- `--check-all` – запущенный с этим параметром, **Dr.Web Monitor** проверит не только свои настройки, но и конфигурацию всех модулей комплекса.

Пример:

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

Запускает компонент **Dr.Web MailD** с временем ожидания конфигурационных данных в 30 секунд и сокетом **Агента**, располагающимся по адресу `local:%var_dir/ipc/.agent`.

Обрабатываемые сигналы

Все постоянно находящиеся в памяти модули программного комплекса **Dr.Web для почтовых серверов UNIX** поддерживают обработку следующих сигналов:

- SIGHUP – при получении этого сигнала модули перечитывают свои конфигурационные файлы. Если этот сигнал получает компонент **Dr.Web Monitor**, то конфигурационные файлы перечитываются всеми запущенными модулями;
- SIGINT и SIGTERM – при получении любого из этих сигналов модули завершают свою работу.

Некоторые модули программного комплекса могут поддерживать обработку дополнительных сигналов:

- компоненты **Receiver** и **Sender** при получении сигнала SIGALRM производят проверку внутренней структуры директорий на предмет "потерянных" по тем или иным причинам сообщений. Если такие сообщения находятся, производится попытка их отправить;
- компонент **Sender**, получив сигнал SIGUSR2, производит попытку отправить все сообщения, находящиеся во внутренней очереди и ожидающие отправки;
- компонент **Receiver**, получив сигнал SIGUSR1, сохраняет в файл статистику работы проверок адресов;
- все компоненты при получении сигнала SIGUSR1 сбрасывают в директорию, указанную в значении параметра **BaseDir** секции [General] конфигурационного файла **Dr.Web MailD**, файлы со статистикой по работе динамических потоков и постоянных соединений.

Внутренняя статистика работы

Статистика по работе пулов потоков и постоянных соединений, связанных с данным пулом, накапливается только если ее явно включить в настройках пула потока (**InPoolOptions** и **OutPoolOptions** конфигурационного файла **Dr.Web MailD**), установив дополнительную настройку **stat** = yes.

Пример:

InPoolOptions = auto, **stat** = yes

Имена файлов, формирующиеся по сигналу `SIGUSR1`, имеют шаблон:

- `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 без части "`drweb-`".
- `callback` – указывается для `callback` интерфейса **Receiver**'а.
- `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 старта сбора статистики, время сброса статистики в файл и временной период отчета.

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 число 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.

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 директории `%etc_dir`. В файле `maild_MTA.conf` содержатся общие **Dr.Web MailD**, в файле `agent.conf` – настройки **Dr.Web Agent**, а в файле `monitor.conf` – настройки **Dr.Web Monitor**.

Базовую настройку **Dr.Web для почтовых серверов UNIX** (при условии расположения всех файлов программного комплекса в директориях по умолчанию) можно осуществить с помощью скрипта `configure.pl` , по умолчанию располагающегося в директории `%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**.

Конфигурационный файл

Настройки компонента **Dr.Web MailD** задаются отдельным конфигурационным файлом `%etc_dir/maild_MTA.conf`. Устройство конфигурационного файла и краткое описание его параметров даны разделе Конфигурационные файлы.

Секция [General]

В секции `[General]` собраны общие настройки работы **Dr.Web MailD**:

BaseDir = {путь к директории}	Основная рабочая директория, в которой содержатся сокеты, база данных и другие файлы. В текущей версии этот параметр не может быть изменен при перезапуске по сигналу SIGHUP. Значение по умолчанию: **BaseDir** = %var_dir
MaxTimeoutForThreadActivity = {время}	Максимальное время закрытия одного потока. Параметр используется при перезапуске, а также при завершении работы. Общее максимальное время завершения работы рассчитывается следующим образом: количество пулов умножается на значение параметра **MaxTimeoutForThreadActivity** и к результату прибавляется некоторая временная константа. Значение по умолчанию: **MaxTimeoutForThreadActivity** = 30s
IpcTimeout = {время}	Максимальное время установки соединения между компонентами. Значение по умолчанию: **IpcTimeout** = 40s
Hostname = {текст}	Имя хоста, на котором работает **Dr.Web для почтовых серверов UNIX**. Значение по умолчанию: **Hostname** =

Секция [Logging]

В секции [Logging] собраны параметры, управляющие ведением протоколов работы основных модулей программного

комплекса **Dr.Web для почтовых серверов UNIX**:

Level = {Quiet | Error | Alert | Info | Debug}	Определяет уровень подробности протокола работы. Значение по умолчанию: **Level** = Info
IpcLevel = {Quiet | Error | Alert | Info | Debug}	Уровень подробности протокола работы библиотеки IPC. Значение по умолчанию: **IpcLevel** = Alert
SyslogFacility = {Daemon | Mail | Local0..Local7}	Тип подсистемы, через которую системный сервис syslogd, ведущий протоколирование работы **Dr.Web для почтовых серверов UNIX** и его подсистем, выдает сообщения о событиях (более подробную информацию вы можете получить в документации по syslogd). Значение по умолчанию: **SyslogFacility** = Mail
FileName = {syslog | путь к файлу}	Путь к файлу протокола или "syslog", если отчет о работе ведется с помощью системной службы syslogd. Значение по умолчанию: **FileName** = syslog

Информация может выводиться в отчет как через демон syslogd, так и в обычный файл. Формат каждой выводимой строки имеет следующий вид (в случае вывода через демон syslogd):

```
'['tid']' name[.sub] level [sid(/mta-id)] text
```

где:

- `tid` – идентификатор потока, выводящего строку;
- `name` – название компонента, производящего вывод - например, название плагина или модуля;
- `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 `lookups`;
- `control` – служба поддержки интерактивного управления;
- `parser` – служба разбора шаблонов отчетов;
- `MRS` – служба приема сообщения по SMTP/LMTP протоколу;
- `smtp` – служба отправления письма по SMTP протоколу;
- `lmtp` – служба отправления письма по LMTP протоколу;
- `pipe` – служба отправления письма через pipe;
- `queue` – служба обработки внутренней очереди;
- `level` – уровень подробности протокола работы. Возможны следующие значения: `FATAL`, `ERROR`, `WARN`, `INFO`, `DEBUG`.
- `sid` – идентификатор сессии для сообщения, к которому относится данная строка лога. 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-прокси и из MTA удалось получить данную информацию;
- `client` – уникальный идентификатор Клиента, к которому привязано данное сообщение;
- `text` – собственно текст выводимого сообщения.

При запуске любого модуля устанавливается `INFO` уровень лога по умолчанию — до момента получения конфигурации от **Dr.Web Agent** и установки обозначенного в конфигурационном файле уровня. Иногда полезно увидеть `DEBUG` уровень загрузки модуля (например, для получения информации о загруженных от **Dr.Web Agent** параметрах) — для этого служит параметр командной строки `--level`, установив который в `debug`, можно получить требуемую информацию.

Секция [MySQL]

В секции `[MySQL]` собраны настройки взаимодействия **Dr.Web MailD** с базой данных 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 = {текст}	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. **Пример**:

	При использовании TCP-сокета: **Port** = tcp://1234 При использовании UNIX сокета: **Port** = unix:///path/to/socket Значение по умолчанию: **Port** =
Connections = {число}	Число одновременных подключений к базе данных MySQL. При значении 0 подключения будут создаваться по мере обращений к базе данных, что потребует дополнительного времени. Заранее открытые подключения могут в порядке очереди обслуживать запросы к базе данных без затрат времени на повторные подключения. Значение по умолчанию: **Connections** = 4
SizeLimit = {число}	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 (см. раздел Lookups). Значение по умолчанию: **SizeLimit** = 10
SkipDomains = {LookupsLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность. Значение параметра может быть указано в локальных настройках lookup. Значение по умолчанию:

	SkipDomains =
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"
```

Секция [PostgreSQL]

В секции [PostgreSQL] собраны настройки взаимодействия **Dr.Web MailD** с базой данных PostgreSQL:

ConnectionsString = {текст}	Строка с параметрами подключения к базе данных 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://postgresql.org/docs/8.3/static/libpq-connect.html. **Примеры**:

	ConnectionString = host=localhost port=5432 user=ai password=qwerty dbname=drweb **ConnectionString** = hostaddr=127.0.0.1:5432 dbname=mailddb user=mailddbuser password=Str0ngPaSSw0rd Значение по умолчанию: **ConnectionsString** =
SizeLimit = {число}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. При значении 0 ограничения отсутствуют. Значение по умолчанию: **SizeLimit** = 10
SkipDomains = {LookupsLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность. Значение параметра может быть указано в локальных настройках lookup. Значение по умолчанию: **SkipDomains** =
Lib = {путь к файлу}	Путь к библиотеке libpq.so. Значение по умолчанию: **Lib** = /usr/lib/libpq.so

Секция [Firebird]

В секции [Firebird] собраны настройки взаимодействия **Dr.**

Web MailD с базой данных 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 = {число}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. При значении 0 ограничения отсутствуют Значение по умолчанию: **SizeLimit** = 10

SkipDomains = {LookupsLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность. Значение параметра может быть указано в локальных настройках `lookup`. Значение по умолчанию: **SkipDomains** =
Lib = {путь к файлу}	Путь к библиотеке `libFBclient.so` Значение по умолчанию: **Lib** = /usr/lib/libFBclient.so

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 = {LookupsLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность. Значение параметра может быть указано в локальных настройках `lookup`. Значение по умолчанию: **SkipDomains** =

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 директории}	Путь к директории для хранения файлов блокировки Berkeley. Значение по умолчанию: **Environment** =
SizeLimit = {число}	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 = {LookupsLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка зачастую позволяет значительно снизить нагрузку на сервер и повысить производительность. Значение параметра может быть указано в локальных настройках `lookup`. Значение по умолчанию: **SkipDomains** =
Lib = {путь к файлу}	Путь к библиотеке `libdb.so`. Значение по умолчанию: **Lib** = /usr/lib/libdb.so

Секция [SQLite]

В секции `[SQLite]` собраны настройки взаимодействия **Dr.Web MailD** с базой данных SQLite:

Database = {путь к файлу}	Путь к файлу базы данных SQLite. Значение по умолчанию: **Database** =
SizeLimit = {число}	Максимальное количество строк, получаемых в ответ на один запрос к базе данных. При значении 0 ограничения отсутствуют Значение по умолчанию: **SizeLimit** = 1
SkipDomains = {LookupsLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка зачастую позволяет значительно снизить нагрузку на сервер и повысить производительность. Значение параметра может быть указано в локальных настройках `lookup`. Значение по умолчанию: **SkipDomains** =
Lib = {путь к файлу}	Путь к библиотеке `libsqlite3.so`. Значение по умолчанию: **Lib** = /usr/lib/libsqlite3.so
BusyTimeout = {время в миллисекундах}	Максимальное время, в течение которого **Dr.Web MailD** будет пытаться осуществить запись в базу данных. Значение по умолчанию: **BusyTimeout** = 2000

Секция [ODBC]

В секции `[ODBC]` собраны настройки взаимодействия **Dr.Web MailD** с базой данных 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 этот параметр не может быть изменен при перезапуске по сигналу SIGHUP. 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. Значение параметра может быть указано в локальных настройках lookup. Значение по умолчанию: **ConnectData** =

SizeLimit = {число}	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 = {LookupsLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка зачастую позволяет значительно снизить нагрузку на сервер и повысить производительность. Значение параметра может быть указано в локальных настройках lookup. Значение по умолчанию: **SkipDomains** =

Секция [Oracle]

В секции [Oracle] собраны настройки взаимодействия **Dr.Web MailD** с базой данных 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 этот параметр не может быть изменен при перезапуске по сигналу SIGHUP. Значение по умолчанию: **Lib** =

ConnectData = {текст}	Параметры Oracle-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, задать название DSN, который ссылается на нужную базу данных. Значение параметра может быть указано в локальных настройках lookup. Значение по умолчанию: **ConnectData** =
SizeLimit = {число}	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 = {LookupsLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка зачастую позволяет значительно снизить нагрузку на сервер и повысить производительность. Значение параметра может быть указано в локальных настройках lookup. Значение по умолчанию: **SkipDomains** =

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 этот параметр не может быть изменен при перезапуске по сигналу SIGHUP.
	Значение по умолчанию: **Lib** = /usr/lib/libldap_r.so

Обратите внимание, что в операционной системе FreeBSD версии 6.4/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 = {число}	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 LDAP-протокола. Для обеспечения защищенной передачи данных с помощью TLS/SSL должен использоваться LDAP-протокол не ниже версии 3. Значение параметра может быть указано в локальных настройках lookup. Значение по умолчанию: **Version** = 3
Bind = {Yes | No}	Необходимость привязки перед выполнением запросов. Для LDAP-протокола 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 = {число}	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 = {число}	Настройка LDAP_OPT_REFERRALS. Для установки данного параметра необходим LDAP-протокол версии не ниже 3. Значение параметра может быть указано в локальных настройках lookup. Значение по умолчанию:

	ChaseReferrals = 0
SkipDomains = {LookupsLite}	Список доменов, для которых не нужно выполнять запрос к базе данных. Данная настройка позволяет значительно снизить нагрузку на сервер и повысить производительность. Значение параметра может быть указано в локальных настройках lookup. Значение по умолчанию: **SkipDomains** =
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

Секция [MailBase]

В секции [MailBase] собраны настройки базы данных **Dr.Web MailD**:

MaxStoredMessages = {число}	Максимальное количество сообщений в базе. При значении 0 ограничения отсутствуют. Если количество писем в базе превышает число, заданное в этом параметре, производится очистка базы от самых старых писем до достижения нужного количества писем в базе. Уже отправленные письма сразу удаляются, еще не отправленные – отправляются и удаляются. Значение по умолчанию: **MaxStoredMessages** = 100000

MaxStorageSize = {размер в байтах}	Максимальный размер базы сообщений в байтах. При значении 0 ограничения отсутствуют. Если размер базы превышает предельный, производится очистка базы от самых старых писем до достижения нужного размера базы (см. описание параметра **MaxStoredMessages**). Значение по умолчанию: **MaxStorageSize** = 0
MaxPoolSize = {число}	Максимальное количество страниц памяти (размером 8 Кб), выделяемых для базы сообщений. При значении 0 количество устанавливается автоматически, исходя из доступного объема физической памяти. В текущей версии этот параметр не может быть изменен при перезапуске по сигналу SIGHUP Значение по умолчанию: **MaxPoolSize** = 0
SendTimeout = {время}	Максимальное время сканирования сообщения подключаемым модулем. В случае, если максимальное время сканирования превышено, считается, что при проверке сообщения произошла ошибка. Действия для такого случая определяются в параметре **ProcessingErrors** секции [Maild]. Значение по умолчанию: **SendTimeout** = 30s

FrozenTimeout = {время}	Дополнительное время на обработку письма. Если плагин не может обработать письмо за время, указанное в значении параметра **SendTimeout**, он может продлить время обработки на величину, заданную в параметре **FrozenTimeout**. Значение по умолчанию: **FrozenTimeout** = 2h
DeleteTimeout = {время}	Максимальное время хранения письма в базе сообщений. Значение параметра **DeleteTimeout** должно быть больше, чем значение параметра **FrozenTimeout**. Значение по умолчанию: **DeleteTimeout** = 48h
BackupPeriod = {время}	Промежуток времени, через который производится резервное копирование базы писем. При значении 0 резервное копирование не производится. Значение по умолчанию: **BackupPeriod** = 0
BackupName = {имя файла}	Имя файла резервной копии базы 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 базе писем. При превышении значения этого параметра письма сохраняются в отдельных внешних файлах. Значение по умолчанию: **MaxBodySizeInDB** = 1k
SyncMode = {Yes | No}	Режим синхронизации, используемый для внутренней БД. Если для данного параметра указано значение yes, то для каждой транзакции вызывается функция fsync. В результате, на диске гарантированно находится БД в актуальном состоянии после каждой транзакции. Однако при этом производительность уменьшается (иногда, значительно). Если указано значение no, то при обновлении БД используется буферизация ОС. В результате, при падении 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 лучше оставить данный параметр в значении no. Значение по умолчанию: **SyncMode** = no

Секция [Filters]

В секции [Filters] сосредоточены общие настройки работы плагинов **Dr.Web MailD**:

LibDir = {путь к директории}	Директория, в которой располагаются подключаемые модули программного комплекса. Значение по умолчанию: **LibDir** = %bin_dir/maild/plugins/
Settings = {список настроек плагинов}	Параметры запуска подключаемых модулей. Настройки модулей перечисляются через запятую в следующем формате: **Settings** = [настройки_плагина], [настройки_плагина]..., где [настройки_плагина] - это название_плагина: [параметр1]|...|[параметрN], а [параметрN] - это 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 ведения протокола в debug, а для подключаемого модуля drweb максимальный размер сообщения в 10 Мб. Значения параметров (за исключением путей к файлам) регистронезависимы. Значение по умолчанию: **Settings** =

В текущей версии для плагинов могут быть заданы следующие параметры:

section = {текст}	Название секции конфигурационного файла, в которой задаются настройки работы подключаемого модуля.
max_size = {размер}	Максимальный размер проверяемого сообщения для каждого подключаемого модуля. При значении 0 ограничения отсутствуют. Ограничение на размер по умолчанию зависит от того, в какой очереди запускается плагин, и определяется значением параметра **MaxSizeBeforeQueueFilters** или **MaxSizeAfterQueueFilters** текущей секции конфигурационного файла. Также параметр может быть задан в секции [Rules] в виде: имя_плагина/максимальный_размер = значение Для **Клиента** правила с **max_size** должны выглядеть следующим образом: [Rule: Client1] ... plugin_name/**max_size** = {размер} [Rules] **md-client:** Client1 cont rule=Client1

	Пример:
	`[Rule:Client1]`
	AdminMail = root@client1.drweb.ru
	SenderAddress = inet:25@10.0.0.0
	ProtectedDomains = client1.drweb.ru, client1
	ProtectedEmails = regex:.*@client1.drweb.ru, regex:.*@client1
	ProtectedNetworks = 10.0.0.0/32
	drweb/**max_size** = 100k
	`[Rules]`
	...
	md-client:Client1 cont **rule**=Client1

log_level = {Quiet | Error | Alert | Info | Debug}	Уровень детализации протокола работы подключаемого модуля. Значение параметра по умолчанию совпадает со значением параметра **Level** секции `[Logging]`.
log_ipc_level = {Quiet | Error | Alert | Info | Debug}	Уровень детализации протокола работы библиотеки IPC. Значение параметра по умолчанию совпадает со значением параметра **IpcLevel** секции `[Logging]`.

syslog_facility = {Daemon | Mail | Local0..Local7}	Тип подсистемы, через которую системный сервис syslogd, ведущий протоколирование работы плагина, выдает сообщения о событиях (более подробную информацию вы можете получить в документации по syslog). Значение параметра по умолчанию совпадает со значением параметра **SyslogFacility** секции [Logging].
log_filename = {syslog | путь к файлу}	Путь к файлу протокола или "syslog", если отчет о работе ведется с помощью системной службы syslogd.
path_to_lib = {путь к файлу}	Путь к библиотеке плагина. Путь может быть как абсолютным, так и относительным. Относительный путь задается от директории, указанной в параметре **LibDir** секции [Filters]. Значение параметра по умолчанию определяется по следующему принципу: к значению параметра **LibDir** добавляется строка LibDir+"/lib"+имя_плагина+".so", причем название плагина указывается в нижнем регистре. Например, для плагина vaderetro в версии **Dr.Web MailD** для Linux при установках по умолчанию путь будет имеет вид: %bin_dir/maild/plugins/libvaderetro.so. С помощью этого параметра также производится подключение и настройка плагинов сторонних производителей. Способ задания значения параметра в этом случае аналогичен вышеописанному: **Settings** = vaderetro: log_level=info| max_size=200k| path_to_lib = /opt/drweb/maild/plugins/libvaderetro.so.

	Значение по умолчанию: **path_to_lib** = LibDir+"/lib"+имя_плагина+".so"
BeforeQueueFilters = {список фильтров}	Список подключаемых модулей, обрабатывающих письмо до его помещения в очередь или базу сообщений.
	Значение по умолчанию: **BeforeQueueFilters** =
MaxSizeBeforeQueueFilters = {размер}	Максимальный размер письма для обработки плагинами, указанными в значении параметра **BeforeQueueFilters**. Используется только для тех подключаемых модулей, для которых значение параметра **max_size** не задано явным образом. При значении 0 ограничения отсутствуют.
	Значение по умолчанию: **MaxSizeBeforeQueueFilters** =
AfterQueueFilters = {список плагинов}	Список плагинов, обрабатывающих письмо после его помещения в очередь или базу сообщений.
	Значение по умолчанию: **AfterQueueFilters** =
MaxSizeAfterQueueFilters = {размер}	Максимальный размер письма для обработки плагинами, указанными в значении параметра **AfterQueueFilters**. Используется только для тех подключаемых модулей, для которых значение параметра **max_size** не задано явным образом. При значении 0 ограничения отсутствуют.

	Значение по умолчанию: **MaxSizeAfterQueueFilters** = 0
PluginsBaseDir = {путь к директории}	Путь к директории, где хранятся рабочие файлы плагинов. Значение по умолчанию: **PluginsBaseDir** = %var_dir/plugins/

Секция [Stat]

В секции [Stat] собраны параметры сбора статистики работы **Dr.Web MailD**:

Detail = {значение}	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. Также включается сбор статистики по всем Клиентам. • 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 в системе пользователю. Для каждого пользователя можно отдельно настраивать необходимость ведения статистики. Доступ к статистике можно получить как через управляющий сокет, так и через web-интерфейс. Статистика, собранная на уровне `low`, также передается в отчетах, если они включены. Для каждого из Клиентов можно настраивать статистику отдельно через параметр **`StatDetail`** в правилах. Значение по умолчанию: **`Detail`** = `low`
`Send` = { `Yes` | `No`}	Отсылка отчета серверу статистики (или серверу **Dr.Web Control Center**, если сервер **Dr.Web для почтовых серверов UNIX** работает в составе антивирусной сети). Значение по умолчанию: **`Send`** = `Yes`
`SendPeriod` = { время}	Промежуток времени, через который статистика отсылается на сервер. Значение по умолчанию: **`SendPeriod`** = `10m`
`Timeout` = { число}	Максимальное время ожидания ответа от сервера статистики. Значение по умолчанию: **`Timeout`** = `30s`

Существует возможность экспорта статистики средствами модуля **Dr.Web MailD** 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` = {`Yes` | `No`}	Возможность осуществлять экспорт статистики в хранилища, перечисленные в соответствующих параметрах (см. ниже). Параметр может быть задан для каждого Клиента отдельно. Данное значение используется для экспорта статистики по всем Клиентам. 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 плагина, 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. **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>)'"` Параметр может быть задан для каждого Клиента отдельно. Данное значение используется для экспорта статистики по всем Клиентам
	Значение по умолчанию: **ExportBlockObjectsStorage** =

ExportStatStorage = {текст запроса}	Возможность осуществлять экспорт статистики в хранилища, перечисленные в соответствующих параметрах (см. ниже). Параметр может быть задан для каждого Клиента отдельно. Данное значение используется для экспорта статистики по всем Клиентам. Экспорт статистики по общему числу обработанных сообщений. Сохранение данных запроса будет выполняться при: • завершении приложения; • через интервал времени, указанный в значении параметре **SendPeriod** секции настроек [Stat]. 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 отчеты; • : 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 отчеты; • : 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 плагина 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}	Отсылка отчета серверу статистики (или серверу **Dr.Web Control Center**, если сервер **Dr.Web для почтовых серверов UNIX** работает в составе антивирусной сети). Экспорт статистики по числу обработанных сообщений для каждого плагина. Статистика сохраняется только для плагинов, 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 плагина, 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: **Send** = Yes

Секция [Reports]

В секции [Reports] собраны параметры создания и отправки отчетов о работе подключаемых модулей программного комплекса.

Send = { Yes | No}	Отсылка отчетов. Параметр задается для каждого Клиента отдельно под именем **ReportsSend**. Данное значение используется по умолчанию для всех Клиентов. Значение по умолчанию: **Send** = Yes
SendTimes = {время}	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.

	Параметр задается для каждого Клиента отдельно под именем **ReportsSendTimes**. Данное значение используется по умолчанию для всех Клиентов. 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. Параметр задается для каждого Клиента отдельно под именем **ReportsMail**. Данное значение используется по умолчанию для всех Клиентов. Пожалуйста, обратите внимание, что если для **ReportsMail** задано значение, то на **AdminMail** отчеты приходить не будут. Значение по умолчанию: **Mail** =
Names = {список подключаемых модулей}	Список подключаемых модулей, для которых создается отчет. Формат списка: имя_плагина1, имя_плагина2, ... Если этот параметр не задан, отчет создается для модулей, перечисленных в значениях параметров **BeforeQueueFilter** и **AfterQueueFilter** секции настроек [Filters]. Параметр задается для каждого Клиента отдельно под именем **ReportsNames**. Данное значение используется по умолчанию для всех Клиентов. Значение по умолчанию:

	Names =
TopListSize = {число}	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. Параметр задается для каждого Клиента отдельно под именем **ReportsTopListSize**. Данное значение используется по умолчанию для всех Клиентов.
	Значение по умолчанию: **TopListSize** = 20
MaxStoreInDbPeriod = {время}	Максимальное время хранения статистики в базе отчетов. При значении 0 старые записи удаляться не будут.
	Значение по умолчанию: **MaxStoreInDbPeriod** = 31d
CheckForRemovePeriod = {время}	Промежуток времени, через который старые записи будут удаляться из базы отчетов.
	Значение по умолчанию: **CheckForRemovePeriod** = 5m

Конфигурационный файл **Dr.Web MailD** содержит ряд дополнительных параметров для создания и отправки отчетов о работе подключаемых модулей программного комплекса **Супер-Администратору**.

GeneralSend = { Yes | No}	Отсылка общих отчетов супер-администратору. В отправляемом отчете по умолчанию включена статистика по каждому из Клиентов, а также общая статистика по всем подключенным плагинам. Все настройки, относящиеся к общему отчету, имеют префикс **General**. Значение по умолчанию: **GeneralSend** = Yes

GeneralSendTimes = { время}	Настройки пула потоков.
	
	Первым определяется количество потоков в пуле:
	
	• График отправки общих отчетов Супер-администратору. 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, отчет будет составляться именно за заданный промежуток времени, если нет, то промежуток времени считается равным 24 часам.
	
	Значение по умолчанию:
	
	GeneralSendTimes = 00:00:00

GeneralClientFilter = {регулярное выражение}	Фильтрация по Клиентам при формировании отчета. Непустое значение параметра представляет собой регулярное выражение, которое сравнивается с идентификатором Клиента. При совпадении информация по Клиенту включается в отчет, иначе - пропускается. В любом случае в общей статистике будет содержаться информация обо всех Клиентах. Значение по умолчанию: **GeneralClientFilter** =
GeneralTotalStat = {Yes | No}	Включение в общий отчет статистики по каждому плагину. Значение по умолчанию: **GeneralTotalStat** = Yes
GeneralMail = {адрес электронной почты}	Адрес, на который высылаются отчеты для Супер-администратора. Значение по умолчанию: **GeneralMail** =
GeneralNames = {список подключаемых модулей}	Список плагинов, для которых создается отчет Супер-администратора. Формат списка: имя_плагина1, имя_плагина2, ... Если этот параметр не задан, отчет создается для модулей, перечисленных в значениях параметров **BeforeQueueFilter** и **AfterQueueFilter** секции настроек [Filters]. Значение по умолчанию: **GeneralNames** = drweb1, drweb2
GeneralTopListSize = {число}	Показ в отчете Супер-Администратора списков часто блокируемых объектов и адресов.

	Значение по умолчанию: **GeneralTopListSize** = 20

Секция [Quarantine]

В секции [Quarantine] собраны настройки работы **Карантина**.

Path = {путь к директории}	Путь к директории карантина. Значение по умолчанию: **Path** = %var_dir/infected/
FilesMode = {числовое значение	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 "_". Параметр может быть задан в Правилах под именем **QuarantineFilenamesPrefix**. Значение по умолчанию: **FilenamesPrefix** = maild
AccessByEmail = {Yes | No}	Запрос на получение писем, сохраненных в карантине, через отправление специальных управляющих писем. Специальное письмо направляется на адрес, указанный в значении параметра **FilterMail** (или в Правилах) со специальным заголовком Subject - 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 автоматически генерируется MUA при нажатии соответствующей ссылки в высылаемых отчетах. Обратите внимание, что поскольку значение параметра **OnlyTrustedControlMails** секции [Maild] по умочанию равно **yes**, управляющие письма должны отправляться из защищенной сети. 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 = {размер в килобайтах}	Общий максимальный размер 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 DBI хранилище, если оно подключено. Данное значение используется по умолчанию для всех Клиентов. В правилах можно задать размер сообщений в карантине и для каждого из Клиентов по-отдельности. Значение по умолчанию: **MaxSize** = 0
MaxNumber = {числовое значение}	Максимальное число сообщений в карантине. Если значение равно нулю, то число сообщений в карантине не ограниченно. Данный параметр влияет только на число сообщений во внутренней БД, и никак не влияет на DBI хранилище, если оно подключено.

	Данное значение используется по умолчанию для всех Клиентов. В правилах можно задать число сообщений в карантине и для каждого из Клиентов по-отдельности. Значение по умолчанию: **MaxNumber** = 0
MoveToDBI = {Yes | No}	Перемещение писем, сохраненных в карантине, из файлового хранилища в DBI-хранилище. Для перемещения в DBI-хранилище должны быть установлены модули Perl File::Temp и DBI. Значение по умолчанию: **MoveToDBI** = No
DBISettings = {текст}	Настройки подключения к DBI-хранилищу. "dbi:Pg:dbname=emails_db" База данных должна быть создана с использованием набора символов SQL-ASCII Значение по умолчанию: **DBISettings** =
DBIUsername = {текст}	Имя пользователя для подключения к DBI-хранилищу. Значение по умолчанию: **DBIUsername** =
DBIPassword = {текст}	Пароль для подключения к DBI-хранилищу. Значение по умолчанию: **DBIPassword** =

SQLInsertCommand = {текст}	Команда добавления письма в DBI-хранилище. Список и порядок элементов строго фиксирован: 1. Номер сообщения. 2. Относительный путь к файлу, из которого взят объект. Формат файла: client/plug-in/id.prefix, где client - идентификатор Клиента или def, если с письмом не связан Клиент, plug-in - имя плагина, 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 values (?,?,?,?,?,?)"
	Значение по умолчанию: **SQLInsertCommand** =

SQLRemoveCommand = {текст}	Команда удаления письма из DBI-хранилища. 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 DBI-хранилище. 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 и порядок возвращаемых элементов строго фиксирован: 1. Номер сообщения. 2. Время помещения сообщения в базу данных. 3. Тело сообщения. 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. 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 DBI-хранилище, и удаляются устаревшие сообщения. При значении 0 запуск программы из значения параметра **PathToDrwebQp** производиться не будет. Значение по умолчанию: **PulseTime** = 5m
PathToDrwebQp = {путь к файлу}	Путь к программе drweb-qp. Значение по умолчанию: **PathToDrwebQp** = %bin_dir/drweb-qp
MoveAll = {Yes | No}	Перемещение всей входящей почты сразу в директорию /Path_parameter_value/def/backup/ для архивирования. Данный параметр имеет смысл при **MoveToDBI** = Yes, иначе директория может быстро заполниться файлами с входящими письмами. Значение по умолчанию: **MoveAll** = No

Секция [Maild]

В секции [Maild] собраны общие параметры работы **Dr.Web**

MailID.

ProtectedNetworks = {lookups}	Список сетей, защищаемых **MailID**. Значения записываются с использованием бесклассовой адресации (CIDR). Этот параметр используется для определения доверенных сетей в соответствующих настройках плагина VadeRetro, и если в параметре SessionRestrictions в секции [Receiver] указано trust_protected_networks. **Пример**: **ProtectedNetworks** = 10.0.0.0/24, 127.0.0.0/8, 192.168.0.68 Значение по умолчанию: **ProtectedNetworks** = 127.0.0.0/8
ProtectedDomains = {lookups}	Задает список доменов, защищаемых **MailID**. Этот параметр используется для определения доверенных доменов, если в параметре SessionRestrictions в секции [Receiver] указано trust_protected_domains. **Пример**: **ProtectedDomains** = example.ru, example.com Значение по умолчанию: **ProtectedDomains** =
IncludeSubdomains = {Yes | No}	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}	Адрес, куда отсылаются сообщения при использовании действия Redirect. Значение по умолчанию: **RedirectMail** = root@localhost
OnlyTrustedControlMails = {Yes | No}	Возможность отправлять управляющие письма только из защищаемой сети. Если компонент **Receiver** не передал информацию об IP-адресе клиента, то необходимо с помощью параметра **GetIpFromReceivedHeader** заставить используемую 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 = {численное значение}	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 = {действия}	Действия, выполняемые, если счет письма превысит значение параметра **MaxScore**. Если указано действие reject и значение параметра **UseCustomReply** установлено в yes, то SMTP-ответ берется из параметра **ReplyMaxScore**. После выполнения всех действий проверка сообщения завершается. К основным действиям (которые должны быть заданы в обязательном порядке) относятся: pass, discard, reject, tempfail. Также дополнительно могут быть заданы следующие действия: quarantine, redirect, add-header, score. Обратите внимание, что при настройке данного параметра может быть одновременно задано несколько значений. Значение по умолчанию: **MaxScoreAction** = reject

MaxMimeParts = {численное 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**. Значение по умолчанию: **MaxMimeParts** = 1000
MaxNestedMimeParts = {численное 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**. Значение по умолчанию: **MaxNestedMimeParts** = 100
LicenseLimit = {действия}	Действия над сообщениями, которые не были проверены из-за лицензионных ограничений. К основным действиям (которые должны быть заданы в обязательном порядке) относятся: pass, discard, reject, tempfail. Также дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Обратите внимание, что при настройке данного параметра может быть одновременно задано несколько значений. Значение по умолчанию: **LicenseLimit** = pass

EmptyFrom = {действия}	Реакция на пустое поле From в заголовках письма. Подобная ситуация возможна при использовании почтовых уведомлений; также это поле не заполняют спамеры. К основным действиям (которые должны быть заданы в обязательном порядке) относятся: continue, discard и reject. Также дополнительно могут быть заданы следующие действия: quarantine, redirect, add-header, score. Обратите внимание, что при настройке данного параметра может быть одновременно задано несколько значений.
	Значение по умолчанию: **EmptyFrom** = continue
ProcessingErrors = {действия}	Действие, применяемое к сообщениям, вызвавшим ошибки сканирования. К основным действиям (которые должны быть заданы в обязательном порядке) относятся: pass, discard, reject, tempfail. Также дополнительно могут быть заданы следующие действия: quarantine, redirect, notify, add-header, score. Обратите внимание, что при настройке данного параметра может быть одновременно задано несколько значений.
	Значение по умолчанию: **ProcessingErrors** = pass
PidFile = {путь к файлу}	Путь к PID-файлу процесса drweb-maild.
	Значение по умолчанию: **PidFile** = %var_dir/run/drweb-maild.pid

RulesLogLevel = {Quiet | Error | Alert | Info | Debug}	Определяет уровень подробности протокола работы обработчика Правил. Значение по умолчанию: **RulesLogLevel** = Alert

В тех случаях, когда сообщение блокируется программным комплексом, его SMTP-ответ состоит из кода ошибки 550 5.7.0 и текстового сообщения, содержание которого может задаваться параметрами, описанными ниже. Значения параметров должны быть заключены в кавычки.

UseCustomReply = {Yes | No}	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}	Ответ, отправляемый при срабатывании действия **EmptyFrom** = reject, если **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}	Ответ, отправляемый при срабатывании действия **ProcessingError** = reject, если **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}	Ответ, отправляемый при срабатывании действия **MaxScoreAction** = reject, если **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 = {Yes | No}	Использование в качестве IP-адреса Клиента значение из заголовка Received в случае, если IP-адрес не определяется компонентом **Receiver**. Значение по умолчанию: **GetIpFromReceivedHeader** = Yes
Control = {Yes | No}	Включение интерактивного модуля управления drweb-maild. Значение по умолчанию: **Control** = No

ControlAddress = {адрес}	Адреса, используемые модулем интерактивного управления drweb-maild. Значение по умолчанию: **ControlAddress** = inet:3009@127.0.0.1
ControlPoolOption = {настройки пула}	Настройки пула потоков для управляющего сокета drweb-maild. Значение по умолчанию: **ControlPoolOption** = auto
SkipDSNOnBlock = {Yes | No}	Пропуск отправления DSN, если при выполнении действий Reject или Tempfail код возврата невозможно вернуть компоненту **Receiver**. Значение по умолчанию: **SkipDSNOnBlock** = No

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, Postfix и в версии, предназначенной для работы в режиме прокси-сервера по протоколу SMTP:

Address = {адрес}	Адрес, используемый компонентом **Receiver** для получения сообщений. В параметре **Address** задается сокет, через который получаются сообщения (либо TCP-сокет в формате inet:порт@имя_хоста, либо UNIX сокет в формате local:путь_к_файлу_сокета).

	Значение по умолчанию: **Address** = inet:25@0.0.0.0
PoolOptions = {настройки пула}	Настройки пула потоков. Первым определяется количество потоков в пуле: • auto – количество потоков определяется автоматически в зависимости от загрузки системы; • N – целое неотрицательное число. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности; • N-M – целые положительное значения, и M ≥ N. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности, пока число потоков не достигнет значения M. 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} – статистика по потокам в пуле. Статистика сохраняется при получении системного сигнала SIGUSR1 в директории, определенной значением параметра **BaseDir** секции [General]. Значение по умолчанию: no

	• **log_level** = {Quiet| Error| Alert| Info| Debug} – уровень подробности файла протокола для потоков в пуле. Если значение не задано, используется значение параметра **LogLevel** секции [Logging]. • **stop_timeout** = {время} – тайм-аут на остановку работающего потока (например при завершении работы программы или когда требуется уменьшить число потоков в пуле). Значение по умолчанию: **PoolOptions** = auto
RealClients = {Yes | No}	Возможность приема соединений напрямую от клиентов. Значение по умолчанию: **RealClients** = Yes
ProcessingErrors = {действия}	Действия, совершаемые 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 "застрявших" писем. "Застрявшие" письма - сообщения, полученные, но не обработанные плагинами вовремя, чтобы быть отправленными компоненту **Checker**. Такая ситуация может случиться при возникновении проблем с сетью или питанием

	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 = {Yes | No}	Добавление заголовка Received ко всем получаемым сообщениям. Значение по умолчанию: **AddReceivedHeader** = Yes
ReturnReject = {Yes | No}	Параметр определяет поведение компонента **Receiver** в случае выполнения действия Reject. При значении Yes возвращается ошибка 5**, при значении No возвращается ошибка 2** и отправителю сообщения высылается DSN-отчет. При работе с почтовой системой Exim и наличии плагинов в списке **BeforeQueueFilters** рекомендуется использовать **ReturnReject** = No, чтобы избежать задержки сообщений на обработку в очереди Exim. Значение по умолчанию: **ReturnReject** = Yes

GreetingString = {текст}	Строка, выводимая в качестве приветствия при подключении нового Клиента. Макрос %host% заменяется на значение параметра **Hostname** из секции настроек [General], макрос %ver% заменяется на текущую версию модуля drweb-receiver. Значение по умолчанию: **GreetingString** = "%host% Dr.Web SMTP receiver v%ver% ready"
MaxRecipients = {численное значение}	Максимальное количество получателей. 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 = {численное значение}	Максимальное количество SMTP-подключений с одного IP-адреса. При значении 0 ограничений нет. Значение по умолчанию: **MaxConcurrentConnection** = 5
MaxMailsPerSession = {численное значение}	Максимальное количество сообщений за одну сессию. При значении 0 ограничений нет. Значение по умолчанию: **MaxMailsPerSession** = 20

MaxReceivedHeaders = {численное 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 = {численное значение}	Максимальное количество ошибок за одну сессию. При значении 0 ограничений нет. Значение по умолчанию: **MaxErrorsPerSession** = 10
MaxMsgSize = {размер}	Максимальный размер сообщения. 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 = {численное значение}	Максимальное число RSET, NOOP и NTFY команд 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 = {численное значение}	Максимальное число HELO, EHLO и LHLO команд на сессию. Если число команд превысит указанное значение, то начнет увеличиваться счетчик ошибок. Если значение равно 0, то данное ограничение не учитывается. Значение по умолчанию: **MaxHELOCommands** = 20
RelayDomains = {lookups}	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 пересылатся не будет. Для задания поддоменов возможно использование регулярного выражения или задание файла со списком регулярных выражений rfile. **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 для почтовых серверов UNIX** параметр **RelayDomains** не поддерживает формат записей wildcard DNS. Таким образом, подобная запись некорректна: **RelayDomains** = `*.domain` Значение по умолчанию: **RelayDomains** =

Следующие несколько параметров определяют проверки, которым подвергаются IP-адреса соединений, не отмеченные как надежные, на различных этапах SMTP-сессии.

По умолчанию надежными считаются соединения с `localhost` и от UNIX-сокетов.

Действия по проверке IP-адреса соединения записываются в значение соответствующего параметра последовательно, через запятую. Очередность выполнения действий соответствует очередности их записи.

SessionRestrictions = {текст}	Параметр задает проверки, осуществляемые непосредственно в начале соединения. Специфические для этого этапа действия: • `trust_protected_network [SCORE]` – если IP-адрес соединения находится в списке, определенном значением параметра **ProtectedNetworks**, адрес помечается как надежный, либо, если указан `SCORE`, значение `SCORE` прибавляется к счету каждого письма, передаваемого в этой сессии, а также данный счет прибавляется к счету IP-адреса отправителя;

	• `trust_protected_domains [SCORE]` – проверка IP-адреса соединения на вхождение в список, определяемый параметром **`ProtectedDomains`**. Проверка осуществляется через двойной DNS-запрос. 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. Если совпадение найдено, то IP-адрес соединения помечается как надежный, либо, если указан `SCORE`, значение `SCORE` прибавляется к счету каждого письма, передаваемого в этой сессии, а также данный счет прибавляется к счету IP-адреса отправителя; • `trust_white_networks [SCORE]` – если IP-адрес соединения находится в белом списке, определенном значением параметра **`WhiteNetworks`**, адрес помечается как доверенный IP-адрес, либо, если указан `SCORE`, значение `SCORE` прибавляется к счету каждого письма, передаваемого в этой сессии, а также данный счет прибавляется к счету IP-адреса отправителя;

	• `trust_white_domains [SCORE]` – проверка, находится ли IP-адрес соединения в белом списке, определенном значением параметра **`WhiteDomains`**. Для этого производится PTR-запрос. При совпадении адрес помечается как доверенный IP-адрес, либо, если указан `SCORE`, значение `SCORE` прибавляется к счету каждого письма, передаваемого в этой сессии, а также данный счет прибавляется к счету IP-адреса отправителя; • `reject_dnsbl [SCORE]` – проверка, находится ли IP-адрес соединения в черных списках RBL/DNSBL, определенных значением параметра **`DNSBLList`**. Для этого производится PTR-запрос. При совпадении сессия закрывается, 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; • `reject_black_networks [SCORE]` – если IP-адрес соединения находится в черном списке, определенном значением параметра **`BlackNetworks`**, сессия закрывается, 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;

	• `reject_black_domains [SCORE]` – проверка, находится ли IP-адрес соединения в черном списке, определенном значением параметра **`BlackDomains`**. Для этого производится PTR-запрос. При совпадении сессия закрывается, 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. Значение по умолчанию: **`SessionRestrictions`** = `trust_protected_network, trust_sasl_authenticated`
`HeloRestrictions` = `{текст}`	Проверки, выполняемые на стадии `HELO/EHLO`. Специфические для этого этапа действия: • `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. Значение по умолчанию: **`HeloRestrictions`** =
`SenderRestrictions` = {текст}	Проверки, выполняемые на стадии `FROM`. • `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; • `trust_sasl_authenticated [SCORE]` – если SASL-аутентификация была успешной, адрес помечается как доверенный IP-адрес, либо, если указан `SCORE`, значение `SCORE` прибавляется к счету передаваемого письма. Значение по умолчанию:

	SenderRestrictions = trust_sasl_authenticated
RecipientRestrictions = {текст}	Проверки, выполняемые на стадии RCPT. Все заявленные получатели проверяются по очереди. • reject_unknown_domain [SCORE] – если имя хоста отправителя 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; • reject_unauth_destination [SCORE] – если домена получателя нет ни в списке **RelayDomains**, ни в списке **ProtectedDomains**, почта на такой адрес блокируется, либо, если указан SCORE, в лог выводится ошибка, а значение SCORE прибавляется к счету передаваемого письма; • reject_unknown_rcpts [SCORE] – проверяет получателя на присутствие в списке **ProtectedEmails**. Если адреса в этом списке нет, почта на такой адрес 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.
	Значение по умолчанию:

	RecipientRestrictions = reject_unauth_destination
DataRestrictions = {текст}	Проверки, выполняемые на стадии DATA. • 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), и имя пользователя находится в списке, определенном значением параметра **SpamTrap**, 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.
	Значение по умолчанию: **DataRestrictions** =

RestrictionStat = {Yes | No}	Сбор статистики по работе с ограничениями. Получить статистику можно, послав сигнал SIGUSER1 процессу drweb-receiver. Статистика хранится в файле restrictions.txt в директории, указанной в значении параметра **BaseDir** секции настроек [General]. Значение по умолчанию: **RestrictionStat** = No
DelayRejectToRcpt = {Yes | No}	Приостановка блокирования почты до стадии RCPT. Установка данного параметра позволяет работать с устаревшими версиями почтовых клиентов и выводить список заблокированных адресов получателей в лог. Значение по умолчанию: **DelayRejectToRcpt** = Yes
BlackNetworks = {lookups} **WhiteNetworks** = {lookups}	Черные и белые списки сетей, используемые в действиях trust_white_networks и reject_black_networks. Синтаксис данного параметра аналогичен синтаксису параметра **ProtectedNetworks**. Значение по умолчанию: **BlackNetworks** = **WhiteNetworks** =

DNSBLList = {lookups}	Список серверов DNSBL. Данный список используется в действии reject_dnsbl. Сервера проверяются по очереди в том порядке, в котором они перечислены в значении параметра, до момента, пока сообщение не будет заблокировано сервером или лист серверов не закончится. 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 = {lookups} **WhiteDomains** = {lookups}	Черные и белые списки доменов, используемые в действиях trust_white_domains и reject_black_domains. Синтаксис аналогичен синтаксису параметра **ProtectedDomains**. Значение по умолчанию: **BlackDomains** = **WhiteDomains** =
SpamTrap = {lookups	Список адресов для ловушки для спама. Данный список используется в действии reject_spam_trap. Значение по умолчанию: **SpamTrap** =
ReputationIPFilter = {список фильтров}	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. 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** =
ProtectedEmails = {lookups}	Список защищаемых адресов. Используется в ограничении **reject_unknown_rcpts**.

	Позволяет отбрасывать неверных получателей и, при использовании фильтра `anti_dha` в Reputation IP Filter, эффективно бороться с DHA-атаками. Соответственно, рекомендуется задавать этот параметр вместе с **`reject_unknown_rcpts`** и использовать совместно с `anti_dha` фильтром.
	Значение по умолчанию: **`ProtectedEmails`** =
`MaxSessionScore` = {численное 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`

Ограничения (Restrictions) 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).

Ограничения работают на следующих этапах SMTP-сессии:

- на этапе подключения нового клиента (параметр **`SessionRestrictions`**);
- при получении команды `HELO/EHLO` (параметр **`HeloRestrictions`**);
- при получении команды `FROM` – т.е. когда клиент устанавливает отправителя для нового письма (параметр **`SenderRestrictions`**);
- при получении команды `RCPT` – т.е. когда клиент

добавляет нового получателя к текущему письму (параметр **RecipientRestrictions**);

- при получении команды DATA – т.е. когда клиент закончил передавать всех получателей письма и готов отправлять тело письма (параметр **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.

С каждым соединением связан флаг Trusted IP. Если он установлен, то проверка на ограничения не производится.

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**.

Эффект от блокировки по ограничению различен в зависимости от этапа SMTP-сессии, на которой происходит проверка. В **SessionRestrictions** 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 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 ограничения, а также имеются ограничения, которые можно использовать на всех (или почти на всех) этапах. К последним относятся:

- `mark_trust [SCORE]` – установить флаг `Trusted IP`. Если есть еще ограничения после даного, то они будут пропущены (т.е. они бесполезны). Если указан счет, то флаг `Trusted IP` устанавливается, только если текущий счет меньше указанного в параметре.
- `sleep SEC [SCORE]` – заснуть на число секунд, указанных в `SEC`. Полезно для блокировки некоторых спамеров, так как большинство из них не собираются ждать даже несколько секунд в ожидании ответа от сервера. Если указан счет, то действие выполняется, только если текущий счет больше указанного в параметре.
- `tempfail [SCORE]` – вернуть временную SMTP-ошибку (код `4*`). Полезно, когда надо временно отклонить клиента, если тот не прошел какие-либо проверки, которые в дальнейшем все-таки может пройти. Если указан счет, то временная ошибка возвращается, только если текущий счет больше указанного в параметре.
- `reject [SCORE]` – вернуть постоянную SMTP-ошибку (код `5*`). Полезно, если клиент не прошел проверку, которую и в дальнейшем он тоже не сможет пройти. Если указан счет, то ошибка возвращается, только если текущий счет больше указанного в параметре.

- pass_sasl_authenticated [SCORE] – пропустить все остальные проверки на данном этапе SMTP-сесии, если клиент успешно прошел SASL аутентификацию. Данную проверку можно использовать на всех этапах, но полезно только для **SenderRestrictions**, **RecipientRestrictions** и **DataRestrictions**, так как авторизацию можно пройти только после получения команды HELO/EHLO. Если указан SCORE, то пропуск проверок выполняется, только если текущий счет меньше указанного в параметре. Обратите внимание, что пропускаются только те проверки, которые указаны для данного этапа SMTP-сессии (т.е. после pass_sasl_authenticated). Проверки на других этапах пропущены не будут.
- set_score SCORE – заменяет текущий счет письма 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.

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.

Секция [SASL]

В секции `[SASL]` собраны параметры аутентификации SASL в версии **Dr.Web для почтовых серверов UNIX**, предназначенной для работы в режиме прокси-сервера по протоколу SMTP:

Use = {Yes | No}	Подключение возможности использования аутентификации SASL. Значение по умолчанию: **Use** = No
Driver = {cyrus}	Имя драйвера аутентификации SASL. В текущей версии доступен только драйвер `cyrus`. Для его использования необходимо поставить и настроить библиотеку `cyrus-sasl2`. Значение по умолчанию: **Driver** = cyrus
BrokenAuthClients = {Yes | No}	Возможность поддержки устаревших SMTP-клиентов, использующих нестандартный синтаксис протокола `AUTH`. Значение по умолчанию: **BrokenAuthClients** = Yes
AuthenticatedHeader = {Yes | No}	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 cyrus-sasl драйвера.

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. Значение по умолчанию: **Path** = maild
ServerHostname = {текст}	Имя хоста. 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. Значение по умолчанию: **ServerHostname** =
ServerRealm = {текст}	Область SASL (SASL realm), в которой находится сервер 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

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 = {Yes | No}	Добавление заголовка **SecureHash** ко всем исходящим сообщениям. Параметры **UseSecureHash** и **SecureHash** не имеют смысла при работе программного комплекса с почтовыми системами Courier и Exim, а также при работе в режиме SMTP-proxy. При работе **Dr.Web для почтовых серверов UNIX** с почтовой системой Sendmail либо с почтовой системой Qmail значение Yes должно быть указано для данного параметра, если для получения и отправки сообщений используется одна и та же почтовая система. Это позволяет исключить возможность зацикливания сообщений и оптимизировать работу системы. В случае, если получение и отправка почты осуществляются через различные почтовые системы, следует указать No, чтобы избежать увеличения заголовка **SecureHash** за пределы системных ограничений. Значение по умолчанию: **UseSecureHash** = Yes При работе **Dr.Web для почтовых серверов UNIX** с почтовой системой **Zmailer** значение Yes должно быть указано для данного параметра, 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. Заголовок **SecureHash** добавляется, чтобы исключить возможность зацикливания и двойной проверки сообщений. Значение по умолчанию: **UseSecureHash** = No

	При работе **Dr.Web MailD** с почтовой системой Postfix значение `Yes` должно быть указано для данного параметра, только если взаимодействие с почтовой системой Postfix производится по протоколу `milter` (используется модуль `drweb-milter`). В этом случае все сообщения, сформированные модулем `drweb-sender`, обрабатываются модулем `drweb-milter`. Заголовок **SecureHash** добавляется, чтобы исключить возможность зацикливания и двойной проверки сообщений. Значение по умолчанию: **UseSecureHash** = No
SecureHash = {текст}	Параметр задает содержимое заголовка **SecureHash**. Значением параметра может быть произвольная строка, рекомендуемая длина строки - не менее 10 символов. Для повышения безопасности настоятельно рекомендуется изменить значение по умолчанию данного параметра. При работе **Dr.Web для почтовых серверов UNIX** 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}	Промежуток времени для обработки "застрявших" писем. "Застрявшие" письма - сообщения, полученные, но не обработанные подключаемыми модулями вовремя, чтобы быть отправленными компоненту **Checker**. Такая ситуация может случиться при возникновении проблем с сетью или питанием. Значение по умолчанию: **StalledProcessingInterval** = 10m
SendingIntervals = {время}	Промежутки времени между попытками отправить "застрявшие" письма. 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 ноль, 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. 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 SMTP-протоколу;

	• LMTP – сообщения отправляются по LMTP-протоколу; • 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. В текущей версии этот параметр не может быть изменен при перезапуске по сигналу SIGHUP. Значение по умолчанию: зависит от дистрибутива.
Address = {адрес}	Адрес, используемый компонентом **Sender** для отправки сообщения. 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-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. Можно указать несколько адресов для отправки сообщений. Значения разделяются запятой (",").

	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, находящаяся 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** до 5 минут, чтобы **Sender** успел переключиться к последнему адресу в случае отсутствия ответа от предыдущих адресов.
	Значение по умолчанию: зависит от дистрибутива.
Options = {строка}	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.
	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-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 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 SMTP/LMTP-протоколу. Значение по умолчанию: **OtherCmdsTimeout** = 2m
PipeTimeout = {время}	Максимальный промежуток времени на получение ответа при использовании pipe. Значение по умолчанию: **PipeTimeout** = 2m
SendDSN = {Yes | No}	Отправка DSN-отчета. Значение по умолчанию: **SendDSN** = No
Router = {строки и файлы}	Правила маршрутизации сообщений в зависимости от их получателей при работе программного комплекса в режиме SMTP-прокси. Сообщения, адресованные разным получателям, могут быть отправлены с разных адресов. В случае, если сообщение имеют несколько получателей и такое сообщение должно быть отправлено с разных адресов, список получателей должен быть разделен на группы таким образом, что на адреса каждой группы сообщение будет отправлено с отдельного адреса. Копия сообщения создается для каждой группы получателей.

	Значения параметра задаются в формате `DOMAIN ADDRESS`, где: • `DOMAIN` – строка, на вхождение которой проверяются конверты получателей. Конверт имеет вид `<user@host>`. Поиск регистронезависимый. Например, при поиске строки `@localhost` она будет обнаружена в конвертах `<test@localhost>` и `<yy@localhost.localdomain>`, а при поиске строки `@localhost>` она будет найдена только только в конверте `<test@localhost>`. • `ADDRESS` – адреса, на которые будут отправляться сообщения, если если строка `DOMAIN` будет найдена в конверте. Формат `ADDRESS` аналогичен формату параметра **Address** данного конфигурационного файла. Возможно указание нескольких адресов с разделением их символом "| ", тогда письмо будет доставлено по первому адресу, с которым удалось установить соединение.
	Пример: **Router** = `@main.server.com> mx:main.server.com|inet:25@backup.server.com` В этом случае письма, получатели которых имеют домен `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`.

	Значение по умолчанию: **Router** =

Использование Router

Параметр **Router** позволяет использовать Lookup (за исключением `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 вначале будет производиться поиск доменного имени отправителя в атрибуте `cn`, в случае обнаружения параметры перенаправления берутся из поля `description`. 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 1025 порт, а затем на порт 2025. Письма, отправленные адресованные домену `mail.com` будут пересылаться на адреса, соответствующие MX-записи `mail.com`. Письма, адресованные домену `domain2.com` будут перенаправляться на адреса, указанные для MX-записи `mail.ru`, либо на порт 25 сервера `mail.backup`. 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
```

В случае если письмо не удалось отослать ни на один из найденных адресов, то в зависимости от кода ответа, который вернет последний MTA:

- **Sender** перейдет к отложенной отправке спустя интервал, указанный в значении параметра **SendingIntervals**. При отложенной отправке также действуют правила, указанные в параметре **Router**. Если отложенная отправка закончится неудачей, то в зависимости от настроек параметра **SendDSN**, **Notifier** сгенерирует DSN. Если для домена отправителя, указанного в конверте недоставленного письма, не заданы специальные маршруты в параметре **Router** или в правилах **maild**, то данный DSN будет отправлен на адрес, указанный в значении параметра **Address**.
- если последний MTA ответит кодом 5**, DSN будет сгенерирована сразу, а письмо удалено из out-очередей. Схема отправки DSN аналогична вышеописанному. Если DSN не может быть доставлен, то спустя интервал, указанный в значении параметра **SendingIntervals**, он будет удален.

При определении маршрутов для одного и того же домена в секции `[Rules]` и в параметре **Router**, будут действовать только маршруты, определенные в секции `[Rules]`.

Обратите внимание, что даже при настройке маршрутизации всей почты с помощью **Router**, значение параметра **Address** не должно быть пустым. В противном случае при запуске **Sender** выведет ошибку и завершит свою работу. При этом необходимо учитывать, что если в таблице маршрутизации для получателя не будет найдено соответствия, то письмо будет

отправлено на адрес, указанный в значении параметра **Address**.

Секция [Milter]

Параметры секции [Milter] управляют работой модуля drweb-milter, отвечающего за взаимодействие программного комплекса с почтовыми системами Postfix и 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 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 почтовой системы. Значение по умолчанию: **Timeout** = 2h

PendedConnections = {численное значение}	Максимальная длина очереди на соединение с почтовой системой (drweb-milter ожидает окончания обработки сообщений от почтовой системы). Значение по умолчанию: **PendedConnections** = 64
CanChangeBody = {Yes | No}	Возможность изменения тела сообщения почтовой системой. Почтовая система Postfix данную функцию поддерживает начиная с версии 2.4. Этот параметр не может быть изменен при перезапуске по сигналу SIGHUP. Значение по умолчанию: **CanChangeBody** = Yes
ProcessingTimeout = {время}	Максимальное время ожидания модулем drweb-milter 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
MinPersistConnection = {численное	Минимальное количество соединений с модулем drweb-maild.

значение}	Значение по умолчанию: **MinPersistConnection** = 2
UseStat = {Yes | No}	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 [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. Первым определяется количество потоков в пуле: • auto – количество потоков определяется автоматически в зависимости от загрузки системы; • N – целое неотрицательное число. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности, • N-M – целые положительное значения, и M ≥ N. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности, пока число потоков не достигнет значения M. 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} – статистика по потокам в пуле. Статистика сохраняется при получении системного сигнала SIGUSR1 в директории, определенной значением параметра **BaseDir** секции [General]. Значение по умолчанию: no • **log_level** = {Quiet| Error| Alert| Info| Debug} – уровень подробности файла протокола для потоков в пуле. Если значение не задано, используется значение параметра **LogLevel** секции [Logging]. • **stop_timeout** = {время} – тайм-аут на остановку работающего потока (например при завершении работы программы или когда требуется уменьшить число потоков в пуле).
	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 = {Yes | No}	Добавление заголовка **SecureHash** ко всем исходящим сообщениям. Если указано значение No, модуль drweb-cgp-receiver не будет проверять сообщения, отправленные по программному каналу (pipe). Если указано значение Yes, модуль drweb-cgp-receiver будет пропускать сообщения с заголовком **SecureHash** без проверки. В случае, если получение и отправка почты осуществляются через различные почтовые системы, следует указать No, чтобы избежать увеличения заголовка **SecureHash** за пределы системных ограничений.

	Значение по умолчанию: **UseSecureHash** = No
SecureHash = {строка}	Параметр задает содержимое заголовка **SecureHash**. Значением параметра может быть произвольная строка, рекомендуемая длина строки - не менее 10 символов. Для повышения безопасности настоятельно рекомендуется изменить значение по умолчанию данного параметра.
	Значение по умолчанию: **SecureHash** = !!!--------- ___EDIT_THIS___!!!
PoolOptions = {настройки пула}	Настройки пула потоков. Первым определяется количество потоков в пуле: • auto – количество потоков определяется автоматически в зависимости от загрузки системы; • N – целое неотрицательное число. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности, • N-M – целые положительное значения, и M ≥ N. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности, пока число потоков не достигнет значения M. 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} – статистика по потокам в пуле. Статистика сохраняется при получении системного сигнала SIGUSR1 в директории, определенной значением параметра **BaseDir** секции [General]. Значение по умолчанию: no • **log_level** = {Quiet| Error| Alert| Info| Debug} – уровень подробности файла протокола для потоков в пуле. Если значение не задано, используется значение параметра **LogLevel** секции [Logging]. • **stop_timeout** = {время} – тайм-аут на остановку работающего потока (например при завершении работы программы или когда требуется уменьшить число потоков в пуле). Значение по умолчанию: **PoolOptions** = auto
SubmitDir = {путь к директории}	Директория, в которую 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 создаваемые уведомления или излеченные 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 протоколов. Значение по умолчанию: **SubmitFilenamesPrefix** = drweb_submit_%s_
SubmitFilenamesMode = {std | tai | rand48}	Настройки пула потоков. 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 [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 директории}	Директория установки почтовой системы Courier. Значение по умолчанию: **BaseDir** = /usr/lib/courier
SocketDirs = {строки и файлы}	Список путей, используемых для создания UNIX-сокетов при взаимодействии с почтовой системой Courier. Сокет создается в первой директории списка, а остальные директории проверяются на наличие UNIX-сокетов, чьи имена совпадают с названием модуля drweb-courier. При нахождении такие UNIX-сокеты удаляются. В текущей версии этот параметр не может быть изменен при перезапуске по сигналу SIGHUP. 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 этот параметр не может быть изменен при перезапуске по сигналу SIGHUP. Значение по умолчанию: **SocketAccess** = 0660

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 [Notifier]

В секции [Notifier] содержатся настройки модуля drweb-notifier, отвечающего за создание и отправку пользователю отчетов о действиях компонентов программного комплекса **Dr.Web для почтовых серверов UNIX**.

PoolOptions = {настройки пула}	Настройки пула потоков. Первым определяется количество потоков в пуле:

	• auto – количество потоков определяется автоматически в зависимости от загрузки системы; • N – целое неотрицательное число. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности, • N-M – целые положительное значения, и M ≥ N. Как минимум N потоков в пуле будут активны, а новые потоки будут создаваться по мере надобности, пока число потоков не достигнет значения M. 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} – статистика по потокам в пуле. Статистика сохраняется при получении системного сигнала SIGUSR1 в директории, определенной значением параметра **BaseDir** секции [General]. Значение по умолчанию: no • **log_level** = {Quiet| Error| Alert| Info| Debug} – уровень подробности файла протокола для потоков в пуле. Если значение не задано, используется значение параметра **LogLevel** секции [Logging].

	• **stop_timeout** = {время} – тайм-аут на остановку работающего потока (например при завершении работы программы или когда требуется уменьшить число потоков в пуле). Значение по умолчанию: **PoolOptions** = auto
TemplatesBaseDir = {путь к директории}	Путь к директории, где хранятся шаблоны отчетов. Значение по умолчанию: **TemplatesBaseDir** = %etc_dir/maild/templates
LngBaseDir = {путь к директории}	Путь к директории, где хранятся языковые файлы для формирования отчетов. Языковые файлы имеют расширение .lng. Используемый язык (ru для русского языка, en для английского языка и т.д.) указан в первой незакомментированной строчке языкового файла. Заданное значение используется в параметре **NotifyLangs**, чтобы определить язык формирования отчетов. Значение по умолчанию: **LngBaseDir** = %etc_dir/maild/lng
AdminMail = {адрес электронной почты}	Адрес системного администратора. Можно указать несколько адресов, тогда все сформированные отчеты будут отправляться на все заданные адреса, и при этом в теле письма с отчетом будут отображаться все указанные адреса. Рекомендуется определить этот параметр, иначе не будут отправляться отчеты

	Значение по умолчанию: **AdminMail** = root@localhost
FilterMail = {адрес электронной почты}	Адрес, указываемый в заголовке From писем с отчетами. Значение по умолчанию: **FilterMail** = root@localhost
NotifyLangs = {список названий языков}	Языки, используемые при формировании отчетов. Значение по умолчанию: **NotifyLangs** = en
TemplatesParserLogLevel = {quiet | error | alert | info | debug}	Уровень подробности протокола работы подсистемы, формирующей отчеты на основе шаблонов. Значение по умолчанию: **TemplatesParserLogLevel** = info
RulesLogLevel = {quiet | error | alert | info | debug}	Уровень подробности протокола работы обработчика Правил. Значение по умолчанию: **RulesLogLevel** = Alert
MsgIdMap = {строка}	Отображение идентификатора сообщения, заданного в компоненте **Receiver**, в идентификатор компонента **Sender**, которому будут отправляться отчеты, сформированные для данного сообщения. Если такое отображение не найдено, все отчеты будут отправляться компоненту **Sender** по умолчанию (с пустым идентификатором). **Пример**: **MsgIdMap** = id[12]

	`sender_notifications` В этом случае отчеты 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 отчете для пути файла в карантине будет выводиться, например, `http://mailhost/quarantine/headersfilter/drweb.quarantine.2kqtvI` Значение по умолчанию: **`QuarantinePrefix`** =

При обработке письма **Dr.Web MailD** любой из подключенных к нему плагинов может запросить отправление уведомления о каком-либо событии (обнаружении вируса, ошибке обработки, блокировке письма по определенному критерию и т.п.). Данные отчеты формируются компонентом **Dr.Web Notifier** (модуль `drweb-notifier`), который генерирует письмо и затем отправляет его с использованием компонента **Sender**.

Все отчеты представлены в виде шаблонных файлов с расширением `.msg` (**Dr.Web Notifier** ищет их в директории, указанной в значении параметра **`TemplatesBaseDir`**). Данные шаблоны поддерживают макросы, условия, циклы и включение внешних файлов (синтаксис данных файлов описан в файлах `notify.*`), и их можно свободно изменять под свои нужды.

Существует три типа отчетов:

- отчеты, которые отправляются на конкретное письмо;
- отчеты, которые отправляются периодически и содержат информацию об общей работе комплекса;
- DSN-отчеты о невозможности доставки письма.

Во всех случаях компонент отправляет модулю `drweb-notifier` имя отчета, который требуется отправить. Все шаблоны кроме DSN-шаблонов по умолчанию поддерживают отчеты двух видов - в html-виде и в простом текстовом виде. Выбор типа отчета происходит на основе настройки `html` из соответствующего раздела правил.

Для отчетов первого типа **Dr.Web Notifier** проверяет с использованием правил (подробнее о них в разделе Секция [Rules]) необходимость отправления отчета каждому участнику из перечисленных:

- отправителю;
- получателям (при этом, если настройки по отчетам отличаются для разных получателей, то будет отправлено больше отчетов, чтобы каждый получатель получил отчет именно в таком виде, в котором он хочет);
- администратору.

Имя отчета получается прибавлением `sender_`, `rcpts_` и `admin_` к названию внешнего модуля с прибавленным к нему расширением `.msg`. Если такой файл не найден, то возникает ошибка.

Для отчетов второго типа **Dr.Web Notifier** отправляет только один отчет администратору, содержащий общую статистику по

работе комплекса. Шаблон для этого отчета содержится в файле `report.msg`. Отчеты второго типа отправляются компонентом **Dr.Web Notifier** независимо каждому Клиенту и Супер-Администратору, и содержат общую статистику по работе комплекса. Шаблон для отчета каждому Клиенту содержится в файле `report.msg`, а для отчета Супер-Администратору - в файле `report_sa.msg`.

Отчеты третьего типа являются DSN-отчетами о невозможности доставки письма и содержатся в файле `dsn.msg`.

Dr.Web Notifier производит загрузку всех файлов с шаблонами, которые удовлетворяют регулярному выражению `(admin|rcpts|sender|report|dsn)_?(.*?)\.msg`. В связи с этим существует возможность менять файл, из которого берется отчет, в зависимости от различных критериев. Для этого в правилах введена переменная **NotificationNamesMap**, которая производит отображение имени отчета, переданного в **Dr.Web Notifier**, в новое значение, из которого затем будет сформировано новое имя файла с шаблоном по вышеозначенным правилам. Отображение имеет смысл производить только в известное компоненту **Dr.Web Notifier** имя, так как в противном случае он просто не найдет требуемый файл.

С помощью данного механизма можно настраивать пользовательские файлы для отчетов второго и третьего типов.

Пример:

```
[Rule:client1]
...
NotificationNamesMap = report r1, dsn d1
...
[Rules]
md-client:client1 cont rule=client1
```

Это правило приведет к тому, что для Клиента `client1` в качестве отчета второго типа будет использоваться файл

report_r1.msg, а для DSN отчетов - файл dsn_d1.msg.

Секция [ProxyClient]

В секции [ProxyClient] находятся настройки модуля drweb-proxy-client, отвечающего за работу прокси-клиента программного комплекса **Dr.Web для почтовых серверов UNIX**.

ProxyServersAddresses = {список адресов}	Список адресов сокетов, на которых слушают компоненты drweb-proxy-server. 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. 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, отвечающего за работу прокси-сервера программного комплекса **Dr.Web для почтовых серверов UNIX**.

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

Секция [POP3]

Dr.Web MailD поддерживает работу с серверами POP3 через программу-фильтр протокола. Программа-фильтр POP3 является прокси-сервером между `drweb-maild` и программой-сервером POP3. Она фильтрует письма, отправляемые сервером пользователю. Программа-сервер POP3 может находиться как на локальном, так и на удаленном компьютере.

В секции `[POP3]` находятся настройки модуля `drweb-pop3`.

ServerAddress = {адрес сокета}	Адрес, по которому следует подключаться к серверу POP3. Значение по умолчанию: **ServerAddress** = `inet:pop3@localhost`
ListenAddress = {адрес сокета}	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, используемые для подключений в качестве TLS/SSL 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`. Подробное описание доступных параметров содержится в общем описании устройства конфигурационного файла.
	
	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, используемые для подключений в качестве TLS/SSL клиента. Настройки задаются через запятую. Подробное описание доступных параметров содержится в общем описании устройства конфигурационного файла. **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 = {численное значение}	Минимальное число соединений между фильтром POP3 и `drweb-maild`. Значение по умолчанию: **MinFilterToMaildConnections** = 2
MaxFilterToMaildConnections = {численное 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}	Параметры дополнительного пула потоков, обрабатывающих сигналы от `drweb-maild` об окончании обработки письма.
	Значение по умолчанию: **CallbackPoolOptions** = auto

MaxConnections = {численное значение}	Максимальное количество входящих соединений. Если указано значение 0, то количество входящих соединений не ограничено. Значение по умолчанию: **MaxConnections** = 0
DoS_Blackhole = {Yes | No}	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 = {Yes | No}	Запретить клиенту передачу имени и пароля в незашифрованном виде. Требует предварительной настройки OpenSSL. Значение по умолчанию: **DisablePlainText** = No
MaxConnectionsPerIp = {численное 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 = {действия}	Действие, применяемое к письму при ошибке, возникшей до отправки письма модулю drweb-maild. Возможные значения: reject либо pass. Значение по умолчанию: **OnFilterErrors** = reject

При каждом подключении POP3-фильтр 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.

Ответ сервера на последнюю команду передается процессу drweb-maild для обработки, а обработанный ответ уже передается пользователю.

Обратите внимание, что если POP3-фильтр блокирует сообщение в связи с настройками какого-либо из подключаемых модулей, а в настройках действий для писем, вызвавших ошибки сканирования, указано redirect - то это действие выполняться не будет. В текущей версии **Dr.Web для почтовых серверов UNIX** POP3-фильтр не имеет возможности связываться с компонентом **Sender** для отправки писем, поэтому несмотря на то, что письмо для перенаправления на адрес администратора формируется, оно никуда не отправляется.

Чтобы настроить взаимодействие POP3-фильтра с почтовой

системой, необходимо в mmc-файле для используемой почтовой системы (`maild_MTA.mmc`) раскомментировать строку:

```
drweb-pop3 local:/var/drweb/ipc/.agent 15 30
  MAIL drweb:drweb
```

Секция [IMAP]

Dr.Web MailD поддерживает работу с серверами IMAP (включая функцию кэширования) через программу-фильтр протокола. Программа-фильтр IMAP является прокси-сервером между `drweb-maild` и программой-сервером IMAP. Она фильтрует письма, отправляемые сервером пользователю. Программа-сервер IMAP может находиться как на локальном, так и на удаленном компьютере.

В секции `[IMAP]` находятся настройки модуля `drweb-imap`.

ServerAddress {адрес сокета}	=	Адрес, по которому следует подключаться к серверу IMAP. Значение по умолчанию: **ServerAddress** = inet:imap@127.0.0.1
ListenAddress {адрес сокета}	=	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, используемые для подключений в качестве TLS/SSL 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. Подробное описание доступных параметров содержится в общем описании устройства конфигурационного файла. **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 программы не поддерживается. Значение по умолчанию: **ServerTLSSettings** =
ClientTLSSettings = {настройки TLS/SSL}	Настройки, используемые для подключений в качестве TLS/SSL клиента. Настройки задаются через запятую. Подробное описание доступных параметров содержится в общем описании устройства конфигурационного файла. **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 программы 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 = {численное значение}	Минимальное число соединений между фильтром IMAP и drweb-maild. Значение по умолчанию: **MinFilterToMaildConnections** = 2
MaxFilterToMaildConnections = {численное 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}	Параметры 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 = {численное значение}	Максимальное количество входящих соединений. Если указано значение 0, то количество входящих соединений не ограничено. Значение по умолчанию: **MaxConnections** = 0
MaxConnectionsPerIp = {численое значение}	Запретить клиенту передачу имени и пароля в незашифрованном виде. Требует предварительной настройки OpenSSL. Значение по умолчанию: **MaxConnectionsPerIp** = 0
DisablePlainText = {Yes | No}	Запретить клиенту передачу имени и пароля в незашифрованном виде. Требует предварительной настройки OpenSSL. Значение по умолчанию: **DisablePlainText** = no
DoS_Blackhole = {Yes | No}	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 = {численное 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 отсутствие 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 {действия}	=	Действие, применяемое к письму при ошибке, возникшей до отправки письма модулю drweb-maild. Возможные значения: reject либо pass. Значение по умолчанию: **OnFilterErrors** = reject

Фильтр IMAP 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 фильтр IMAP 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.

Чтобы настроить взаимодействие IMAP-фильтра с почтовой системой, необходимо в mmc-файле для используемой почтовой системы (maild_MTA.mmc) раскомментировать строку:

```
drweb-imap local:/var/drweb/ipc/.agent 15 30
  MAIL drweb:drweb
```

Секция [Rules]

В секции [Rules] содержатся правила обработки почты. Они предназначены для тонкой настройки поведения программного комплекса в зависимости от конкретных нужд пользователя. С их помощью можно задать реакцию компонентов на конкретный набор элементов. К таким элементам относятся адреса отправителя и получателя, названия найденных вирусов или других объектов, вызывающих блокировку письма, а также различные дополнительные характеристики (IP-адрес отправителя, IP-адрес сервера получателя, размер письма и т. п.). В зависимости от произвольной комбинации данных параметров можно менять процедуру обработки сообщения. Кроме того, в правилах хранятся настройки для каждого **Клиента**: для этого у них должно быть только одно условие - уникальный идентификатор **Клиента**. Правила проверяются сверху вниз в порядке их задания.

Каждое правило имеет вид:

```
CONDITION stop| cont [ SETTINGS]
```

где CONDITION - условие, которое должно быть истинно, чтобы вступили в силу настройки, указанные в SETTINGS. Также можно с помощью CONDITION загружать настройки SETTINGS из внешнего источника (ldap, mysql и т.п.) в случае, если SETTINGS в правиле не указано.

Условия правил

Каждое из условий представлено в виде:

```
[ prefix_name: ][ value]
```

где prefix_name - имя параметра, а value - значение параметра.

Возможны следующие имена и значения параметров:

- **any** – либо отправитель, либо получатель; значение параметра - lookup;
- **from** (или **sender**) – отправитель; значение параметра

- lookup;

- **to** (или **rcpt**) – получатель; значение параметра - lookup;
- **block** – объект, вызывающий блокировку письма (например, название вируса); значение параметра - lookup;
- **client-ip** – IP-адрес отправителя письма (если получение информации об IP-адресе отправителя было задано в настройках компонента **Receiver**); значение параметра - список защищаемых сетей;
- **client-port** – порт клиента, отправившего письмо (если получение информации о порте клиента было задано в настройках компонента **Receiver**); значение параметра - номер порта;
- **server-unix-socket** – абсолютный путь к файлу UNIX-сокета, на котором было принято соединение (если получение информации об адресе сокета было задано в настройках компонента **Receiver**); значение параметра - путь к UNIX-сокету;
- **server-ip** – IP-адрес интерфейса, на котором Receiver принял письмо (если получение информации об IP-адресе интерфейса было задано в настройках компонента **Receiver**); значение параметра - список защищаемых сетей;
- **server-port** – порт сервера, на котором было принято соединение (если получение информации о порте сервера было задано в настройках компонента **Receiver**); значение параметра - номер порта;
- **id** – уникальный идентификатор **Receiver'a** принявшего письмо (если был задан в настройках компонента **Receiver**); значение параметра - строка с идентификатором;
- **auth** – прошел ли авторизацию клиент, отправивший письмо (если получение информации об авторизации клиента было задано в настройках компонента **Receiver**); значение параметра - не указывается;
- **size** – размер сообщения. Перед размером можно указывать отношения: {!= | == | < | > | <= | >=}. Если отношение не указано, то по умолчанию

предполагается, что это ≤. При использовании отношений необходимо заключать значения в кавычки, т.к. в отношениях используются служебные символы "! " и "=";

Пример:

```
"size:>=10m" cont scan=no
```

означает, что все сообщения размером больше 10 Мб должны быть пропущены без проверки. Обратите внимание на использование кавычек: они тут необходимы.

- **md-client** – уникальный идентификатор Клиента, для которого задаются настройки. Перед стартом **Dr.Web MailD** запрашивает у правил все настройки для каждого из активных Клиентов.

  Пример:

  ```
  "md-client:client1"
  ```

  данное условие будет истинным, если письмо получено для Клиента client1.

- **score** – счет сообщения. Перед размером можно указывать отношения: {!= | == | < | > | <= | >=}. Если отношение не указано, то по умолчанию предполагается, что это ≤. При использовании отношений необходимо заключать значения в кавычки, т.к. в отношениях используются служебные символы "! " и "=".

Если имя параметра не указано, то по умолчанию используется значение параметра **any**. Если в значении параметра содержатся пробелы или символы |&)(!=, то его надо заключать в кавычки. Чтобы задать символ " внутри кавычек, его надо предварять знаком \.

Также вместо выражения [prefix_name:][value] можно указывать специальные ключевые слова: true или false - которые всегда принимают положительное или отрицательное значение соответственно.

Пример:

```
true cont some_settings
```

Указанные настройки будут применяться всегда (если до этого правила дошла очередь при проверке).

Единичные условия можно объединять в сложные с помощью скобок и операторов `AND` (`&&`), `OR` (`||`), `NOT` (`!`) (в скобках представлено альтернативное написание).

Примеры:

```
sender: test && "size: >=10k"
```

Данное условие будет истинным, если отправителем письма является `test` и размер письма больше 10 килобайт.

```
!("rcpt: ldap: ///??sub?(mail=$s)" OR auth:)
```

Данное условие будет истинным, если хотя бы один получатель не найден в `ldap` по полю `mail` и отправитель не является авторизованным.

При обработке письма плагином, последний может запросить значение какого-либо параметра у **Dr.Web MailD**. Подходящий параметр выбирается согласно следующему алгоритму:

Прежде всего **Dr.Web MailD** проверяет, находится ли каждый из получателей письма в базе данных. Если он находится в БД, то поиск значения параметра осуществляется в сохраненных в БД настройках получателя, а также в настройках всех групп, в которые получатель входит.

Если в БД значение параметра не было найдено, то его поиск осуществляется в правилах, заданных в конфигурационном файле. Проверяются правила сверху вниз в порядке их задания. Прежде всего проверяется условие `CONDITION` - и если оно истинно, то значение параметра ищется среди элементов `SETTINGS`.

Если искомый параметр не найден, и после `CONDITION` стоит `stop`, то происходит поиск нужного параметра в секции с настройками по умолчанию **Dr.Web MailD**. Если искомый

параметр не найден, и после CONDITION стоит cont, то происходит дальнейшая проверка на соответствие остальным условиям.

Применение stop позволяет сократить время поиска параметра, если точно известно, что среди всех дальнейших условий, следующих за первым подходящим, больше не найдется соответствий.

Пример:

```
rcpt (sender, any): [адрес или регулярное выражение] stop| cont [настройки]
```

Это правило позволяет задать настройки для индивидуального пользователя.

Настройки правил

Настройки SETTINGS , в свою очередь, имеют вид:

```
[plugin_name/]param1 = value1, [plugin_name/] param2 = value2 ...
```

где plugin_name - название подключаемого модуля, к которому относится параметр, **paramN** - название параметра, а valueN - его значение.

Обратите внимание, что SETTINGS разбираются только когда в самом параметре возникла необходимость. Таким образом, при обычной загрузке конфигурационного файла ошибки в значениях параметров в правилах видны только в момент использования. Чтобы проверить правила до использования, надо использовать механизм проверки корректности настроек (с помощью параметра командной строки --check-only).

При задании настроек в базе данных полная их корректность проверяется немедленно, и правила с неверными параметрами блокируются для добавления в БД.

Пример:

```
sender: a@drweb.com cont headersfilter/Action =
```

```
pass, vaderetro/max_size = 100k
```

В этом случае, для отправителя `a@drweb.com` выбирается значение параметра **`Action = pass`** для плагина `headersfilter` и максимальный размер проверяемого сообщения (**`max_size`**), равный `100k`, для плагина `vaderetro`.

Если в `valueN` содержится запятая, то перед ней следует поставить обратный слэш "\ ".

Пример:

```
to: a@drweb.com cont drweb/ProcessingErrors = pass\, redirect(err@drweb.com)
```

В данном случае нельзя заключать значение `pass, redirect(err@drweb.com)` в кавычки, так как в этом случае парсер будет воспринимать его как одно значение и не разбивать на подстроки при разборе параметра **`ProcessingErrors`**.

`SETTINGS` можно не указывать. В этом случае предполагается, что параметры можно запросить непосредственно с сервера при помощи команды `lookup` в `CONDITION`. Это полезно, например, при работе с LDAP:

```
to: regex:.*@drweb.com && "ldap:///?drwebRules?sub?(mail=$s)" cont
```

здесь, если получатель находится в домене `drweb.com`, и отправитель или все получатели удовлетворяют ldap-условию `mail=$s`, то параметры берутся из поля `drwebRules`. 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 базе данных LDAP заключен в кавычки: это связано с присутствием в нем скобок.

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.

Вся обработка правил происходит сверху вниз и слева направо, и новые параметры будут затирать значение более старых, т.е. если в списке параметров задать **html**=yes, **html**=no, то в итоге будет установлено значение **html**=no.

Это правило относится почти ко всем параметрам, однако существует несколько параметров, которые обладают совершенно другой семантикой.

Каждый раз, когда находится такой параметр, его новое значение не затирает предыдущее найденное, а добавляется к нему. Соответственно, если параметр найден, то для него не прерывается поиск, а продолжается дальше по всем правилам (как в базе данных, так и в конфигурационном файле), для которых имеет место совпадение CONDITION. В результате, все найденные значения объединяются в одно.

Такой семантикой обладают следующие параметры:

- **LocalRules** для подключаемого модуля Dr.Web Modifier;
- **AcceptCondition** для подключаемого модуля headersfilter;
- **RejectCondition** для подключаемого модуля headersfilter;
- **AcceptPartCondition** для подключаемого модуля headersfilter;
- **RejectPartCondition** для подключаемого модуля headersfilter;
- **MissingHeader** для подключаемого модуля headersfilter;
- **WhiteList** для подключаемого модуля VadeRetro;
- **BlackList** для подключаемого модуля VadeRetro;
- **RegexsForCheckedFilename** для подключаемого модуля drweb.

Для каждого из таких параметров эта особенность описана явно.

Обратите внимание, что в том случае, если параметры, переопределяемые для адреса, пользователя или домена, хранятся в базе данных, они должны быть записаны в одну строку.

Пример:

Address	Rules
test1@drweb.com	VadeRetro/SubjectPrefix = \"spam\", modifier/localrules=select message\, append_text "Some Text"
test2@drweb.com	headersfilter/MissingHeader = Date, headersfilter/MissingHeader = From, headersfilter/MissingHeader = To

Также следует отметить, что `stop` для таких параметров работает как обычно - он прерывает поиск по правилам и возвращает текущее накопленное значение параметра.

Пример:

Допустим, база данных, настроенная через `odbc`, имеет следующий вид:

Address	Rules
test1@drweb.com	modifier/LocalRules = select message\, append_text "Scanned 3333!

В конфигурационном файле прописано следующее правило:

```
true cont modifier/LocalRules = select message\, append_text "Scanned 44444 - global rules!", modifier/LocalRules = quarantine
```

В настройках для соответствующего пользователя, хранящихся в БД, записаны 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!", modifier/LocalRules = quarantine
2: true cont modifier/LocalRules = select message\, append_text "Scanned 2222!"
3: "rcpt:odbc:select rules from maild where a='$s'" cont
custom:
```

В этом случае, с учетом всех настроек, если письмо идет получателю `test@drweb.com` и попадает на обработку подключаемому модулю Dr.Web Modifier, то для него будут использоваться следующие значения параметра **`LocalRules`**:

```
select message, append_text "Scanned!", quarantine, select message, append_text "Scanned 2222!", select message, append_text "Scanned 3333!", select message, append_text "Scanned 44444 - global rules!", quarantine
```

Обратите внимание на порядок следования значений: сперва идут значения из базы данных, затем - значения из конфигурационного файла.

Если в письме кроме `test@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 то правило, значение которого совпадает для всех получателей:

```
select message, append_text "Scanned 44444 - global rules!", quarantine
```

Если в строке с правилом при обработке будет найдена ошибка (во всех случаях, кроме обработки `lookups`), то она выводится в отчет, а само правило игнорируется.

Значения lookups и значения конкретных переменных не обрабатываются сразу - их разбор происходит только тогда, когда возникает реальная необходимость в их использовании. Соответственно, 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). Чтобы сразу проверить конфигурацию на наличие ошибок, рекомендуется запускать `drweb-maild` с параметром `--check-only`.

Секции [Rule]

При необходимости вы можете определить часто используемые группы параметров в специальных пользовательских секциях конфигурационного файла. Пользовательские секции задаются следующим образом

```
[Rule: <имя секции>]
```

где `<имя секции>` - уникальное название пользовательской секции, которое может содержать латинские символы, числа и пробелы. В секции задаются параметры, по одному параметру на строку. Окончанием секции считается либо начало следующей секции, либо окончание конфигурационного файла. Имена секций не зависят от регистра.

Обратите внимание, что в секциях `[Rule]` не определяются сами правила, там содержатся только параметры которые могут быть использованы в правилах, заданных в секции `[Rules]`. Параметры, определенные в пользовательской секции могут быть применены в правилах при помощи директивы `rule=<имя секции>`. При этом каждая пользовательская секция должна быть задана перед ее использованием в правилах. В текущей версии **Dr.Web MailD** в каждом правиле может быть задано не более одного параметра `rule`. Количество пользовательских секций не ограничено.

В конфигурационном файле есть особая секция

пользовательских параметров - секция параметров по умолчанию. Эта секция называется `default`, причем в заголовке секции ключевое слово `Rule` может быть опущено. В секции `default` устанавливаются значения по умолчанию для всех параметров, которые встречаются в правилах. Чтобы применить в правиле настройки по умолчанию, вы можете использовать параметр `rule=default`.

Пример:

Эти строки объявляют пользовательскую секцию MySection, которая задает два параметра (блокировать отчеты и отключать перемещение в карантин):

```
[Rule: MySection]
quarantine = no
notify = block
```

Следующие два правила используют эту пользовательскую секцию, чтобы установить соответствующие значения параметров `quarantine` and `notify`:

```
[Rules]
Rcpt: regex: example\.com cont rule=MySection
Sender: lol@foo.com && block: vir1 cont notify.
Skip=allow, notify.Virus=allow, rule=MySection
```

Поле того, как будут определены эти правила, будут заблокированы отчеты и перемещение файлов в карантин для писем, адрес получателя которых принадлежит домену example.com. Если же письмо отправлено с адреса lol@foo.com и найден блокирующий объект `vir1`, то будут разрешены только отчеты о найденных вирусах, а перемещение файлов в карантин будет запрещено. Обратите внимание, что в начале примера задан параметр `notify.Skip=allow`, но его значение оказалось "перекрыто" параметром `notify=block` из секции `MySection`.

Параметры, используемые в правилах

Существуют параметры следующих видов:

- те, которые встречаются только в правилах;
- те, значения которых задаются в конфигурационных файлах других модулей;
- те, которые предназначены для использования только **Клиентами** и должны задаваться с одним единственным условием: уникальным идентификатором **Клиента** (`md-client`).

Для каждого параметра второго типа в его описании в документации обозначено, можно ли его использовать в правилах. Ниже приведено описание параметров второго типа:

`html` = { `Yes` | `No`}	Значение `Yes` данного параметра указывает **Dr.Web MailD** формировать уведомления в html-формате, в противном случае отчеты генерируются в текстовом виде. Значение по умолчанию: **`html`** = `Yes`
`quarantine` = { `Yes` | `No`}	При значении `Yes` данного параметра письмо перемещается в карантин. Значение по умолчанию: **`quarantine`** = `Yes`

scan = {текст}	Данный параметр указывает, какие подключаемые модули **Dr.Web MailD** будет использовать для проверки письма. При значении `All` параметра, будут задействованы все подключаемые модули. При значении `No` параметра, никакие модули использоваться не будут. Разделителем между именами плагинов служит двоеточие ":". Чтобы исключить из использования какие-либо плагины, следует перед их именем ставить знак минус "-" без пробела между минусом и именем плагина. Обратите внимание, что при указании значения `All` не разрешается использовать имена плагинов без знака минус, так как это является бессмысленным. **Примеры**: **scan** = `all` - использовать все плагины; **scan** = `no` - не использовать ни одного плагина; **scan** = `all:-foo` - использовать все плагины, кроме `foo`; **scan** = `Foo:Bar` - использовать только плагины `Foo` и `Bar`; **scan** = `all:foo` - неверно, т.к. нельзя использовать имена плагинов без "-" после `all`; **scan** = `-foo:all` - неверно, т.к. `all` может быть только на первой позиции; **scan** = `-foo` - неверно, т.к. в начале отсутствует `all`.
	Значение по умолчанию: **scan** = `All`

notify[.{тип уведомления}] = {allow | block} [({типы адресов})]	Данный параметр управляет выводом уведомлений разных типов. Значение allow разрешает вывод соответствующего уведомления, значение block - запрещает. Если тип уведомления не указан, то значение параметра применяется ко всем уведомлениям. Возможные типы уведомлений зависят от того, какие виды отчетов поддерживает модуль drweb-notifier. Дополнительно установленные плагины 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 пропущенных письмах; • **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, уведомления для каких типов адресов подпадают под действие параметра. Можно указать несколько видов адресов, разделенных двоеточиями. Возможные значения модификатора:

- `sender` – уведомления отправителю письма;
- `rcpt` – уведомления получателям письма;
- `admin` – уведомления администратору;
- `any`, либо модификатор отсутствует – уведомления для всех типов адресов.

Примеры:

Notify=`block` или **notify**=`block (any)` – блокирование всех типов уведомлений.

notify. Virus=`block (sender: admin)` – блокирование уведомлений о найденных вирусах для администратора и отправителя письма.

Если правило для заданного типа уведомления не найдено, а также не найдено общее правило (для которого тип не задан), то предполагается, что уведомление отключено.

Значения параметра **Notify** объединяются по тем же правилам, что и значения параметра **LocalRules** для подключаемого модуля **Dr.Web Modifier**.

	Например, если имеются следующие правила:
	`true cont` **`notify.virus`** `= allow (sender)`
	`true cont` **`notify.virus`** `= allow (admin),` **`notify`** `= block`
	то они аналогичны правилу:
	`true cont` **`notify.virus`** `= allow (sender: admin),` **`notify`** `= block`
	Т.е. в результате будут отправляться только отчеты об обнаруженном вирусе администратору и отправителю письма, а все остальные отчеты будут заблокированы.

	Значение по умолчанию: По умолчанию, устанавливаются следующие параметры уведомлений: • **`notify`** `= block(any)` • **`notify.Virus`** `= allow(any)` • **`notify.Cured`** `= allow (admin: sender)` • **`notify.Skip`** `= allow (sender)` • **`notify.Archive`** `= allow (admin: sender)` • **`notify.Error`** `= allow (admin: sender)` • **`notify.Rule`** `= allow (admin)` • **`notify.License`** `= allow (admin)`
`plugin_name/ max_size` `= {размер}`	Этот параметр позволяет указать максимальный размер проверяемого сообщения для каждого подключаемого модуля.

plugin_name/use = {Yes | No}	Значение yes подключает использование указанного модуля для проверки конкретного письма, а значение no предписывает пропустить проверку письма указанным модулем.
NotificationNamesMap = name1 file_name1, name2 file_name2 ...	Позволяет отображать названия отчета в новое. Например, может использоваться для назначения других отчетов в зависимости от конверта. Здесь, соответственно: • nameN – имя запрашиваемого отчета, для которого устанавливается новый файл. Список имен можно найти чуть выше в описании настройки notify. Кроме того, здесь можно для общих отчетов указывать слово "report", а для DSN - "dsn". • file_nameN – часть имени нового файла с отчетом. Полное имя составляется по следующему правилу: 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 директории, указанной 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: "to:mysql:select * from adr" cont SenderAddress = address1|address2|address3 сообщение, удовлетворяющее условию "to:mysql:select * from adr", 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. Если **Sender** поддерживает данный параметр, то он передаст письмо именно на указанный адрес. На данном этапе этот параметр поддерживает только модуль drweb-sender с SMTP/LMTP методом отправки почты.
rule = {название секции}	Применить параметры из пользовательской секции с заданным названием ([Rule:название секции]). См. выше.

Рассмотрим случай с несколькими получателями в письме. Все следующие параметры:

plugin_name/max_size

NotifyLangs

AdminMail

html

scan

plugin_name/use

а также все параметры третьего типа, которые относятся к настройкам Клиентов, обрабатываются для каждого получателя отдельно: т.е. если у двух получателей для какого-либо параметра будут указаны разные значения, то для каждого из получателей будет сделана отдельная копия письма, к которой будут применены свои настройки.

Для остальных параметров, если `CONDITION` выполняется для каждого получателя, то используется значение параметра из этого правила, если не выполняется - используется значение по умолчанию.

Для проверки корректности правил в модуле `drweb-maild` через параметры командной строки доступен специальный интерфейс. С помощью этих параметров задаются различные свойства предполагаемого письма, и модуль выводит на консоль все настройки из правил, которые будут применимы к данному письму. Сейчас доступно задание следующих свойств:

- `-s [ --sender ] arg` – отправитель письма (из конверта);
- `-r [ --rcpt ] arg` – получатель письма (из конверта). Для задания нескольких получателей, надо несколько раз указать данный параметр;
- `-b [ --block ] arg` – блокирующий объект, найденный в письме (например, название вируса). Для задания нескольких блокирующих объектов, надо несколько раз указать данный параметр;
- `--client-ip arg` – IP-адрес клиента, от которого получено письмо;
- `--server-ip arg` – IP-адрес интерфейса сервера, на который получено письмо;
- `--client-port arg` – порт клиента, с которого было получено письмо;
- `--server-port arg` – порт сервера, на который получено письмо;
- `--server-us arg` – название UNIX-сокета сервера, на который получено письмо;

- `--id arg` – уникальный идентификатор **Receiver**'а, от которого получено письмо;
- `--auth` – письмо получено от авторизованного пользователя;
- `--size arg` – размер проверяемого письма (значение имеет тип `{size}`);
- `--score arg` – счет письма;
- `--md-client arg` – уникальный идентификатор Клиента.

Пример:

```
$ ./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
```

Lookups

`lookup` - это обобщенный интерфейс для 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.

Также, в составе lookup-запросов к базам данных 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 `$`.

Существуют следующие варианты префикса:

- `value` – за ним указывается непосредственно искомое значение. Этот префикс используется, если, к примеру, в 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;
- `regex` – значение является регулярным выражением (совместимым с регулярными выражениями Perl) - при проверке ищется подстрока, а не полное совпадение;
- `rfile` – значение является путем к файлу. Файл содержит набор регулярных выражений (совместимых с регулярными выражениями Perl), каждое из которых должно находиться в новой строке. При проверке ищется подстрока, а не полное совпадение;
- `ldap` – значение представляет собой путь к поиску на LDAP-сервере;

  Формат значения следующий:

  `[` **`param1`**`=val1|` **`param`**`2=val2|...|] ldap_url`

  где `ldap_url` - это интернет-адрес LDAP запроса, а **`param1`** и т.п. это локальные параметры из секции `[LDAP]` конфигурационного файла **Dr.Web MailD** для данного `lookup`. Из данной секции можно указывать только те параметры, для которых явно написано, что это возможно.

Интернет-адрес 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=*)
```

Dr.Web MailD работает с LDAP через библиотеку `OpenLDAP` (должна быть не ниже версии 2.0);

- `odbc`, `oracle` – значение представляет собой SQL-запрос к хранилищу ODBC или Oracle. Формат значения следующий:

```
[param1=val1|param2=val2|...|] sql_request
```

где **param1** и т.п. это локальные параметры из секций `[ODBC]` и `[Oracle]` конфигурационного файла **Dr.Web MailD** для ODBC `lookup` и Oracle `lookup`, соответственно. Из данной секции можно указывать только те параметры, для которых явно написано, что это возможно. В запросе можно указывать те же специальные символы, что и в `ldap`.

Новые настройки DSN будут активированы только после перезапуска комплекса (сигнал `SIGHUP` не форсирует переинициализацию соединений).

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.

С Oracle **Dr.Web MailD** работает через библиотеку `libclntsh`, которая поставляется совместно с клиентом Oracle и поддерживает версию `OTLv8` или выше.

Для подключения к Oracle необходимо указать в значении параметра **`ConnectionString`** имя пользователя, пароль и название подключения: `user/password@connectionname`.

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 БД 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)))
```

- `postgres` – значение представляет собой SQL-запрос к хранилищу PostgreSQL. Синтаксис аналогичен запросу к ODBC;
- `cdb` – значение представляет собой текстовое имя ключа в базе данных. Сама база данных CDB представляет собой доступное только для чтения хранилище пар `[текстовый ключ]:[текстовое значение]`. Для создания файла базы данных можно использовать пакет `tinycdb`. Для инициализации данного `lookup` следует указать список файлов баз данных CDB. Каждый файл представлен в виде одной таблицы, названием которой является только имя файла (без указания полного пути к нему: `/path/to/table.cdb --> table.cdb`).

  Считать запись из таблицы можно при помощи следующих команд:

  ```
  select * from table.cdb where key='123'
  select * from table.cdb where key='$'
  ```

  База данных CDB не поддерживает язык запросов SQL.

Поэтому драйвер эмулирует SQL единственную команду для унификации работы с `lookups`:

```
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 те же специальные символы, что и в `ldap`.

Пример:

```
cdb:skipdomains=regex:^inbox|select * from my_file where key='$s'
```

- `berkeley` – обеспечивает взаимодействие с Berkeley DB. Формат запроса аналогичен `cdb`. 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`). `Lookup` предназначен для работы с библиотеками версий 4.3-4.6. В запросе можно указывать те же специальные символы, что и в `ldap`;
- `firebird` – значение представляет собой SQL-запрос к хранилищу Firebird. Синтаксис аналогичен запросу к ODBC. Для подключения к хранилищу необходимо в поле **`Host`** указать по какому адресу находится сервер базы данных:

  `Host``=somehost` `#somehost:3050`

  `Host``=somehost/1234` `#somehost:1234`
- `sqlite` – значение представляет собой SQL-запрос к хранилищу SQLite. Синтаксис аналогичен запросу к ODBC. Данный `lookup` предназначен для работы с базами данных SQLite версии 3.x.

  В силу особенностей работы SQLite файл базы данных блокируется каждый раз, когда в него осуществляется запись. Соответственно, 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** секции [SQLite] конфигурационного файла **Dr.Web MailD**. 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.

В запросе можно указывать те же специальные символы, что и в ldap.

Если база данных SQLite, из которой брались настройки для ряда параметров (с помощью правил с lookups к 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;

- mysql – значение представляет собой SQL-запрос к хранилищу MySQL. Синтаксис аналогичен запросу к ODBC.

После префикса можно (но не обязательно) указывать список локальных параметров для lookups в формате:

```
NAME1 = VALUE1 | NAME2 = VALUE2 | ... |
```

где NAMEM - имя параметра (не зависит от регистра), VALUEN - значение параметра.

Доступны следующие локальные параметры:

- **SkipDomains** - список доменов, для которых не надо

выполнять lookup-запрос.

- **OnError** = {ignore | exception} - задает процедуру обработки ошибок в lookups. При значении ignore - игнорировать ошибку (происходит только регистрация ошибки в отчете), при значении exception - генерировать исключение, которое будет обрабатываться как стандартная ошибка (например, с помощью параметра **ProcessingError**). Значение exception полезно использовать, если в компании принята соответствующая политика безопасности, и каждое из писем должно быть так или иначе обработано, несмотря на возможные ошибки в остальных компонентах системы (например, со стороны базы данных, к которой обращается lookup).

Параметр **OnError** также можно записывать в каждую из секций lookups (LDAP, ODBC, и т.п.), и тогда его значение будет применяться для всех lookups данного типа, если для них не указано соответствующие локальное значение параметра.

Обратите внимание, что этот способ обработки ошибок с помощью параметра **OnError** работает исключительно на этапе собственно поиска значений с помощью lookups, а не на этапе запуска **Dr.Web MailD**. Соответственно если при запуске **Dr.Web MailD** не удается корректно разобрать lookups, то это все равно воспринимается как фатальная ошибка - независимо от настройки параметра **OnError**.

Примеры использования lookups

```
ProtectedDomains = "odbc:select domain from maild where domain='\$s'"
```

С помощью этого запроса отмечаются все письма, домен которых был найден в столбце domain таблицы maild в хранилище ODBC, как принадлежащих защищаемому домену.

```
ProtectedEmails = file:%etc_dir/email.ini,
```

```
localhost, ldap:skipdomains=regex:.*fake.com$|
ldaps:///??sub?(mail=$s)
```

С помощью этого запроса отмечаются как защищаемые следующие почтовые адреса: все адреса, находящиеся в файле `%etc_dir/email.ini`, адрес `localhost` и все адреса, которые были найдены по ldap-запросу `ldaps:///??sub?(mail=$s)` (за исключением адресов, заканчивающихся на `fake.com` - для них запроса не происходит).

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 `lookups` могут использоваться в правилах.

```
"rcpt:ldap:///?rules?sub?(mail=$s)" cont
```

Запрос `"rcpt:ldap:///?rules?sub?(mail=$s)" cont` 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 `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'
```

По запросу `"any:sqlite:select skipaddr from domain where skipaddr = '$s'" cont scan=all:-drweb` проверяются адреса, и если адрес отправителя или получателя содержится в поле `skipaddr` таблицы `domain` базы данных SQLite, то для них не будет использоваться плагин `drweb`. В данном примере также можно отметить использование кавычек.

Ограничения по использованию lookups

Существуют некоторые ограничения по использованию определенных типов `lookups`.

`LookupsLite` - это тип значений аналогичный `lookups`, в котором можно указывать только либо непосредственное значение, либо `lookups` типа `file`. Он используется:

- в настройках самих `lookups` (например, в настройке **`SkipDomains`** для каждого `lookup`);
- во всех настройках плагинов.

При попытке задать запрещенный `lookup` в каком-либо из случаев, перечисленных выше, в лог 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.
```

Проверка lookups

Для проверки правильности создания запросов с помощью `lookup` можно использовать специальную утилиту `drweb-lookup`. Данная утилита запускается следующим образом:

```
$ %bin_dir/drweb-lookup [параметры] запрос
```

где `запрос` - это собственно различные типы `lookups`, где будет производиться поиск, а `[параметры]` — это параметры командной строки.

Доступны следующие параметры:

- `-h [ --help ]` – вывод помощи;
- `-v [ --version ]` – вывод текущей версии;
- `-l [ --level ] arg` – уровень подробности ведения протокола работы компонента;
- `--syslogfacility arg` – служба `syslog`;
- `-i [ --ipc-level ] arg` – уровень подробности протокола работы библиотеки `IPC`;
- `--log-filename arg` – имя файла отчета;
- `-a [ --agent ] arg` – путь к **Агенту**, откуда можно получить дополнительную конфигурацию по используемым `lookups`;
- `-t [ --timeout ] arg` – время ожидания на взаимодействие с **Агентом** для получения конфигурации;
- `-q [ --query ] arg` – строка, значение которой мы ищем. Если указано "-", то читаем со стандартного входа;
- `-e [ --exist ]` – требуется только проверка наличия запрашиваемого элемента в `lookups` без получения значения для него.

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
```

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 из себя 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], а также параметра **StatDetail**, который можно задавать в правилах для каждого Клиента отдельно.

- `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. Также включается сбор статистики по всем Клиентам. 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 модуля **Dr.Web MailD** с помощью типа `storage`. Причем обе эти возможности можно включить одновременно.

Экспорт статистики через **Dr.Web Agent** по умолчанию выключен и может работать только для статистики по всем Клиентам. В то время как экспорт с помощью типа `storage` можно настраивать как для Супер-Администратора, так и для каждого Клиента отдельно.

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 защиты **Dr.Web**, если работает в `Enterprise`-режиме (подробнее смотрите секцию `[EnterpriseMode]` конфигурационного файла `agent.conf`

).

Синтаксис `storage` аналогичен `lookup` (смотрите раздел Dr.Web Monitor), за исключением иного списка префиксов и того, что в нем нельзя использовать `$s`. Существуют следующие варианты префиксов:

- `odbc` – синтаксис аналогичен тому же в LDAP.

  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), `type` - тип параметра, под которым надо сохранять его в базе данных. Для каждого объекта приводится его тип по умолчанию, менять его не рекомендуется.

  Типы по умолчанию:

  - `varchar_long` – аналог `SQL_LONGVARCHAR` в ODBC;
  - `timestamp` – аналог `TIMESTAMP_STRUCT` в ODBC;
  - `int` – 32-битное знаковое целое число;
  - `char(length)` – строка, оканчивающаяся символом конца строки.
- `oracle` – синтаксис аналогичен тому же в ODBC;
- `postgres`, `mysql`, `sqlite`, `firebird` – синтаксис аналогичен тому же в ODBC, за исключением того, что тип `char(length)` не поддерживается, и для строковых данных следует использовать тип `varchar_long`.

Пример:

ExportStatStorage = "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.

Для включения экспорта статистики через тип `storage` по всем Клиентам прежде всего, 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 плагина.

Подробнее описание каждого из приведенных выше параметров можно найти в главе Секция [Stat].

Чтобы включить экспорт статистики отдельно для каждого Клиента, требуется задать параметры **ExportStat**, **ExportBlockObjectsStorage**, **ExportStatStorage**, **ExportPluginStatStorage** в правилах для каждого Клиента отдельно, аналогично тому, как они задаются в секции `[Stat]`.

Пример:

```
[Rule:client1]
...
ExportStat = yes
      ExportBlockObjectsStorage = "odbc:insert into client1_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>)"
...
```

Карантин

Письма попадают в карантин как по запросу от основного модуля `drweb-maild`, так и по запросу от любого из подключенных плагинов. Далее они сохраняются в директории `/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 директории `/path/def/backup/`.

Кроме сохранения тела письма, в директории 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 директории 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 DBI-хранилище занимается внешняя программа `drweb-qp`, путь к которой указывается в настройке **PathToDrwebQp**. Для ее работы необходим Perl (версии не ниже 5.0). Запуск данной программы 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 из карантина не в файловой системе, а в DBI-хранилище. Для этого необходимы Perl (версии не ниже 5.0), установленные модули `DBI` и `File::Temp` и настроенное DBI-хранилище. 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 исключительно через утилиту `drweb-qp`, путь к которой указывается в значении параметра **`PathToDrwebQp`**.

Для использования DBI необходимо установить **`MoveToDBI`** в `Yes` и настроить **`DBISettings`**, **`DBIUsername`** и **`DBIPassword`** соответствующим образом для доступа к DBI-хранилищу.

Также надо настроить соответствующие SQL-команды для выполнения требуемых действий:

- **`SQLInsertCommand`** – команда добавления письма в DBI-хранилище.
- **`SQLRemoveCommand`** – команда для удаления письма из DBI. Используется, если задано ограничение на время хранения писем в карантине.
- **`SQLSelectCommand`** – команда доступа к сохраненному сообщению в DBI-хранилище. 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`** секции `[Notifier]` конфигурационного файла **Dr.Web MailD**, либо в локальных правилах для данного письма. Настройка ACL 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` 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 при нажатии соответствующей ссылки в отчетах, высылаемых службой уведомления программного комплекса при помещении того или иного сообщения в карантин.

Управление через drweb-qcontrol

Также для управления карантином и осуществления поиска в нем предназначена специальная программа `drweb-qcontrol`. Ее интерфейс в большинстве случаев не зависит от того, где хранятся письма в файлах (на диске или в DBI-хранилище). Если

карантин хранится локально, то для работы с ним требуется, чтобы `drweb-maild` был запущен.

Доступны следующие параметры командной строки:

- `-h [ --help ]` – вывод справки;
- `--version` – вывод версии программы;
- `-v [ --verbose ]` – вывод на консоль всей информации о совершаемых программой действиях;
- `-l [ --level ] arg (=error)` – уровень подробности протокола работы модуля. Возможные значения: `Quiet`, `Error`, `Alert`, `Info`, `Debug`;
- `-i [ --ipc-level ] arg (=error)` – уровень подробности протокола взаимодействия с модулем `drweb-maild`. Возможные значения: `Quiet`, `Error`, `Alert`, `Info`, `Debug`;
- `--syslogfacility arg (=mail)` – тип подсистемы, через которую системный сервис `syslogd`, ведущий протоколирование работы модуля, выдает сообщения о событиях. Возможные значения: `Daemon`, `Mail`, `Local0-7`;
- `--log-filename arg (=syslog)` – имя файла отчета или значение `syslog`, если протоколирование работы модуля осуществляется с помощью системного сервиса `syslogd`;
- `--sendmail arg (=/opt/drweb/drweb-inject)` – путь к программе `drweb-inject`;
- `-s [ --socket ] arg (=local:/var/drweb/ipc/.ctl)` – путь к управляющему сокету **Dr.Web MailD**;
- `--agent arg (=local:/var/drweb/ipc/.agent)` – путь к сокету **Агента**, для получения от него конфигурации;
- `--timeout arg (=60)` – максимальное время ожидания ответа от **Агента** при получении конфигурации.

Список писем, над которыми выполняются действия, определяется через уникальные идентификаторы: относительные пути сохраненных в карантине файлов. При задании этих идентификаторов можно использовать специальные символы: "%" - ноль или более произвольных символов, "_" - один произвольный символ. При задании идентификаторов def/ в начале указывать обязательно.

Пример:

def/%00014F7F% – все сообщения с номером 00014F7F, сохраненные в карантине;

def/drweb/% – все сообщения, сохраненные плагином drweb.

Идентификаторы файлов, над которыми требуется выполнить операции, берутся из условий поиска или из командной строки (первые два места объединяются), либо из стандартного потока ввода (если не было указано ни одного условия поиска, и в командной строке не было указано ни одного идентификатора).

Возможные действия:

- --view – просмотреть с помощью программы, указанной в переменной окружения PAGER, все письма с заданными идентификаторами. Если в PAGER не указано значение, то используется программа cat.
- --send – отправить все письма с заданными идентификаторами оригинальным получателям. Для отправления используется программа drweb-inject.
- --redirect [list_of_rcpts] – переслать все письма с заданными идентификаторами на адреса из списка. Для отправления используется программа drweb-inject.
- --remove – удалить из карантина все письма с заданными идентификаторами.
- --stat – вывод статистической информации о найденных в карантине сообщениях.

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 действий потребуется указание дополнительных SQL-команд в командной строке:

- `--sql-remove-command` – команда удаления письма из карантина по его файловому идентификатору. Единственным параметром тут является файловый идентификатор.

  Пример:

  ```
  --sql-remove-command "DELETE FROM mail_export WHERE filename LIKE ?"
  ```

Программа `drweb-qcontrol` предоставляет также простой интерфейс для поиска в сохраненных письмах. 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.

Пожалуйста, обратите внимание, что если в качестве аргументов для параметров `--search-headers` и `--search-inbody` указываются спецсимволы `*`, `^`, `$` - то для того, чтобы поиск работал, их необходимо экранировать символом "\".

Пример:

```
--search-inbody \* --stat
```

Каждый из указанных критериев поиска проверяется независимо, т.е. они объединяются по принципу `OR`.

Пример:

```
--search-to addr1 --search-to addr2
```

будет искать письма, в конверте которых среди получателей есть либо `addr1`, либо `addr2`.

Пример:

```
--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`.

Если одновременно указаны какой-либо критерий поиска и список файлов в командной строке, то поиск будет производиться исключительно в списке файлов из командной строки.

Пример:

```
drweb-qcontrol --stat --search-from ai@5 def/
backup/%
```

выводит на консоль информацию обо всех заархивированных сообщениях, 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
```

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 директории `%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.

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 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 IPv4, так и IPv6 формат.

Ниже приводится список общих команд с их описаниями:

- **help** `[section| command]` – вывод справки по имеющимся секциям команд. После 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 по команде **help** `all`.
- **option** `[regex]` – вывод значений настроек, с которыми работает как модуль `drweb-maild`, так и загруженные плагины (которые получили свои настройки через `drweb-maild`), и имена которых совпадают с заданным регулярным выражением. Если регулярное выражение на задано, то выводятся все настройки;
- **db-state** – вывод текущего состояния внутренней 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. `0` в `NM` или `SM` указывает на отсутствие ограничений сверху;
- **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 `drweb-maild` (контролируемом параметром `OutPoolOptions`);
- **send-stat** – форсирует отправление/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 Агенту, а также экспорт статистики для всех Клиентов, которые включили эту возможность;

- **send-report** [period] – форсирует отправление письма с отчетом по работе подключаемых модулей для каждого из Клиентов, 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 выполнение программы 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;

  Соответственно, idN – номера запрашиваемых сообщений, 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 плагина, который попросил задержки сообщения в БД. Символ "-" эквивалентен отсутствию параметра. При отсутствии параметров выводятся все сохраненные в БД

сообщения.

Пример:

get – drweb – вывод всех сообщений, задержанных плагином 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. для них **send**=no в выводе **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 директории, в которую нужно производить сохранение файлов. Если директория не указана, 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.
- **notify** – проверка генерации уведомлений. Подробно команда описана в файлах notify.* в каталоге с документацией по продукту.
- **version** – вывод текущей версии продукта;
- **stop** – остановка продукта;
- **reload** – отправление процессу drweb-maild сигнала SIGHUP.

Управление пользователями, группами и алиасами

Специфические настройки для каждого получателя письма можно гибко задавать в правилах. Кроме того, в правилах можно объединить произвольное число пользователей в группы и присвоить уже группам определенные настройки, которые будут применены к каждому пользователю соответствующей группы.

В случае, если число пользователей велико, этот подход неэффективен, так как сложность поиска настроек пропорциональна количеству правил. В связи с этим возможно сохранять правила для каждого пользователя в локальную базу данных. В этом случае поиск правил более эффективен, и, кроме того, оптимизируется использование памяти.

Для каждого пользователя в базе данных кроме, собственно, его правил, сохраняется дополнительная информация. Вся информация о пользователе выводится по команде `email-info` для управляющего сокета. Формат вывода этой информации следующий:

```
[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..
...
```

Здесь:

- `client-id1` – идентификатор клиента, к которому относится пользователь;
- `A=active1` – активен пользователь или нет. Если пользователь не активен, то все связанные с ним правила игнорируются;
- `S=stat1` – 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 для конкретного пользователя.

Пользователей можно объединять в группы. С каждой группой связан тот же набор настроек, что и для пользователя: идентификатор клиента, название группы, активна ли группа, вести ли для нее отдельную статистику, список пользователей, входящих в группу, а также дополнительная служебная информация. Вся информация о группе выводится по команде `groups-info` для управляющего сокета. Формат вывода этой информации следующий:

```
[client-id1/]group1 A=active1 S=stat1
emails:
email1
email2
...
custom:
tag1: info1..
tag2: info2..
...
```

Если для одного пользователя имеется несколько почтовых адресов, то можно все их объединить, выделив один главный адрес, а остальные установив для него как алиасы. В результате, все эти адреса будут считаться единым целым и для них будет использоваться одинаковый набор настроек и будет собираться единая статистика.

При выборе значения параметра, которое следует применить к обрабатываемому письму, используется следующий алгоритм:

- Сперва поиск происходит в правилах для пользователя.
- Если в правилах пользователя значение параметра не было обнаружено, то происходит поиск в правилах для групп, в которые входит пользователь. Поиск осуществляется, начиная с настроек самой последней в списке группы, и до первой в списке группы - пока необходимое значение не будет найдено.
- Если в правилах для пользователя и группы значение параметра не найдено, то берется значение данного параметра из правил для Клиента, заданных в

конфигурационном файле.

- Если и там значение параметра отсутствует, то осуществляется поиск в глобальных настройках из конфигурационного файла.
- Если поиск в конфигурационном файле не увенчался успехом, то берется значение по умолчанию, вшитое в программу.

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 (описанных в главе Секция [Rules] данного руководства) 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 Клиента, либо глобальная настройка из конфигурационного файла, либо берется значение по умолчанию, вшитое в программу.

Обратите внимание, что про проверке 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.

Управление пользователями, группами пользователей и алиасами осуществляется с помощью специальных команд. В

командах используются следующие общие понятия:

- **Клиент** – администратор **Dr.Web для почтовых серверов UNIX**. Ему присвоен пустой идентификатор.
- **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.
- **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** – строка из правил **Dr.Web MailD**. 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.

Вывод результата выполнения каждой команды заканчивается пустой строкой.

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` для `Клиента`, 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` для `Клиента`, 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 настроек или 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 и для Клиента, заданного в client-email. Если email или Клиент не существуют, 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 и Клиента, 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 и Клиента, 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 для Клиента, заданного 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 для Клиента, заданного 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 или настроек 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 и для Клиента, заданного 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 и для Клиента, заданного в client-group. Нумерация (index) начинается с 1. Если group или Клиент 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 для всех Клиентов. Если встречаются только идентификаторы Клиента, то выводится информация по всем группам этого Клиента. 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 через управляющий сокет

Работа с карантином через управляющий сокет осуществляется с помощью специальных команд. В командах используются следующие общие понятия:

- **Клиент** – администратор **Dr.Web для почтовых серверов UNIX**. Ему присвоен пустой идентификатор.
- **id** – представляет собой 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 идентификатор def/drweb/E/00020EBE.maild.xeAX4u ссылается на письмо, тело которого расположено в файле /var/drweb/infected/def/drweb/E/00020EBE.maild.xeAX4u.

  Здесь:
  - def – идентификатор Клиента (или слово "def" - если идентификатор не назначен);
  - drweb – плагин, заблокировавший письмо. Если письмо заблокировано самим компонентом drweb-maild, 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 плагином

drweb.

Тема письма сохраняется в базе данных в декодированном виде (в кодировке UTF8), 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 плагина 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 плагина `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-remove client2/drweb/
E/00020EFE.maild.Q5FRbO
```

Из карантина удаляется письмо с заданным идентификатором.

- **quarantine-limits** – вывод текущих ограничений, установленных для карантина.

  Формат вывода:

  ```
  client-id1: 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-id1 – идентификатор Клиента, для которого выводится информация.
  - 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. Формат вывода определяется по результатам отправки для каждого письма:

  ```
  RES in sending (to RCPTS_LIST): id
  ```

  где:

- RCPTS_LIST – фактический список получателей письма.
- RES – OK или ERROR, в зависимсоти 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 через управляющий сокет

С помощью командного интерфейса управляющего сокета можно получать статистику по работе **Dr.Web для почтовых серверов UNIX** для Клиентов, пользователей и групп. Получение статистической информации осуществляется с помощью специальных команд. В командах используются следующие общие понятия:

- **Клиент** – администратор **Dr.Web для почтовых серверов UNIX**. Ему присвоен пустой идентификатор.
- **email** – почтовый адрес (в соответствии с RFC5322). Он может быть заключен в угловые скобки <>. Также его можно заключать в одинарные кавычки (' ').
- **client-email** = [client-id/]email, где client-id – уникальный идентификатор Клиента. Он может быть пустым для Клиента по умолчанию. client-id – регистрозависимый.
- **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 ', то перед ним должен ставиться повторный символ '.

- **client-group** = [client-id/]group - где client-id - уникальный идентификатор Клиента. Он может быть пустым для Клиента по умолчанию. client-id - регистрозависимый.

При работе со статистикой следует учитывать, что общая статистика по проверенным сообщениям для Клиентов и пользователей/групп собирается по-разному:

- статистика для Клиентов собирается во внутренний кэш, который сбрасывается раз в 5 минут во внутреннюю базу данных (сохранение также происходит при выполнении команды dump-cache-stat, при получении сигнала HUP и при завершении работы комплекса **Dr.Web для почтовых серверов UNIX**);
- 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, в то время как статистика по Клиентам сохраняется в базе данных периодически, и таким образом в данном случае имеется задержка в появлении общей статистики, которая составляет, в худшем случае, 5 минут.

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 плагина, где name – имя плагина, по работе которого необходимо предоставить статистику. Если параметр **plugin** не указан, то выводится информация по всем плагинам. Если указан несуществующий плагин, 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** client-id| *| - [period] [ignore] [plugin] – получение статистики по заданному Клиенту client-id. Если Клиента с указанным client-id не существует, то выводится ошибка и выполнение команды прерывается. Если вместо client-id указано *, то выводится статистика по всем Клиентам, если же указано "-" (без кавычек), то выводится статистика для Клиента по умолчанию (с пустым идентификатором). Факультативные параметры могут идти в произвольном порядке.

- **stat-group** client-group [period] [ignore] [plugin] – получение статистики для группы client-group. Если группа существует, то выводится ошибка и выполнение команды прерывается. Факультативные параметры могут идти в произвольном порядке.
- **stat-email** client-email [period] [ignore] [plugin] – получение 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** client-id|*|- [period] [ignore] [plugin] – удаление статистики по заданному Клиенту client-id. Если Клиента с указанным client-id не существует, то выводится ошибка и выполнение команды прерывается. Если вместо client-id указано *, то удаляется статистика по всем Клиентам, если же указано "-" (без кавычек) , то удаляется статистика для Клиента по умолчанию (с пустым идентификатором). По результатам выводится количество удаленных записей.
- **stat-remove-group** client-group [period] [ignore] [plugin] – удаление статистики для группы client-group. В результате выводится количество удаленных записей.
- **stat-remove-email** client-email [period] [ignore] [plugin] – удаление статистики для конкретного пользователя client-email. В результате выводится количество удаленных записей.
- **remove-old-stat** [time] – удаление всей статистики по всем Клиентам, 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** – сброс внутреннего кэша общей

статистики по Клиентам 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 плагина, по которому выводится статистика. Если указано *, то выводится общая статистика по всему комплексу (например, здесь учитываются сообщения, которые не были пропущены ни через один плагин).
- 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 – число/размер в байтах сообщений, для которых было выполнено действие pass;
- R/RS – число/размер в байтах сообщений, для которых было выполнено действие reject;
- D/DS – число/размер в байтах сообщений, для которых было выполнено действие discard;
- T/TS – число/размер в байтах сообщений, для которых было выполнено действие tempfail;
- Q/QS – числ/размер в байтах сообщений, для которых было выполнено действие quarantine;
- RE/RES – число/размер в байтах сообщений, для которых было выполнено действие redirect;
- N/NS – число/размер в байтах сообщений, для которых было выполнено действие notifiy;
- C/CS – число/размер в байтах чистых сообщений;
- S/SS – число/размер в байтах сообщений, помеченных как спам;
- U/US – число/размер в байтах сообщений, помеченных как безусловный спам;
- F/FS – число/размер в байтах сообщений, заблокированных фильтром;
- I/IS – число/размер в байтах сообщений, содержащих вирусы;
- DI – число инфицированных вложений;
- DM – число вложений, зараженных модификацией известного вируса;
- DSV – число вложений, зараженных неизвестным вирусом;
- DC – число излеченных вложений;
- DD – число вложений, которые были удалены;
- DSK – число вложений, которые по различным причинам были пропущены без антивирусной проверки;
- DAR – число вложений, которые были пропущены без антивирусной проверки из-за ограничений на архивы;
- DE – число вложений, при проверке которых произошла

ошибка;
- DTA – число вложений, содержащих программы для показа рекламы;
- DTD – число вложений, содержащих программы дозвона;
- DTJ – число вложений, содержащих программы-шутки;
- DTR – число вложений, содержащих потенциально опасные программы;
- DTH – число вложений, содержащих программы, предназначенные для получения несанкционированного доступа к компьютерным системам;
- WT – время в миллисекундах, потраченное плагином 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.

Использование drweb-inject для отправки писем

Программа 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 "не ноль" при ошибке.

Доступны следующие параметры командной строки:
- --help – вывод справки;
- --version – вывод версии программы;

- `--agent arg` – адрес **Агента** для получения конфигурации (или пустая строка для работы с параметрами по умолчанию - в этом случае обращения к **Агенту** происходить не будет);
- `--timeout arg` – максимальное время ожидания получения конфигурации от **Агента**;
- `--id arg` – уникальный идентификатор компонента **Sender**, которому надо отправить сообщение;
- `-f [ --env-from ] arg` – отправитель сообщения (для конверта);
- `-F [ --from ] arg` – если в отправляемом письме нет поля `From`, то при его установке `Full Name` будет взято из данного аргумента;
- `-i [ --ignore-dot ]` – не воспринимать строку с единственным символом - точкой (". "), как признак завершения ввода тела сообщения;
- `-t [ --extract-recipients ]` – надо ли добавлять к получателям в конверте всех получателей из поля `To`.

Если программа `drweb-inject` находится не в директории по умолчанию, то путь к ней можно указать с помощью параметра `--sendmail` при запуске `drweb-qcontrol`. Для более подробного информирования о происходящем можно использовать параметр `-v`.

В случае, когда отправитель не указан, используется имя пользователя, с правами которого выполняется программа. Если имени пользователя найти не удалось, то программа завершается с ненулевым кодом ошибки.

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 почтовыми системами или SMTP-прокси;
- для получения возможности дифференцировать настройки для каждого **Receiver**/**Sender** (что позволит, к примеру, использовать разные настройки для прослушиваемых интерфейсов);
- для получения возможности перенаправлять сообщения из одних MTA в другие - т.е. для маршрутизации.

Возможность запуска одновременно нескольких компонентов осуществляется следующим образом:

- каждому компоненту необходимо присвоить уникальный идентификатор (т.е. у каждого элемента из группы **Receiver**'ов и группы **Sender**'ов будет уникальный ID, при этом ID какого-либо **Receiver**'а может совпадать с ID какого-либо **Sender**'а);
- затем каждому компоненту необходимо сообщить, откуда ему получать свои настройки;
- потом каждому письму, посланному **Receiver**, присваивается в качестве тега уникальный идентификатор данного компонента;
- после обработки сообщения `drweb-maild` ищет доступный **Sender** с тем же идентификатором. Если он не найден, то письмо будет отправлено в **Sender** по умолчанию (это **Sender**, у которого уникальный идентификатор не задан - может быть только один), который должен быть всегда доступен.
- список доступных **Sender**'ов инициализируется при старте и обновляется при получении сигнала `SIGHUP`;
- проблема маршрутизации писем, сгенерированных в `drweb-notifier`, решается при помощи 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 **Receiver**'ов.

Уникальный идентификатор задается для **Receiver**/**Sender** через параметр командной строки `--unique-id`. При запуске с заданным параметром компоненты создают в директории `%`

`var_dir/msgs/{in|out}` набор поддиректорий для своей очереди писем, а в директории `%var_dir/ipc/` для **Sender** создается специальный UNIX сокет (название директории и UNIX сокета определяется на основе заданного уникального идентификатора).

Когда запускается второй экземпляр одного и того же компонента (например, второй экземпляр `drweb-receiver`), то необходимо выполнить дополнительную настройку: определить, как вторая копия будет получать конфигурацию. Для получения конфигурации можно использовать два способа: либо создавать новую копию `*.conf` файла, либо модифицировать существующий. Последний вариант более прост, но менее гибок.

Чтобы модифицировать существующий *.conf файл, необходимо:

- создать новый *.amc файл **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 соответствующих секций в конфигурационном файле.

Остальной список параметров надо скопировать из настройки оригинального компонента. Подробнее о синтаксисе `*.amc` файлов можно почитать в документации по **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 **Dr.Web Agent**, чтобы он прочитал новую информацию;
- запустить новый компонент, указав ему дополнительно следующие параметры командной строки:
    - `--unique-id id` – где `id` - уникальный идентификатор компонента;
    - `--component name` – где `name` - имя, под которым новый компонент известен **Dr.Web Agent** (`drweb-receiver2` из примера выше);
    - `--section` – новое название основной секции компонента (`Receiver2` из примера выше).

Пример:

```
%bin_path/drweb-receiver --unique-id id1 --component drweb-receiver2 --section Receiver2
%bin_path/drweb-sender --unique-id id1 --component drweb-sender2 --section Sender2
```

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 `*.conf` файла и настроить в нем параметры по своему усмотрению (при этом никаких секций переименовывать не требуется);
- создать новый `*.amc` файл, в который надо включить только информацию по новому компоненту(ам). Также в нем надо прописать путь к конфигурационному файлу, созданному на предыдущем шаге. В отличие от предыдущего способа, для нового компонента менять название основной секции не требуется;
- запустить/перезапустить **Dr.Web Agent**, чтобы он прочитал новую информацию;
- запустить новый компонент, указав ему дополнительно следующие параметры командной строки:
    - `--unique-id id` – где `id` - уникальный идентификатор компонента;
    - `--component name` – где `name` - имя под которым новый компонент известен **Dr.Web Agent** (`drweb-receiver2` из примера выше).

Пример:

```
%bin_path/drweb-receiver --unique-id id2 --component drweb-receiver2
%bin_path/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 **Dr.Web MailD** необходимо добавить строки по запуску новых компонентов. Подробнее о синтаксисе `*.mmc` файлов можно почитать в документации по **Dr.Web Monitor**.

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 типе переменных `Action` можно использовать необязательное действие `score(SCORE)` – где `SCORE` - целое число, которое будет добавлено к текущему счету письма;
- счет, выставляемый антиспам-плагином `vaderetro`, прибавляется к общему счету письма, и уже это итоговое значение потом сравнивается с порогами спама.
- с помощью ограничений `add_score` и `set_score` можно изменять значение счета (также это возможно с помощью параметров в некоторых ограничениях).
- с помощью Reputation IP Filter можно изменять значение счета всех сообщений в данной сессии.

Использовать значение счета письма можно следующим образом:

- в плагине `vaderetro` он сравнивается с пороговыми значениями для спама;
- в правилах можно в условиях использовать значение счета письма (префикс `score`). Подробнее об этом можно прочитать в главе Секция [Rules] данного руководства;
- в плагине `Modifier` счет также можно использовать в условиях и изменять его (подробнее см. в главе Плагин Dr.Web Modifier);

- 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.

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, либо предпринимать другие действия. Данная технология позволяет эффективно распознавать распространителей спама, а также бороться с DHA-атаками.

Модуль **Reputation IP Filter** включается, если задан хоть один фильтр в настройке **ReputationIPFilter**, либо если **MaxConcurrentConnection** установлен не в `0`. По умолчанию значение параметра **ReputationIPFilter** установлено в `score_filter`, и, соответственно, IP-фильтр включен и IP-адреса будут отфильтровываться на основании среднего значения счета, выставленного всем сообщениям и сессиям с этих IP-адресов.

Вся информация по IP-адресам хранится в оперативной памяти и периодически сбрасывается в файлы. Сохранение в файлы происходит либо при получении сигнала `SIGALRM` процессом `drweb-receiver` (он периодически сам себе шлет этот сигнал в соответствии с настройкой **StalledProcessingInterval**), либо при завершении

процесса `drweb-receiver`. Чтение файлов происходит только при запуске `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 директорию, указанную в параметре **`BaseDir`** секции [General] в файлы `ipv4.bin` (для IPv4 адресов) и `ipv6.bin` (для IPv6 адресов). Если при сохранении или чтении возникает ошибка, то это будет отражено в логах. Информация, сохраняемая в эти файлы, имеет бинарный формат и зависит от системы, на которой работает продукт, поэтому в общем случае данные файлы нельзя переносить для использования на другие системы.

Проверка IP в **Reputation IP Filter** происходит сразу после проверки **`SessionRestrictions`** в случае, если IP не помечен как `Trusted IP` (подробнее о **`*Restrictions`** и `Trusted IP` можно узнать в главе Секция [Receiver]).

Таким образом, если необходимо обезопасить некоторые IP от блокировки в **Reputation IP Filter**, то их необходимо пометить как `Trusted IP` в **`SessionRestrictions`**. Аналогично, если в **Reputation IP Filter** был по ошибке временно заблокирован IP-адрес, то его необходимо пометить как `Trusted IP` в **`SessionRestrictions`**, и тогда все следующие соединения с этого IP будут игнорироваться репутационным IP-фильтром.

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`.

Репутационный IP-фильтр проверяет IP-адрес сразу после прохождения им проверки **`SessionRestriction`**, если адрес

этот не отмечен как trusted (т.е., например, если в результате проверок **SessionRestriction** адрес отмечается как trusted, то он не будет проходить через Репутационный IP-фильтр).

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 переданных на проверку в drweb-maild сообщений, 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 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 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 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 (**`ProtectedEmails`**).

  Специфические параметры:

  - **`wrong_per_valid_rcpts`**=`D` – отношение ошибочных получателей письма (которые были отклонены после 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 переданным в drweb-maild сообщениям. 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 (сумма всех счетов сообщений, отправленных с данного IP, и счетов, выставленных сессиям (например, другими репутационными IP фильтрами или ограничениями)) к переданным в drweb-maild сообщениям. 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 переданным в drweb-maild сообщениям 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 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 к переданным в drwebmaild сообщениям 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 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 команды RCPT TO) к корректным получателям равно 0.02 (`wrong_per_valid_rcpts=0.02`);
- число ошибочных получателей письма (которые были отклонены после команды RCPT TO), переданных SMTP-клиентом, равно 20 (`min_wrong_rcpts=20`).

Плагины

На текущий момент реализованы следующие плагины программного комплекса **Dr.Web для почтовых серверов UNIX**: антивирусный плагин `drweb`, антиспам плагин `vaderetro`, плагин `headersfilter`, осуществляющий фильтрацию заголовков писем, и плагин `modifier`, позволяющий произвольно изменять части писем.

Антивирусный плагин drweb

`drweb` - плагин программного комплекса **Dr.Web для почтовых серверов UNIX**, осуществляющий антивирусную проверку почтовых сообщений.

Для работы плагина необходимы модуль `drwebd` (**Dr.Web Daemon**) и антивирусное ядро **"Доктор Веб"**, которые осуществляют непосредственную антивирусную проверку сообщений. Модуль `drwebd` и антивирусное ядро **"Доктор Веб"** входят в базовый пакет программного комплекса **Dr.Web для почтовых серверов UNIX**, который должен быть установлен до установки плагина `drweb`.

Сообщения передаются модулю `drwebd` на проверку уже разобранными на части, поэтому поддержка `MIME`-разбора в антивирусном ядре или модуле `drwebd` не требуется. Закончив анализ письма, плагин передает модулю `drweb-maild` результаты проверки и (при значении `Yes` параметра **`AddXHeaders`** конфигурационного файла плагина) может добавить в него следующие заголовки:

- `X-Antivirus: Name` - где `Name` - название и версия антивируса;
- `X-Antivirus-Code` - где `Code` - код завершения работы модуля `drwebd`.

Управление плагином drweb осуществляется с помощью конфигурационного файла plugin_drweb.conf.

Подключение плагина

Чтобы подключить плагин drweb к программному комплексу **Dr.Web для почтовых серверов UNIX** достаточно в конфигурационном файле **Dr.Web MailD** добавить строку drweb в список плагинов обрабатывающих письмо.

В том случае, если письмо должно обрабатываться плагином drweb до помещения в базу данных, плагин следует добавлять в список значений параметра **BeforeQueueFilters** секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
BeforeQueueFilters = drweb, vaderetro
```

Если же письмо должно попадать к плагину уже после помещения в базу данных, плагин добавляется в список значение параметра **AfterQueueFilters** секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
AfterQueueFilters = drweb
```

Настройка плагина

Все основные параметры работы плагина задаются 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 плагина drweb:

Секция [Antivirus]

Address = {адрес}	Сокет, через который антивирусный плагин взаимодействует с демоном drwebd. Допускается указание нескольких сокетов для взаимодействия с демонами 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 **Демона**, 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 демоном drwebd. Если значение параметра равно 0, время ожидания не ограничено. Значение по умолчанию: **Timeout** = 30s
HeuristicAnalysis = {Yes | No}	Эвристический анализатор позволяет демону drwebd обнаруживать неизвестные вирусы. При отключении эвристического анализатора 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 анализатора может привести к увеличению времени сканирования. Значение по умолчанию: **HeuristicAnalysis** = Yes
TCP_NODELAY = {Yes | No}	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 демона drwebd. 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 = {Yes | No}	Если указано значение Yes, к каждому проверенному демоном drwebd сообщению добавляются заголовки X-Antivirus и X-Antivirus-Code. Значение по умолчанию: **AddXHeaders** = Yes
Paranoid = {Yes | No}	Если у параметра указано значение Yes, все сообщения будут сканироваться в "параноидальном" режиме. В этом случае демон drwebd 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
RegexsForCheckedFilename = {список регулярных выражений}	Список регулярных выражений, используемых подключаемым модулем при проверке имен файлов в отчете, присылаемом демоном drwebd 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 производится только для файлов, в которых не найдено вирусов.
	Значение по умолчанию: **RegexsForCheckedFilename** =
LicenseLimit = {действие}	Действие, применяемое к сообщениям, которые не были проверены демоном drwebd по причине окончания срока действия лицензии. Обязательно должно быть задано одно из основных значений: pass, tempfail, discard, reject. Также может быть задано одно или несколько дополнительных значений: quarantine, redirect, notify, add-header, score.
	Значение по умолчанию: **LicenseLimit** = pass

Infected {действие}	=	Действие, совершаемое с сообщениями, зараженными известными вирусами. Обязательно должно быть задано одно из основных значений: `cure`, `remove`, `discard`, `reject`. Также может быть задано одно или несколько дополнительных значений: `quarantine`, `redirect`, `notify`. Значение по умолчанию: **Infected** = `cure, quarantine`
Suspicious {действие}	=	Действие, совершаемое с сообщениями, которые могут быть заражены неизвестным вирусом. Обязательно должно быть задано одно из основных значений: `pass`, `remove`, `discard`, `reject`. Также может быть задано одно или несколько дополнительных значений: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **Suspicious** = `reject, quarantine, notify`
Incurable {действие}	=	Действие, совершаемое с сообщениями, зараженными неизлечимо. Обязательно должно быть задано одно из основных значений: `remove`, `discard`, `reject`. Также может быть задано одно или несколько дополнительных значений: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **Incurable** = `reject, quarantine, notify`

Adware = {действие}	Действие, совершаемое с сообщениями, которые содержат программы для показа рекламы. Обязательно должно быть задано одно из основных значений: pass, remove, discard, reject. Также может быть задано одно или несколько дополнительных значений: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Adware** = reject, quarantine, notify
Dialers = {действие}	Действие, совершаемое с сообщениями, содержащими программы дозвона. Обязательно должно быть задано одно из основных значений: pass, remove, discard, reject. Также может быть задано одно или несколько дополнительных значений: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Dialers** = reject, quarantine, notify
Jokes = {действие}	Действие, совершаемое с сообщениями, содержащими программы-шутки. Обязательно должно быть задано одно из основных значений: pass, remove, discard, reject. Также может быть задано одно или несколько дополнительных значений: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Jokes** = reject, quarantine, notify

Riskware {действие}	=	Действие, совершаемое с сообщениями, содержащими потенциально опасные программы. Обязательно должно быть задано одно из основных значений: pass, remove, discard, reject. Также может быть задано одно или несколько дополнительных значений: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Riskware** = reject, quarantine, notify
Hacktools {действие}	=	Действие, совершаемое с сообщениями, содержащими программы, предназначенные для получения несанкционированного доступа к компьютерным системам. Обязательно должно быть задано одно из основных значений: pass, remove, discard, reject. Также может быть задано одно или несколько дополнительных значений: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **Hacktools** = reject, quarantine, notify
SkipObject {действие}	=	Действие, совершаемое с сообщениями, содержащими объекты, которые не могут быть проверены демоном drwebd. Возможны следующие причины: • Во вложении находятся защищённые паролем или запорченные архивы, символические ссылки, файлы нестандартных форматов.

	• Достигнуто максимальное время ожидания проверки сообщения. (Для получения более подробной информации обратитесь к описанию параметров **SocketTimeout** и **FileTimeout** в главном конфигурационном файле drweb32.ini). Обязательно должно быть задано одно из основных значений: pass, remove, discard, reject. Также может быть задано одно или несколько дополнительных значений: quarantine, redirect, notify, add-header, score. Значение по умолчанию: **SkipObject** = pass
ArchiveRestriction = {действие}	Действие, совершаемое с сообщениями, содержащими архивы, которые не могут быть проверены демоном drwebd. Возможны следующие причины: • Степень сжатия архивов превышает значение параметра MaxCompressionRatio в главном конфигурационном файле drweb32.ini. • Размер запакованных объектов превышает значение параметра **MaxFileSizeToExtract** в главном конфигурационном файле drweb32.ini. • Степень вложенности архивов превышает значение параметра **MaxArchiveLevel** в главном конфигурационном файле drweb32.ini.

	Обязательно должно быть задано одно из основных значений: `pass`, `remove`, `discard`, `reject`. Также может быть задано одно или несколько дополнительных значений: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **`ArchiveRestriction`** = `reject, quarantine, notify`
`ScanningErrors` = `{действие}`	Действие, совершаемое с сообщениями, вызывающими у демона `drwebd` ошибки в процессе проверки. Обязательно должно быть задано одно из основных значений: `pass`, `remove`, `discard`, `reject`, `tempfail`. Также может быть задано одно или несколько дополнительных значений: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **`ScanningErrors`** = `reject, quarantine`
`ProcessingErrors` = `{действие}`	Действие, совершаемое с сообщениями, вызывающими у антивирусного модуля ошибки в процессе проверки. Обязательно должно быть задано одно из основных значений: `pass`, `discard`, `reject`, `tempfail`. Также может быть задано одно или несколько дополнительных значений: `quarantine`, `redirect`, `notify`, `add-header`, `score`. Значение по умолчанию: **`ProcessingErrors`** = `reject`

BlockByFilename = {действие}	Действия, выполняющиеся в случае совпадения одного из регулярных выражений, указанных в настройках параметра **RegexsForCheckedFilename**, с именем файла из отчета, присылаемого демоном drwebd после сканирования сообщения. Обязательно должно быть задано одно из основных значений: pass, discard, reject, tempfail. Также может быть задано одно или несколько дополнительных значений: quarantine, redirect, notify, add-header, score. Обратите внимание, что в случаях, когда связь с **Демоном** 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 антивирусным плагином, SMTP-ответ **Dr.Web MailD** состоит из кода ошибки 550 5.7.0 и текстового сообщения, содержание которого может задаваться идущими далее параметрами. Значения параметров должны быть заключены в кавычки.

UseCustomReply {Yes | No}	=	Использование настраиваемых сообщений в SMTP-сессии для случаев, когда сообщения отклоняются. Значение по умолчанию: **UseCustomReply** = No
ReplyInfected {текст}	=	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}	=	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}	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}	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}	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}	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}	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"

Плагин headersfilter

headersfilter – плагин, осуществляющий фильтрацию писем по их заголовкам. При задании правил фильтрации можно использовать регулярные выражения (синтаксис Perl).

Подключение плагина

Чтобы подключить плагин headersfilter к программному комплексу **Dr.Web для почтовых серверов UNIX**, достаточно в конфигурационном файле **Dr.Web MailD** добавить строку headersfilter в список плагинов, обрабатывающих письмо.

Если письмо должно обрабатываться плагином headersfilter до помещения в базу данных, то название плагина следует добавлять в список значений параметра **BeforeQueueFilters** секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
BeforeQueueFilters = drweb, headersfilter
```

Если же письмо должно обрабатываться плагином уже после помещения в базу данных, то название плагина добавляется в список значений параметра **AfterQueueFilters** секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
AfterQueueFilters = headersfilter
```

Настройка плагина

Все основные параметры работы плагина задаются 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 плагина headersfilter.

Параметры фильтрации писем задаются с помощью правил, описанных ниже. Проверка правил осуществляется в порядке их задания, т.е. правило, заданное первым, будет проверено первым. Поиск соответствия правилу осуществляется до первого подходящего правила, после чего возвращается действие, заданное для этого правила.

Если почтовое сообщение попало под действие правила **Reject***, то дальнейшая проверка сообщения не производится. Если почтовое сообщение попало под действие правила **Accept***, то оставшиеся правила игнорируются и продолжается обработка сообщения остальными плагинами **Dr.Web MailD**.

Секция [Headersfilter]

ScanEncodedHeaders = { Yes | No}	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 закодированные заголовки будут просканированы дважды: до и после перекодирования. Значение по умолчанию: **ScanEncodedHeaders** = Yes
RejectCondition = { набор условий}	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. Правила могут быть заданы для любого из заголовков. Обычно каждое правило состоит из имени заголовка и регулярного выражения. HEADER = 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 - условие, позволяющее отфильтровать сообщения, у которых какой-либо из заголовков отсутствует. **Пример** : **RejectCondition** No From - отфильтровывает все письма с отсутствующим заголовком From. • HEADER = "8bit" - условие, позволяющее отфильтровать сообщения, у которых в заголовках содержатся 8-битные символы.
	Значение по умолчанию: **RejectCondition =**
AcceptCondition = {набор условий}	Правила принятия сообщений. Если какой-либо из заголовков письма попадает под действие того или иного из указанных правил, сканирование его заголовков прекращается, и письмо передается для дальнейшей обработки другим модулями. Правила могут быть заданы для любого из заголовков. Все сказанное про набор условий **RejectCondition** справедливо и для условий **AcceptCondition**.
	Значение по умолчанию: **AcceptCondition =**

FilterParts = {Yes | No}	Значение Yes разрешает обработку правил, заданных параметрами **RejectPartCondition** и **AcceptPartCondition**. Значение по умолчанию: **FilterParts** = Yes
RejectPartCondition = {набор условий} **AcceptPartCondition** = {набор условий}	Правила, аналогичные **RejectCondition** и **AcceptCondition**, но работающие с заголовками вложенных объектов. Также может быть использовано следующее правило: **FileName** = mask - где "mask" — регулярное выражение, соответствующее стандарту POSIX 1003.2. Обработка сообщений в соответствии с этими правилами возможна только в том случае, когда параметр **FilterParts** имеет значение Yes. Значение по умолчанию: **RejectPartCondition** = **AcceptPartCondition** =
MissingHeader = {текст}	Набор заголовков, отсутствие которых в письме становится условием его отфильтровывания. **Пример**: **MissingHeader** = "To", "From" Значение по умолчанию: **MissingHeader** =

Action = {действие}	Действие, совершаемое с отфильтрованными сообщениями. Обязательно должны быть указано одно из основных значений: pass, tempfail, discard, reject. Также может быть указано одно или несколько дополнительных значений: quarantine, redirect, notify, add-header, score. Обратите внимание, что при настройке данного параметра может быть одновременно задано несколько значений, через запятую.
	Значение по умолчанию: **Action** = reject, notify

В тех случаях, когда сообщение блокируется антивирусным плагином, SMTP-ответ **Dr.Web MailD** состоит из кода ошибки 550 5.7.0 и текстового сообщения, содержание которого может задаваться идущими далее параметрами. Значения параметров должны быть заключены в кавычки.

UseCustomReply = {Yes | No}	Использование настраиваемых сообщений в SMTP-сессии для случаев, когда сообщения отклоняются.
	Значение по умолчанию: **UseCustomReply** = No
ReplyRuleFilter = {текст}	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."

Антиспам плагин vaderetro

vaderetro - плагин программного комплекса **Dr.Web для почтовых серверов UNIX**, 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.

В зависимости от результатов анализа каждому письму, проверенному библиотекой **VadeRetro**, дается оценка - целое число в диапазоне от `-10000` до `+10000`. Чем меньше эта величина, тем больше вероятность, что письмо не является спамом. Пороговое значение оценки задается в параметре **SpamThreshold** конфигурационного файла плагина (если данная сообщению оценка равна значению параметра **SpamThreshold**, либо больше его, 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 о невозможности доставки. 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 о невозможности доставки письма.

  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 библиотекой **VadeRetro** с использованием счетчика **SpamThreshold** как спам или как вирус, плагин 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 вирусы с использованием счетчика **UnconditionalSpamThreshold**, может добавляться префикс заголовка из **UnconditionalSubjectPrefix**.

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 плагина

Чтобы подключить плагин `vaderetro` к программному комплексу **Dr.Web для почтовых серверов UNIX**, достаточно в конфигурационном файле **Dr.Web MailD** добавить строку `vaderetro` в список плагинов, обрабатывающих письмо.

При запуске, **Dr.Web MailD** временно переименовывает библиотеку libvaderetro.so в libvaderetro.so.cache для избежания конфликтов при обновлении.

Если письмо должно обрабатываться плагином `vaderetro` до помещения в базу данных, то название плагина следует добавлять в список значений параметра **BeforeQueueFilters** секции `[Filter]` конфигурационного файла **Dr.Web MailD**.

Пример:

```
BeforeQueueFilters = drweb, vaderetro
```

Если же письмо должно обрабатываться плагином уже после помещения в базу данных, то название плагина добавляется в список значений параметра **AfterQueueFilters** секции `[Filter]` конфигурационного файла **Dr.Web MailD**.

Пример:

AfterQueueFilters = vaderetro

Настройка плагина

Все основные параметры работы плагина задаются 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 плагина vaderetro:

Секция [VadeRetro]

FullCheck = {Yes | No}	При значении Yes данного параметра производится полная проверка сообщения на наличие спама. По результатам прохождения проверки сообщение получает оценку в виде целого числа из диапазона значений от -10000 до +10000. Чем меньше оценка, тем больше вероятность того, что письмо не является спамом. Пороговое значение (по достижении которого письмо однозначно классифицируется как спам) задается параметром **SpamThreshold** конфигурационного файла плагина. Обратите внимание, что при использовании этой возможности, общая скорость работы может уменьшиться.
	Значение по умолчанию: **FullCheck** = Yes
NoHamFrom = {Yes | No}	При значении Yes данного параметра не производится проверка писем, идущих на встроенные адреса **Dr.Web MailD** (ящики типа nospam@domain.ru).
	Значение по умолчанию: **NoHamFrom** = Yes

AddXHeaders = { Yes | No}	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 = { Yes | No}	Добавление к сообщению заголовка X-Drweb-SpamVersion, содержащего информацию о версии плагина **VadeRetro**. Значение по умолчанию: **AddVersionHeader** = No
AddXDrwebSpamStateNumHeader = { Yes | No}	Добавление к сообщению заголовка X-Drweb-SpamState-Num, в который входит числовое значение, присвоенное сообщению библиотекой **VadeRetro** по результатам классификации: • 0 - письмо не является спамом; • 1 - письмо является спамом; • 2 - письмо содержит вирус; • 3 - письмо является уведомлением о невозможности доставки сообщения. Значение по умолчанию: **AddXDrwebSpamStateNumHeader** = No

AddXSpamLevel = { Yes | No}	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 = { Yes | No}	Эвристическая проверка спамовых сообщений на наличие вирусов.
	Значение по умолчанию: **CheckForViruses** = Yes
CheckDelivery = { Yes | No}	Возможность отдельной фильтрации уведомлений о доставке сообщений.
	Значение по умолчанию: **CheckDelivery** = No
AllowRussian = { Yes | No}	Определяет, добавлять или нет дополнительные баллы к счету письма, если оно содержит кириллический текст.
	Значение по умолчанию: **AllowRussian** = Yes
AllowCJK = { Yes | No}	Определяет, добавлять или нет дополнительные баллы к счету письма, если оно содержит текст на китайском, японском или корейском языке.
	Значение по умолчанию: **AllowCJK** = Yes

WhiteList = {Lookups}	Белый список отправителей. Адреса отправителей получаются из поля From в теле письма. Если тело письма не будет содержать полей From, либо если перед полем From в теле письма будет стоять одна или несколько пустых строк - то, соответственно, поиск отправителя в белом списке производится не будет. Если в теле содержатся два поля From, то адрес будет взят из первого найденного поля. Также допустимо использование шаблонов. Например, чтобы добавить в белый список все адреса, принадлежащие конкретному домену, достаточно указать символ * вместо имени пользователя: *@mycompany.com. Если адрес из поля From был найден в белом списке, то от общего счета письма отнимается 5000 баллов. Обратите внимание, что поскольку белый список не сортируется, один и тот же адрес может быть случайно указан несколько раз. В таком случае 5000 баллов будут отниматься от общего счета письма столько раз, сколько раз адрес встречается в списке (указан 3 раза - отнимется 15000 баллов). **Пример**: hello@myneighbourhood.co.uk *@mycompany.com
	Значение по умолчанию: **WhiteList** =

BlackList = {Lookups}	Черный список отправителей. Адреса отправителей получаются из поля From в теле письма. Если тело письма не будет содержать полей From, либо если перед полем From в теле письма будет стоять одна или несколько пустых строк - то, соответственно, поиск отправителя в черном списке производится не будет. Если в теле содержатся два поля From, то адрес будет взят из первого найденного поля. Также допустимо использование шаблонов. Например, чтобы добавить в черный список все адреса, принадлежащие конкретному домену, достаточно указать символ * вместо имени пользователя: *@mycompany.com. Если адрес из поля From был найден в черном списке, то к общему счету письма прибавляется 5000 баллов. Обратите внимание, что поскольку черный список не сортируется, один и тот же адрес может быть случайно указан несколько раз. В таком случае 5000 баллов будут прибавляться к общему счету письма столько раз, сколько раз адрес встречается в списке (указан 3 раза - прибавится 15000 баллов).
	Значение по умолчанию: **BlackList** =
SubjectPrefix = {текст}	Префикс, добавляемый к теме сообщения, если оно отмечено как спам. Он добавляется в случае, когда оценка, полученная письмом, больше, чем значение параметра **SpamThreshold** (что позволяет однозначно классифицировать сообщение как спам).
	Значение по умолчанию:

	SubjectPrefix =
UnconditionalSubjectPrefix = {текст}	Префикс, добавляемый к теме сообщения, если оно отмечено как безусловный спам. Он добавляется в случае, когда оценка, полученная письмом, больше, чем значение параметра **UnconditionalSpamThreshold**. Значение по умолчанию: **UnconditionalSubjectPrefix** =
NotifySubjectPrefix = {текст}	Префикс, добавляемый к теме сообщения, если оно является уведомлением о невозможности доставки (и, соответственно, определено в 3 класс писем библиотекой **VadeRetro**). Значение по умолчанию: **NotifySubjectPrefix** =
PathToVadeRetro = {путь к файлу}	Путь к антиспам библиотеке **VadeRetro**. Возможно ее динамическое обновление с помощью модуля 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 плагина. Значение по умолчанию: **PathToVadeRetro** = %var_dir/lib/libvaderetro.so

UnconditionalSpamThreshold = {численное значение}	Если оценка, полученная письмом, равна значению данного параметра или превышает его, письмо считается безусловным спамом. В этом случае к письму применяется действие, заданное параметром **UnconditionalAction**. Значение параметра **UnconditionalSpamThreshold** должно быть равным значению параметра **SpamThreshold** или превышать его. Значение по умолчанию: **UnconditionalSpamThreshold** = 1000
SpamThreshold = {численное значение}	Если оценка, полученная письмом, равна значению данного параметра или превышает его, письмо считается спамом. В этом случае к письму применяется действие, заданное параметром Action. Данная проверка происходит только в том случае, если оценка письма меньше значения, заданного параметром **UnconditionalSpamThreshold**. Значение параметра **SpamThreshold** должно быть равным значению параметра **UnconditionalSpamThreshold** или быть меньше его. Значение по умолчанию: **SpamThreshold** = 100
UnconditionalAction = {действие}	Действие, совершаемое с безусловным спамом. Обязательно должны быть указано одно из основных значений: pass, reject, discard, tempfail. Также может быть указано одно или несколько дополнительных значений: quarantine, redirect, add-header, score.

	Значение по умолчанию: **UnconditionalAction** = pass
Action = {действие}	Действие, совершаемое со спамом. Обязательно должны быть указано одно из основных значений: pass, reject, discard, tempfail. Также может быть указано одно или несколько дополнительных значений: quarantine, redirect, add-header, score. Значение по умолчанию: **Action** = pass
NotifyAction = {действие}	Действие, совершаемое с письмом, если оно является уведомлением о невозможности доставки (и, соответственно, определено в 3 класс писем библиотекой **VadeRetro**). Обязательно должны быть указано одно из основных значений: pass, reject, discard, tempfail. Также может быть указано одно или несколько дополнительных значений: quarantine, redirect, add-header, score. Значение по умолчанию: **NotifyAction** = pass
UseCustomReply = {Yes | No}	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 выполняется действие **Action**, **UnconditionalAction**, **NotifyAction** = 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: **SpamCustomReply** = "Dr.Web vaderetro plugin: this is spam!"
FromProtectedNetworkScoreAdd = {численное значение}	Если письмо идет из **ProtectedNetworks**, к его текущему счету добавляется указанное значение (может быть отрицательным). Для отключения этой функции можно указать значение 0.
	Значение по умолчанию: **FromProtectedNetworkScoreAdd** =
UseReplyCache = {Yes | No}	Управляет работой параметров **ProtectedNetworkReplyCacheLifeTime** и **ReplyToProtectedNetworkScoreAdd**. Если параметры отключены, то reply_cache не используется.
	Значение по умолчанию: **UseReplyCache** =

ProtectedNetworkReplyCacheLifeTime = {время}	Если письмо идет из **ProtectedNetworks**, адреса всех получателей этого письма добавляются в reply_cache на время, заданное данным параметром (если адрес уже был в reply_cache, запись для него обновляется). Соответственно, для ответных писем, отправители которых находятся в reply_cache, можно корректировать счет письма с помощью параметра **ReplyToProtectedNetworkScoreAdd**.
	Значение по умолчанию: **ProtectedNetworkReplyCacheLifeTime** =
ReplyToProtectedNetworkScoreAdd = {численное значение}	Значение, добавляемое к счету писем, отправители которых находятся в reply_cache.
	Значение по умолчанию: **ReplyToProtectedNetworkScoreAdd** =

Плагин Dr.Web Modifier

Плагин **Dr.Web Modifier** используется для:

- контентного анализа – поиска в телах обрабатываемых писем объектов с определенными MIME-типами (графика, исполняемые файлы, медиа-файлы), а также MIME-объектов, удовлетворяющих определенным условиям и т. п;
- модификации тел писем – удаления MIME-объектов, удовлетворяющих определенным условиям, модификации заголовков выбранных MIME-объектов и их содержимого.
- блокировки, помещения в карантин, перенаправления, добавление заголовков и счета в зависимости от

найденных в телах обрабатываемых писем MIME-объектов.

Dr.Web Modifier поддерживает следующие версии регулярных выражений: базовые (basic regular expressions), расширенные (extended regular expressions) и Perl-совместимые (Perl-compatible regular expressions) регулярные выражения.

В настройках конфигурационного файла плагина **Dr.Web Modifier** могут быть заданы правила обработки писем. Правила подразделяются на четыре категории.

Первая категория правил воздействует на все письмо целиком:

- `pass`, `accept` - пропустить письмо. В случае глобальных правил после получения любой из этих команд дальнейшая обработка письма не ведется. В случае локальных правил после получения команды `accept` **Dr.Web Modifier** переходит к обработке письма с помощью глобальных правил;
- `reject` - отклонить письмо;
- `discard` - отклонить письмо, не уведомляя отправителя;
- `notify` - оповестить администратора, обработка письма не прекращается. После этой команды необходимо указать имя шаблона отчёта, который будет использован при оповещении, иначе при обработке письма будут возникать ошибки. Шаблоны лежат в директории, указанной в значении параметра **`TemplatesBaseDir`** конфигурационного файла **Dr.Web MailD**.

  Пример :

  `GlobalRules` `= select message, notify rule`

  Нужный префикс `admin_` и расширение `.msg` будут автоматически подставлены компонентом **Dr.Web Notifier**;

- `tempfail` - уведомить отправителя о сбое сервера;
- `redirect` - перенаправить письмо на заданный адрес;
- `quarantine` - отправить письмо в карантин.

Команда `stop` прекращает обработку правил. 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 в части локальных правил. Для глобальных правил, accept равноценна pass.

Команды reject discard и tempfail являются "решающими" – после них обработка письма прекращается, вне зависимости от того, указаны ли там в правиле еще какие-либо команды.

Все вышеперечисленные команды требуют дополнительного текстового поля для вставки текста уведомления.

Пример :

```
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.

Пример :

```
GlobalRules = select mime.headers Subject "word1|word2|wordN", if found, notify rule, quarantine, reject, endif
```

В данном случае, при наличии в заголовке письма слов "word1", "word2", либо "wordN" письмо копируется в карантин, после чего администратору передаётся уведомление и письмо будет отклоняется.

Пример :

```
GlobalRules = select mime.headers Subject "word1|word2|wordN", if found, tempfail, endif, select mime.headers Subject "word1|word2|
```

```
wordN", if found, pass, endif
```

В приведенном примере при наличии в заголовке письма слов "word1", "word2", либо "wordN" письмо будет отклонено, а отправителю отправлено сообщение о сбое сервера. Следующая за действием `tempfail` часть правила обрабатываться не будет.

Остальные категории правил применяются для индивидуальной обработки различных элементов письма, поскольку каждое письмо можно представить как иерархический набор элементов, в который могут входить `MIME`-объекты, их заголовки и содержимое, вложенные `MIME`-объекты для многокомпонентных сообщений. Над выбранными элементами можно производить действия, такие как удаление, добавление подписи, замена или модификация текста и т.п.

Каждая команда должна предваряться типом операции над выборкой: `select`, `or`, `and`, `nand`, `nor`.

После каждой команды должны быть указаны параметры выборки.

Доступны следующие команды:

- `select message`

  Данная команда выбирает корневой MIME-элемент письма.

- `select mime(headers), select mime.headers`

  `select mime(prologue), select mime.prologue`

  `select mime(body), select mime.body`

  `select mime(epilogue), select mime.epilogue`

  Данные команды выбирают `MIME`-объекты. Различие между командами со скобками и командами с точкой заключается в том, что команды со скобками выбирают `MIME`-объекты, содержащие указанный элемент, а команды с точкой - сам элемент.

Пример:

Команда, удаляющая все видео-фрагменты из письма:

```
select mime( headers) Content-type "x-video"
remove
```

Команда, удаляющая информацию о типе данных из всех видео-фрагментов:

```
select mime. headers Content-type "x-video"
remove
```

Нельзя выбрать составной MIME-объект, кроме как когда он является самим письмом.

- `select mime( headers) тип_заголовка регулярное_выражение`
  `select mime( prologue) регулярное_выражение`
  `select mime( body) регулярное_выражение`
  `select mime( epilogue) регулярное_выражение`

  Данные команды выбирают элементы, содержащие текст, удовлетворяющий шаблону.

- `select sender регулярное_выражение`
  `select recipient регулярное_выражение`

  Данные команды выбирают записи об отправителе и получателях. Данные об отправителе и получателях берутся из конверта письма. В случае нахождения искомой последовательности символов эти команды действуют эквивалентно `select message`.

Пример:

Если письмо предназначено для администратора, можно добавить в конец письма приветствие с помощью следующих команд:

```
select recipient "root@localhost",
append_text "hello, root"
```

Иногда оказывается необходимым выбрать элементы, соответствующие нескольким критериям. Для этого перед каждым следующим правилом для выборки элементов нужно поставить одно из зарезервированных слов:

- `and` - оставить в выборке только те элементы, которые попадут под указанное правило.
- `nand` - оставить в выборке только те элементы, которые НЕ попадут под указанное правило.
- `or` - добавить в выборку те элементы, которые попадут под указанное правило.
- `nor` - добавить в выборку те элементы, которые НЕ попадут под указанное правило.

Обратите внимание, что эти операторы работает только с выборками, содержащими `MIME`-объекты, в составе которых находятся те или иные элементы (и, соответственно, не работает с выборками, содержащими сами элементы).

Пример:

Нужно выбрать фрагменты, написанные на `html` и содержащие слово "`<script`":

```
select mime(headers) Content-type html
and mime(body) "\<script"
```

Это два разных правила, применяющихся последовательно. Первое выбирает все элементы, содержащие в заголовке `Content-type` слово "`html`", а второе оставляет в списке выбранных элементов 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 фрагменты, содержащие в заголовке `Content-type` слово "`html`". В соответствии со вторым критерием из этой выборки

будут исключены все фрагменты, 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 фрагменты, содержащие в заголовке Content-type слово "html". В соответствии со вторым критерием к этой выборке будут добавлены также все фрагменты, содержащие последовательность символов "<script" в любом регистре.

Пример:

```
select mime( headers) Content-type html
nor mime( body) "\<script"
```

В соответствии с первым критерием будут выбраны все фрагменты, содержащие в заголовке Content-type слово "html". В соответствии со вторым критерием к этой выборке будут добавлены также все фрагменты, НЕ содержащие последовательность символов "<script" в любом регистре.

Если перед последующим правилом указано select, то предыдущая выборка будет очищена.

Пример:

```
select mime( headers) Content-type html
select mime( body) "\<script"
```

Полученная в итоге выборка будет осуществлена только в соответствии со вторым критерием - т.е. будут выбраны только фрагменты, содержащие последовательность символов "<script" в любом регистре.

Если ни логических операторов, ни команды select перед последующим правилом не указано, то оно 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, содержащие в заголовке `Content-type` слово "`html`".

Для заголовков в целях совместимости с подключаемым модулем **Vaderetro** также можно использовать команды сравнения `>n` и `<n`. Сравниваемый заголовок считается попадающим под правило, если он содержит целое число (например: `X-Drweb-SpamScore "30"`) и удовлетворяет какому-либо правилу, например:

```
select mime( headers) X-Drweb-SpamScore "<50"
```

В данном случае обратный слэш перед знаком "<" не нужен. Если бы правило было задано как:

```
select mime( headers) X-Drweb-SpamScore "\<50"
```

то в результате были бы выбраны элементы с заголовком `X-Drweb-SpamScore "<50"`

Команда `select_mimes` позволяет перейти от выборки заголовков к выборке `MIME`-объектов, их содержащих. Это позволяет ускорить работу плагина 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.

Следующая категория правил применяется для преобразования выбранных элементов.

Эти правила действуют только на содержимое `MIME`-объектов, если не указано иное.

- `replace выражение_для замены заменяемое_регулярное_выражение`
  `replace_all новый_текст`

  Данные команды заменяют один текст другим.

  Пример:

  Поиск и переименование исполняемых файлов во вложениях:

  ```
  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 подобъектами команды:

  ```
  select message
  replace_all "text"
  ```

  не произведут 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

  "`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 "old: ${self} new: ${lc}"
```

Заголовок `Subject` письма, соответствующий указанному паттерну, например: "`This is Subj`" - будет заменен на: "`old: This is Subj new: this is subj`".

Пример:

```
select mime.body ".*", replace
"Upper: ${uc}" "http://\\S+"
```

В теле письма текст, соответствующий указанному паттерну, например: "`Text1 http://vasya.pup.kin Text2`" - будет заменен на: "`Text1 Upper: HTTP://VASYA.PUP.KIN Text2`".

Пример:

```
select mime.body ".*", replace "http://
check-url.com?url=${urlencode}" "http://\
\S+"
```

В теле письма текст, соответствующий указанному паттерну, 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.

Пример:

Например, нельзя использовать команду remove в правилах

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 фрагмент в формате `plain-text` или `html` в выбранные `MIME`-объекты.

  Пример:

  ```
  select message
  append_html "<h1>checked by antispam</h1>" [[7b:]encoding]
  ```

  Добавление подписи к письму, где необязательный параметр `encoding` - название кодировки добавляемого текста, а префикс "`7b:` " указывает на использование 7-битной `context transfer` кодировки.

  Источником данных для вставки может также служить `lng`-файл, используемый для задания сообщений в конкретной кодировке. Для использования строк из `lng`-файла, следует использовать формат записи `$1, $2 ... $n`, где `n` — номер строки в `lng`-файле.

  Пример:

  Пусть в lng-файле есть строки:

  ```
  1 = строка1
  2 = другая строка
  ...
  ```

тогда выражение `append_text $2` будет эквивалентно выражению `append_text "другая строка"`.

Также, возможно использование механизма `lookups` через тип значений `LookupsLite`, в котором можно указывать только либо непосредственное значение, либо `lookups` типа `file`.

Пример:

```
append_text "lookup: file: path_to_file"
```

Добавление заголовков осуществляется с помощью следующих команд:

```
select message, addheader "foo: bar"
```

Это набор команд добавляет к выбранному элементу письма заголовок с именем `foo` и значением `bar`. Имя и значение заголовка отделяются друг от друга двоеточием (": ").

Последняя категория правил служит для создания ветвлений типа if/else:

- `goto` - безусловный переход;
- `goto( y)` - условный переход если был выбран хоть один элемент;
- `goto( n)` - условный переход если не был выбран ни один элемент.

Аргументом служит положительное целое, указывающее, сколько команд следует пропустить.

Пример:

```
mime( header) Content-type "executable"
goto( n) 1
reject
```

Этот набор команд позволяет отклонить письма, к которым приложены выполняемые файлы. Он соответсвуют следующему

псевдокоду:

```
selection=find( mimes with content type
"*executable*")
if( selection){
  reject mail;
}
```

Также можно использовать команды `if [not] found .... else .... endif`.

Пример:

```
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 плагином **Vaderetro** как спам.

Обратите внимание, что при использовании в правилах кавычек может потребоваться экранировать их несколькими символами "\". В текущей версии программы для экранирования кавычки требуется шесть символов "\".

Пример:

```
GlobalRules = select mime.headers Subject ".*\\
\\\\"", if found, reject, endif
```

Также существует возможность проверки оценки каждого письма. Письму присваивается некая оценка (score), сперва равная нулю. При обработке письма плагины могут менять эту оценку. При помощи команд `if score`, `add_score` и `set_score` можно произвести проверку и изменение этой оценки. 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`.

Обратите внимание, что ввиду обработки конфигурационного файла несколькими синтаксическими анализаторами, для экранирования символа "\", его необходимо повторить 7 раз.

Пример:

```
GlobalRules = select mime.headers "Subject" "^\
\\\\\\\$", if found, reject, endif
```

Отклоняет письма с темой, состоящей из одного символа "\".

Пример:

```
....
if found,\
  set_score 10,\
endif,\
```

Устанавливает счет 10 для письма, удовлетворяющего некоему условию.

Пример:

```
....
add_score 11,\
```

Увеличивает счет письма на 11.

Пример:

```
....
if score >100,\
  reject,\
else,\
  add_score -5,\
endif
```

Если счет письма превышает 100, то письмо отклоняется. В противном случае его счет уменьшается на 5.

Аргумент `if score` должен быть записан одной строкой, без пробелов - т.е. `<100` но не `< 100` - и должен состоять из символа операции сравнения и целого аргумента.

Для `if score` возможны следующие операции сравнения:

- `if score <2` - если `score` меньше 2
- `if score >5` - если `score` больше 5
- `if score =8` - если `score` равно 8

Аргумент `if score` может быть 32-битным целым числом из диапазона от -2млрд до +2млрд. 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` при задании правил (т.е., к примеру, не задавать 2000000000 для параметра `add_score`).

Добавление к `MIME`-объектам текстовой информации ведет к сбросу списка выбранных объектов.

	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.header	+	+	+	-	-	-	-	-	-	-	-	-	-	-	-	-	-
mime.prologue	+	+	+	-	-	-	-	-	-	-	-	-	-	-	-	-	-
mime.epilogue	+	+	+	-	-	-	-	-	-	-	-	-	-	-	-	-	-
mime.body	+	+	+	-	-	-	-	-	-	-	-	-	-	-	-	-	-
mime(headers)	+	*	*	-	-	-	-	+	-	-	-	-	-	-	-	-	-
mime(prologue)	+	*	*	-	-	-	-	+	-	-	-	-	-	-	-	-	-
mime(epilogue)	+	*	*	-	-	-	-	+	-	-	-	-	-	-	-	-	-
mime(body)	+	*	*	-	-	-	-	+	-	-	-	-	-	-	-	-	-
sender	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
recipient	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
message	-	-	-	-	-	-	-	-	+	+	+	+	+	+	+	+	+

Таблица 1. Воздействие команд на различные типы объектов.

- ***** такое же, как для mime.body;
- **+** применимо;
- **–** игнорируется.

Примеры:

- выбрать элементы по двум признакам:

```
GlobalRules = select mime( headers) Content-type
"text", and mime( body) "typical spam",\
```

- если такие элементы имеются, удалить все письмо:

```
goto( n) 1,\
discard,\
```

- иначе найти все исполняемые файлы и удалить:

```
select mime( headers) Content-disposition ".
exe",\
remove,\
```

- добавить в конец тела письма подпись:

```
select message, append_text "проверено!"
```

Пожалуйста, обратите внимание, что после `goto(n) 1,\` и `discard,\` не должно быть пробелов! Иначе правило не будет работать.

Добавление в письмо файла в формате html:

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

Удаление писем от выбранных пользователей:

```
GlobalRules = select mime(headers) From
"weirdohacker@server.net", if found, reject,
endif
```

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 следующее правило:

- выбор сообщений по указанным признакам:

```
GlobalRules = select mime.headers Subject
"Help",\
```

```
if found,\
select mime.headers To "someaddress@my-net.
com",\
if found,\
```

- перенаправление найденных сообщений на указанный адрес:

```
redirect "anotheraddress@my-net.com",\
```

- удаление исходных сообщений, чтобы они не были доставлены оригинальным получателям:

```
discard,\
endif,\
stop,\
endif,\
```

Перенаправление писем, приходящих на общий корпоративный ящик, в зависимости от их темы:

```
GlobalRules = \
```

- проверка писем в техподдержку:

```
select mime.headers Subject "техподдержка|
проблем[ аы]|помощь",\
if found,\
```

- следующие команды будут пропущены, если шаблон не был найден:

```
select mime.headers To "@company.com",\
if found,\
redirect "support@company.com",\
endif,\
pass, \
endif,\
```

- сюда перейдет управление, если клиент хочет купить товар:

```
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,\
```

- все прочие темы:

```
select mime.headers To "@company.com",\
redirect "inbox@company.com",\
pass
```

Поиск и переименование исполняемых файлов во вложениях:

```
select mime.headers Content-disposition "filename=.*\\.exe", or mime.headers Content-type "name=.*\\.exe",\
replace "\\.ex_" "\\.exe",\
pass
```

Подключение плагина

Чтобы подключить плагин **Dr.Web Modifier** к программному комплексу **Dr.Web для почтовых серверов UNIX**, достаточно в конфигурационном файле **Dr.Web MailD** добавить строку `modifier` в список плагинов, обрабатывающих письмо. Если письмо должно обрабатываться плагином `modifier` до помещения в базу данных, то название плагина следует добавлять в список значений параметра

BeforeQueueFilters секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
BeforeQueueFilters = modifier
```

Если же письмо должно обрабатываться плагином уже после помещения в базу данных, то название плагина добавляется в список значений параметра **AfterQueueFilters** секции [Filter] конфигурационного файла **Dr.Web MailD**.

Пример:

```
AfterQueueFilters = modifier
```

Настройка плагина

Все основные параметры работы плагина задаются 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 плагина modifier:

Секция [Modifier]

GlobalRules = {список правил, перечисленных через запятую}	Настройки пула потоков. Список общих правил для обработки писем. Правила подразделяются на 4 категории: • первая категория правил воздействует на все письмо целиком. Возможные значения: pass, reject, discard, notify, tempfail. • вторая категория правил выбирает для индивидуальной обработки различные элементы письма.

		• третья категория правил включает в себя команды преобразования выбранных элементов. • четвертая категория правил служит для создания ветвлений типа `if/else`. **Пример**: Данное правило добавляет в письмо файл в формате html: **GlobalRules** = `select message, append_html "lookup: file: /maild-files/somehtml. html"` Это правило удаляет письма от выбранных пользователей: **GlobalRules** = `select mime (headers) From "weirdohacker@server. net", if found, reject, endif`
		Значение по умолчанию: **GlobalRules** =
Encoding `{кодировка}`	=	Кодировка, которую плагин будет указывать для текста, вставляемого командами `append_text` и `prepend_text` непосредственно из правил.
		Значение по умолчанию: **Encoding** = `koi8-r`
UseCustomReply `{Yes | No}`	=	Отправлять в качестве SMTP-ответа сообщение, заданное в параметре **ReplyRuleFilter**, если входящее сообщение отклонено плагином `modifier`.
		Значение по умолчанию:

	UseCustomReply =
ReplyRuleFilter = {текст}	Настраиваемое сообщение в SMTP сессии для случаев, когда сообщение отвергается плагином modifier. Значение по умолчанию: **ReplyRuleFilter** =

Интеграция с почтовыми системами

В данной главе рассматриваются особенности интеграции программного комплекса с различными почтовыми системами. 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 используемой MTA. 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 с использованием специального транспорта;
- Для Postfix реализуется схема подключения AfterQueue;
- Zmailer настраивается для использования в режиме контекстного фильтра на этапе SMTP-соединения.

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-proxy

Тот факт, что **Dr.Web MailD** может работать как прокси-сервер для почтовых протоколов, позволяет использовать его совместно с большинством почтовых систем. В этом режиме в роли SMTP/LMTP-сервера выступает модуль `drweb-receiver`, а в роли SMTP/LMTP-клиента - `drweb-sender`. Кроме того, модуль `drweb-sender` имеет возможность передавать письма непосредственно локальной почтовой системе.

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` (RFC2920)
- `8BITMIME` (RFC1652)
- `ENHANCEDSTATUSCODES` (RFC3463)
- `SIZE` (RFC1870)
- `AUTH` (RFC4954)

При работе напрямую с Internet полезны технологии, встроенные в `drweb-receiver` и позволяющие фильтровать письма непосредственно на этапе SMTP-сессии: это ограничения (Restrictions) и Reputation IP Filter.

Все параметры настройки модулей `drweb-receiver` и `drweb-sender` сосредоточены в секциях `[Receiver]` и `[Sender]` конфигурационного файла **Dr.Web MailD** и описаны в главах Receiver и Sender данного руководства.

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:

- соединиться с CGP через программу удаленного администрирования `WebAdmin`;
- перейти в меню `Settings -> General -> Helpers`;
- добавить новый `content-filter` с параметрами:

  `Use Filter`: `DrWeb Maild`

  `Log`: `Problems`

  `Path`: `%bin_dir/drweb-cgp-receiver`

  `Time-Out`: `2 minutes`

  `Auto-Restart`: `15 seconds`
- проверить, достаточен ли уровень привилегий, с которыми исполняется CGP, для запуска `drweb-cgp-receiver`;
- перейти в меню `Settings -> Queue -> Rules`;
- создать новое правило типа "`проверять сообщения размером менее N байт`".

Чтобы создать новое правило, необходимо:

- выбрать имя правила (например, `drweb-filter`) и нажать кнопку **Create New**;
- нажать кнопку **Edit** и установить для поля **Action** значение `External Filter`;
- в поле **Parameters** внести значение поля **Filter** из меню `Settings -> General -> Helpers`.

Чтобы избежать многократной проверки писем, пришедших из GROUP, LIST или RULES (http://www.communigate.com/CommuniGatePro/Transfer.html), вы можете добавить к правилу также настройку:

```
"Submit Address", "not in", "GROUP*,LIST*, RULES*"
```

Обратите внимание, что при загрузке письма через PIPE теряется флаг authenticated. Следовательно, если есть плагины, подключенные в очереди **AfterQueueFilters**, то имеет смысл также добавить к правилу следующую строку:

```
Any Recipient not in alldomains@main.domain, all@*
```

где main.domain - это 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 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 директорию 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.

Так как drweb-cgp-sender загружает новые письма в CGP через драйвер PIPE, то, во избежание зацикливания писем, необходимо добавлять в письма специальный заголовок, который задается параметрами **UseSecureHash** и **SecureHash** группы настроек [CgpSender] конфигурационного файла **Dr.Web MailD**.

В таком случае модуль drweb-cgp-receiver будет пропускать письма с этим заголовком без проверки (предварительно удалив его значение). Можно также отключить использование такого заголовка, задав для параметра **UseSecureHash** секции настроек [CgpSender] конфигурационного файла **Dr.Web MailD** значение No. В таком случае модуль drweb-cgp-receiver будет пропускать без проверки все письма, поступившие от драйвера PIPE.

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 процессы:

- drweb-notifier
- drweb-cgp-sender
- drweb-maild

Принцип работы

Dr.Web MailD работает с почтовой системой CGP следующим образом:

- Письмо приходит в CGP.
- После проверки своих настроек, CGP при необходимости отправляет сообщение на проверку в helper, в роли которого выступает компонент drweb-cgp-receiver.
- При получения письма, компонент drweb-cgp-receiver ищет 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`;
- `drweb-maild` применяет к письму плагины, 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.
- Письмо передается в **Sender** и, после добавления заголовка `SecureHash` (при значении параметра `UseSecureHash = yes`), перемещается в директорию 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.

- После проверки директории `/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 директории `%bin_dir/lib/`. Необходимо скопировать библиотеки (или сделать на них линки) `libstdc++.so.6` и `libgcc_s.so.1` из `%bin_dir/lib/` в системную директорию с библиотеками.

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.

Также 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.

MailD полностью совместим с Sendmail 8.12.3 и выше. При работе с более ранними версиями могут возникать проблемы (см. раздел Возможные 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`. Sendmail и модуль `drweb-milter` могут быть запущены на разных компьютерах, в то время как модули `drweb-milter` и `drweb-maild` должны быть запущены на одном компьютере.

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:

```
-------------------- cut ------------------
############################
# Input mail filters
############################
O InputMailFilters=drweb-filter
O Milter.LogLevel=6
############################
#   Xfilters
############################
Xdrweb-filter, S=__ADDRESS__,
F=T, T=C:1m; S:5m; R:5m; E:1h
------------------ cut ------------------
```

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:

```
-------------------- cut ------------------
# privacy flags
O PrivacyOptions=goaway,noetrn,nobodyreturn
-------------------- cut ------------------
```

Или в `{sendmail_src}/cf/cf/feature/msp.m4`:

```
-------------------- cut ------------------
define(`confPRIVACY_FLAGS'
`goaway,noetrn,nobodyreturn,restrictqrun')
-------------------- cut ------------------
```

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:

```
------------------- cut -------------------
INPUT_MAIL_FILTER(`drweb-filter',
`S=__ADDRESS__,
```

```
F=T, T=C:1m;S:5m;R:5m;E:1h')
define(`confMILTER_LOG_LEVEL',`6')
------------------- cut -------------------
```

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.

Для UNIX сокетов адрес задается в формате:

```
local:__SOCKPATH__
```

где `__SOCKPATH__` должен указывать путь, доступный с теми правами, с которыми будет запущен фильтр.

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:

```
------------------- cut -------------------
O Milter.macros.envfrom=i,{auth_type}, ...
------------------- cut -------------------
```

(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:

```
------------------- cut -------------------
O Milter.macros.connect=_,{if_addr}, ...
------------------- cut -------------------
```

(многоточием обозначены остальные параметры, значение которых не важно).

Для подавления вывода в syslog сообщений вида:

```
------------------- cut -------------------
X-Authentication-Warning:      some.domain.com:
drweb set sender to DrWeb-DAEMON@some.domain.
com using -f
------------------- cut -------------------
```

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:

```
------------------- cut -------------------
####################
#  Trusted users    #
```

```
####################
Tdrweb
------------------- cut -------------------
```

Либо добавив в файл `submit.mc` строку:

```
------------------- cut -------------------
define(`confTRUSTED_USERS', `drweb')
------------------- cut -------------------
```

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 Sender и Milter соответственно.

При работе **Dr.Web MailD** с почтовой системой Sendmail в системе должны быть запущены следующие процессы:

- `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 - 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-filter): 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-filter): select(read): timeout before data write"
```

```
"... Milter (drweb-filter): 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-filter, 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:

SMTP/LMTP-сервер `drweb-receiver` (компонент **Receiver**) получает новое письмо от SMTP-модуля системы Postfix, после чего передает его модулю `drweb-maild` для анализа. По результатам анализа это письмо либо высылается почтовой системе (возможно, модифицированным), либо блокируется (в таком случае почтовой системе могут отсылаться дополнительные отчеты). Пересылка писем почтовой системе Postfix осуществляется через SMTP/LMTP-клиент `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`. Postfix и модуль `drweb-milter` могут быть запущены на разных компьютерах, в то время как модули `drweb-milter` и `drweb-maild` должны быть запущены на одном компьютере.

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 настроек [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 следующий: `unix:pathname` - где `pathname` - абсолютный путь к UNIX сокету.

В том 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 режиме **BeforeQueueFilters**. 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 для вывода информации о нем в лог drweb-milter и информацию об авторизации.

Настройка Dr.Web MailD

Если программный комплекс запускается посредством **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 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 программе `sendmail` из пакета Postfix.

Также следует 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 должно быть установлено значение `Yes` для параметра **UseSecureHash** в этой же секции.

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 Receiver, Sender и Milter соответственно.

При работе **Dr.Web MailD** с почтовой системой Postfix в системе должны быть запущены следующие процессы:

- `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 **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.

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:

```
------------------ cut ----------------------
##############################################
```

```
#           MAIN CONFIGURATION SETTINGS       #
###############################################

trusted_users = drweb
------------------ cut ----------------------
```

Также следует заметить, что в случае, если Exim производит доставку почты сразу после ее получения от `drweb-sender`, и в этой доставке случаются значительные задержки (например, она происходит по SMTP-протоколу), возможно истечение времени ожидания, заданного значением параметра **`PipeTimeout`** секции `[Sender]` конфигурационного файла **Dr.Web MailD**, так как Exim не возвращает код успешного получения в `drweb-sender` до окончания длительного процесса доставки. Чтобы избежать этой проблемы, можно настроить Exim отправлять все полученные от **Dr.Web MailD** письма сначала в очередь, и только потом производить 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:

```
------------------ cut ----------------------
#################################################
#        ROUTERS CONFIGURATION                  #
# Specifies how remote addresses are handled #
#################################################
#          ORDER DOES MATTER                    #
#   A remote address is passed to each in       #
#       turn until it is accepted.              #
#################################################
------------------ cut ----------------------
```

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:

```
------------------ cut ----------------------
################################################
#      TRANSPORTS CONFIGURATION                #
################################################
#         ORDER DOES NOT MATTER                #
#  Only one appropriate transport is called    #
#           for each delivery.                 #
################################################
------------------ cut ----------------------
```

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.

Затем следует перезапустить **Dr.Web MailD**, а вслед за ним и почтовую систему 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.

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 директорию `exim*/Local/`.
- Добавьте в `Makefile` системы Exim, расположенный в директории `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 сделать на них линки) `libstdc++.so.6` и `libgcc_s.so.1` из `%bin_dir/lib/` в системную директорию с библиотеками.
- Можно выполнить в консоли

  ```
  $ 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 директории исполняемых файлов Exim).

Ниже приведено описание дополнительных параметров для конфигурационного файла Exim:

DrwebTimeout = {время в секундах}	Период, в течение которого SendMail 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** = 60 s
DrwebBaseDir = {Yes | No}	Базовая директория MailD, в которой хранятся сокеты, база данных и т.д.

	Значение по умолчанию: **DrwebBaseDir** = %var_dir/
DrwebProcessingError = { pass | discard | reject | tempfail}	Действия для писем сообщениям, вызвавших ошибки сканирования (например, если антивирусному плагину не хватает памяти, либо он не может подключиться к drweb-maild). Для **DrwebProcessingError** если ничего не задано в конфиге, или по ошибке задано 2 значения (например, discard и pass) - будет браться по дефолту tempfail Значение по умолчанию: **DrwebProcessingError** = tempfail
DrwebLogLevel = { Quiet | Error | Alert | Info | Debug }	Уровень подробности ведения файла отчета. Значение по умолчанию: **DrwebLogLevel** = Debug
DrwebIpcLevel = { Quiet| Error | Alert | Info | Debug }	Устанавливает уровень подробности протокола работы библиотеки IPC. Значение по умолчанию: **DrwebLogLevel** = Debug
DrwebSyslogFacility = { Daemon | Mail | Local0 .. Local7 }	Тип подсистемы, через которую системный сервис syslogd, ведущий протоколирование, выдает сообщения о событиях. Значение по умолчанию: **DrwebSyslogFacility** = Daemon
DrwebMaxSize = { Daemon | Mail |	Максимальный размер проверяемого сообщения. При значении 0 ограничения на размер отсутствуют.

Local0 .. Local7 }	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 модуль **Receiver** встраивается в саму систему Exim, нет необходимости в запуске модуля drweb-receiver. Если запуск программного комплекса происходит через **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 Receiver и Sender соответственно.

При работе **Dr.Web MailD** с почтовой системой Exim в системе должны быть запущены следующие процессы:

- drweb-notifier
- drweb-sender
- drweb-maild
- drweb-receiver

Известные проблемы

Если при запуске почтовая система Exim выдает ошибку вида:

```
transport drweb_transport: cannot find transport driver "lmtp"
```

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 директориях. Список таких директорий 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.

Настройка Qmail

Для подключения системы **Dr.Web MailD** к почтовой системе Qmail необходимо осуществить следующую последовательность действий:

- Сохраните оригинальный `qmail-queue` и запомните путь к директории, в которую вы его сохранили, так как он понадобится нам далее.
- Перепишите `qmail-queue` из директории `%bin_dir` в директорию `qmail/bin/`. После этого не забудьте правильно выставить права и владельцев как для нового `qmail-queue` (являющегося фильтром **Dr.Web MailD**), так и для скопированного `qmail-queue.original`.

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 `qmail-queue`:

```
-rws--x--x  X drweb   qmail    SIZE DATE qmail-queue
-rws--x--x  X qmailq  qmail    SIZE DATE qmail-queue.original
```

Это можно сделать, выполнив следующие команды:

```
$ chown drweb:qmail qmail-queue
$ chmod 4711 qmail-queue
$ chown qmailq:qmail qmail-queue.original
$ chmod 4711 qmail-queue.original
```

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 Sender и Qmail соответственно.

При работе **Dr.Web MailD** с почтовой системой Qmail в системе должны быть запущены следующие процессы:

- drweb-notifier
- drweb-sender
- drweb-maild
- drweb-qmail

Известные проблемы

Описание:

При запуске Qmail выдает одну из следующих ошибок:

1. terminate called after throwing an instance of 'St9bad_alloc'
   what(): St9bad_alloc
2. bash: xmalloc: cannot allocate 2 bytes (0 bytes allocated)
3. qmail-queue.real: error while loading shared libraries: libc.so.6: failed to map segment from shared object: Cannot allocate memory
4. /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 SMTP-протоколу, 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** программного комплекса **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 SMTP-протоколу, Qmail после получения тела сообщения выводит в консоль строку 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 директории `%bin_dir/lib/`. Необходимо скопировать библиотеки (или сделать на них линки) `libstdc++.so.6` и `libgcc_s.so.1` из `%bin_dir/lib/` в системную директорию с библиотеками.

Интеграция с почтовой системой Courier

Настройка Courier

Для подключения **Dr.Web MailD** к почтовой системе Courier необходимо проделать следующую последовательность действий:

1. Установить права на модуль `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 директорий и поддиректорий 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 директорию фильтров Courier (обычно это `/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/lib/courier/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 **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 Sender и Courier соответственно.

При работе **Dr.Web MailD** с почтовой системой Courier в системе должны быть запущены следующие процессы:

- drweb-notifier
- drweb-sender
- drweb-maild
- drweb-courier

Интеграция с почтовой системой ZMailer

Модуль drweb-zmailer совместим с указанной почтовой системой только начиная с версии ZMailer 2.99.55.

Dr.Web MailD можно использовать вместе со ZMailer в двух режимах:

- В режиме контекстного 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 контекстного 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** программного 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 директорию:
  `$(ZMAILER_SRCHOME)/smtpserver`

  где `ZMAILER_SRCHOME` - путь к директории исполняемых файлов 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 процессы:

- `drweb-notifier`
- `drweb-sender`
- `drweb-maild`

Режим контекстного фильтра на этапе SMTP-соединения

Чтобы включить поддержку **Dr.Web Maild** в ZMailer необходимо:

- скопировать (или создать символическую ссылку) файл `drweb-zmailer.sh` в директорию `$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 `drweb-zmailer.sh`.

Режим контекстного 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 SMTP envelope отправителя (так обычно рассылаются сообщения об ошибках либо о запрете доставки писем, также рассылка таких писем 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 системы **Dr.Web MailD** - Zmailer все настройки `drweb-zmailer` указываются в командной строке (их можно задать, например, в скрипте `drweb-zmailer.sh`).

В командной строке модуля `drweb-zmailer` могут быть заданы следующие параметры:

- `-h [ --help ]` – вывод справки и выход;

- `-v [ --version ]` – вывод версии и выход;
- `-u [ --user ] arg (=drweb)` – пользователь, с правами которого запущен модуль `drweb-maild`.

  Поскольку ZMailer запускает модуль `drweb-zmailer` с правами `root`, то требуется либо весь комплекс **Dr.Web MailD** запускать с правами `root` (и установить пустую строку в качестве значения для параметра `-u ""`), либо установить для данного параметра необходимое значение;

- `-l [ --level ] arg (=info)` – уровень подробности протокола работы модуля. Возможные значения: `Quiet`, `Error`, `Alert`, `Info`, `Debug`;
- `-i [ --ipclevel ] arg (=info)` – уровень подробности протокола взаимодействия с модулем `drweb-maild`. Возможные значения: `Quiet`, `Error`, `Alert`, `Info`, `Debug`;
- `-f [ --facility ] arg (=mail)` – тип подсистемы, через которую системный сервис `syslogd`, ведущий протоколирование работы модуля, выдает сообщения о событиях (более подробную информацию вы можете получить в документации по `syslogd`). Возможные значения: `Daemon`, `Mail`, `Local0-7`;
- `-b [ --basedir ] arg (=%var_dir)` – базовая директория **Dr.Web MailD**. Аналог параметра **`BaseDir`** секции `[General]` конфигурационного файла **Dr.Web MailD**;
- `-t [ --timeout ] arg (=30)` – время ожидания обработки одного письма;
- `--file arg` – путь к файлу для проверки. Должен устанавливаться только в режиме работы контекстного фильтра на этапе маршрутизации;
- `--hash arg` – значение из параметра **`SecureHash`** в секции `[Sender]` конфигурационного файла **Dr.Web MailD**.

  Должен устанавливаться только в режиме работы контекстного фильтра на этапе маршрутизации;

- `--interface arg (=1)` – версия `smtpserver`. Необходимо задавать только в режиме работы контекстного фильтра на этапе SMTP-сессии: значение `0` - для версии 2.99.55 и ниже; значение `1` - для версии 2.99.56 и выше;
- `-e [ --error-action ] arg (=reject)` – действие, выполняемое над сообщением при возникновении внутренней ошибки фильтра.

  Возможные значения: `pass`, `reject`, `discard`, `tempfail`.

Использование прокси

Использование прокси, который входит в состав **Dr.Web для почтовых серверов UNIX**, позволяет достичь нескольких целей:

1. Модули обработки почтового трафика (**Receiver** и **Sender** для входящего и исходящего трафика соответственно) и модуль проверки почты (`drweb-maild`) распределяются по разным хостам, и между этими хостами с помощью прокси 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 компоненты `drweb-maild` в настоящий момент не поддерживают кластерную реализацию и не обмениваются внутренней информацией друг с другом (статистикой, карантином, настройками в базе данных и т.п.) - в результате, для каждого из `M` компонентов `drweb-maild` будет своя статистика, свой карантин, свои настройки.

Прокси состоит из двух компонентов: `drweb-proxy-client`

и drweb-proxy-server.

- drweb-proxy-client – работает на компьютере, где запущены **Receiver** и **Sender**, и запускается вместо модуля drweb-maild. Остальные компоненты воспринимают drweb-proxy-client как drweb-maild, ни имея никакого представления о существовании прокси.
- drweb-proxy-server – работает на компьютере, где запущен основной модуль drweb-maild, и выполняет роль компонентов **Receiver** и **Sender**.

Оба этих компонента взаимодействуют друг с другом, обеспечивая передачу оригинальных сообщений и их модификаций с целью обработки их другими компонентами **Dr.Web для почтовых серверов UNIX**, расположенными на разных хостах.

Компоненты drweb-notifier, drweb-monitor и drweb-agent работают на каждом из хостов.

Общая схема работы с использованием прокси выглядит следующим образом:

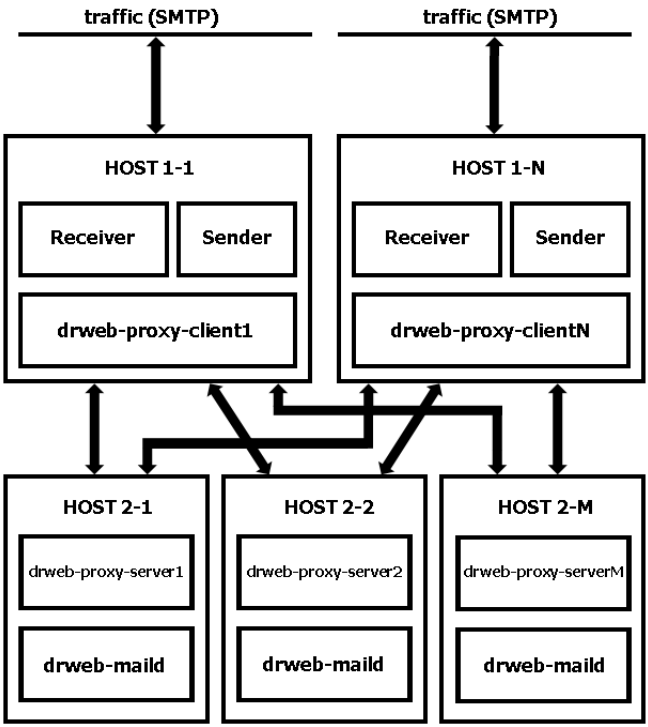


Рис. 16. Схема работы с использованием прокси

Из данной схемы видно, что как `drweb-proxy-client`, так и `drweb-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.

Параметр **ProxyServersAddresses** из секции [ProxyClient] задает адреса HOST2-* (см. схему), на которых слушают компоненты drweb-proxy-server*. Параметр **ProxyClientsAddresses** из секции [ProxyServer] задает адреса HOST1-* (см. схему), на которых слушают компоненты drweb-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. Если вес 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 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. Если не удалось передать сообщение на адрес с наибольшим весом, то выбирается следующий по весу адрес (его вес должен быть больше или равен 1). (Если доступных адресов с весом ≥ 1 больше не осталось, то переходим к пункту 3.)
2. Предпринимается попытка отправить сообщение по выбранному адресу. Если происходит ошибка, то адрес отмечается как недоступный, и осуществляется возврат к

пункту 1.

3. Если все адреса с весами, большими `0`, оказались недоступны для передачи, то пытаемся послать сообщение на запасной адрес (запасные адреса проверяются в порядке их задания в списке). Если и запасные адреса недоступны, то возвращается ошибка.

Выбор веса следует осуществлять на основе имеющихся ресурсов на каждом из узлов.

Когда письма отправляются на проверку модулю `drweb-maild` 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), и письма, сгенерированные самим `drweb-maild` (отчеты, уведомления) - независимо от того, в какой из очередей (**BeforeQueue** или **AfterQueue**) находятся плагины.

При отправке почты клиентам из списка **ProxyClientsAddresses** будут учитываться настройки из Правил (**SenderAddress**).

Обратите внимание, что при использовании прокси вместе с почтовыми системами Milter, Qmail, Courier (и модулем `drweb-milter`, соответственно) лучше воздержаться от помещения плагинов в **AfterQueue**. В настоящий момент прокси не поддерживает backdoor-соединения с **Receiver**'ом. Соответственно, если ответ не идет сразу в **Receiver** (например, когда плагин помещается в **AfterQueue**), то `drweb-milter` завершает 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 HOST1-1 (т.е. на том хосте, через который будет проходить проверяемый трафик и где будет располагаться компонент drweb-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 HOST1-1 и проверить, что почта корректно обрабатывается.
3. Установить **Dr.Web для почтовых серверов UNIX** на HOST2-1 (т.е. на хост, где будет осуществляться фактическая проверка почты и располагаться компонент drweb-proxy-server). При установке можно не настраивать компоненты **Receiver** и **Sender** - они тут не понадобятся.
4. Настроить конфигурацию HOST2-1 аналогично HOST1-1.
5. На HOST2-1 в mmc файле (из каталога %etc_dir/monitor) **Dr.Web для почтовых серверов UNIX** необходимо закомментировать запуск компонентов **Receiver** и **Sender** и раскомментировать запуск компонента drweb-proxy-server.
6. На HOST2-1 в конфигурационном файле **Dr.Web для почтовых серверов UNIX** надо задать значение параметра **ProxyClientsAddresse** из секции [ProxyServer], указав в нем IP-адрес HOST1-1, на который будет отправляться почта (тот же, что и в

значении параметра **Address** секции [ProxyClient]).

7. Проверить корректность настройки на хосте HOST2-1 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 HOST2-1 полностью настроен и готов к работе.
8. На HOST1-1 в конфигурационном файле **Dr.Web для почтовых серверов UNIX** надо задать значение параметра **ProxyServersAddresses** из секции [ProxyClient], указав в нем IP-адрес HOST2-1, на который будут отправляться запросы на проверку сообщений (тот же, что и в значении параметра **Address** секции [ProxyServer]).
9. На HOST1-1 в mmc файле **Dr.Web для почтовых серверов UNIX** необходимо закомментировать запуск drweb-maild. Там же надо раскомментировать запуск компонента drweb-proxy-client.

   Пожалуйста, обратите внимание, что при попытке одновременно запустить на одном хосте компоненты drweb-proxy-client и drweb-maild **Dr.Web Monitor** завершит свою работу и никакие компоненты не будут загружены. Информация об ошибке будет выведена в лог.
10. Проверить корректность настройки на хосте HOST1-1 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 HOST2-1.
11. Опционально на HOST1-1 теперь можно отключить **Dr.Web Daemon** и обновление через **Dr.Web Updater** (если

на данном хосте нет других продуктов **"Доктор Веб"**), так как они больше там не нужны.

Масштабирование для случаев, когда `M` и/или `N` больше `1`, тривиально: 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.

Консоль Dr.Web для почтовых серверов UNIX

Настройка программного комплекса **Dr.Web для почтовых серверов UNIX** может быть осуществлена через специально разработанный веб-интерфейс **Dr.Web консоль 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 веб-интерфейса **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;
- `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 директорий.
- `Encode::CN` – модуль для работы с китайской кодировкой.
- `Encode::HanExtra` – модуль с дополнительным набором китайских кодировок.

Недостающие модули рекомендуется устанавливать из командной строки. 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.

Для установки модулей Perl вы можете использовать следующие команды (в зависимости от установленной операционной системы).

Для Debian/Ubuntu:

```
apt-get install libperl libjson-perl libjson-xs-perl libxml-parser-perl libxml-xpath-perl libtimedate-perl libmime-tools-perl
```

Для установки требуются права `root`.

Для CentOS 5:

```
yum install perl-JSON perl-JSON-XS perl-XML-Parser perl-XML-XPath perl-TimeDate perl-MIME-tools
```

Для установки требуются права `root`.

Установка через CPAN:

```
cpan JSON JSON::XS XML::Parser XML::XPath Date::Parse MIME::Words MIME::Entity MIME::Parser MIME::Head
```

Для установки требуются права `root`.

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, веб-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 директории `%bin_dir/web/`).

Подключение модуля, а также настройка дополнительных параметров самого Webmin осуществляется через его веб-интерфейс.

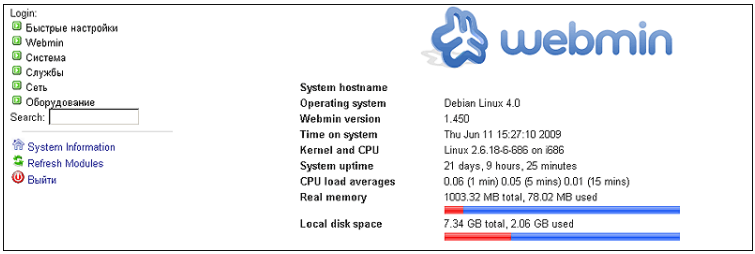


Рис. 17. Главная страница Webmin

Установка дополнительных модулей происходит в разделе **Настройка Webmin** секции **Webmin** основного меню, в подразделе **Модули Webmin**.

Рис. 18. 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 директорий вашей системы, в котором вы сможете выбрать соответствующий установочный пакет (`%bin_dir/web/drweb-maild-web.wbm.gz` по умолчанию).

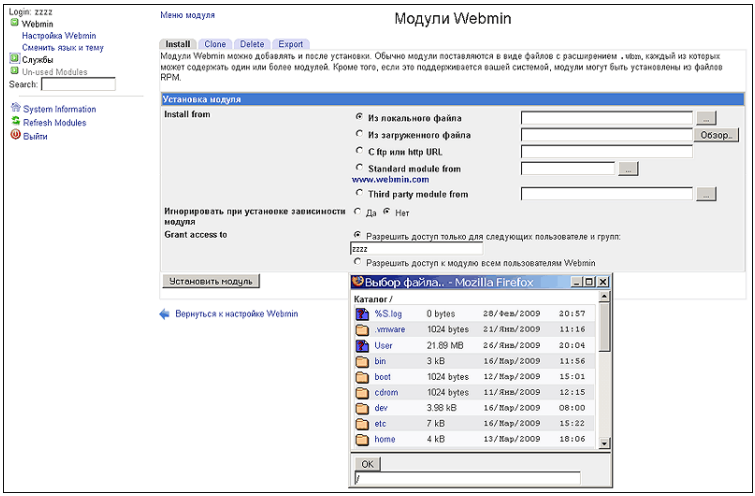


Рис. 19. Добавление модулей Webmin

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 иконку или название директории, последняя 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**.

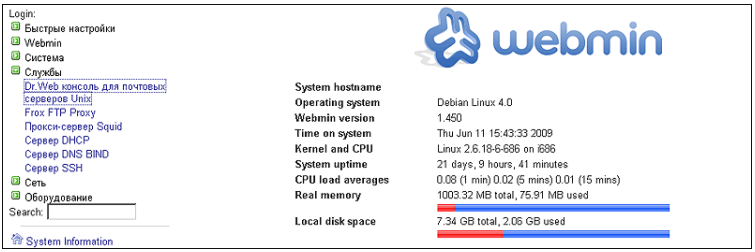


Рис. 20. Новый пункт меню "Dr.Web консоль для почтовых серверов Unix"

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.

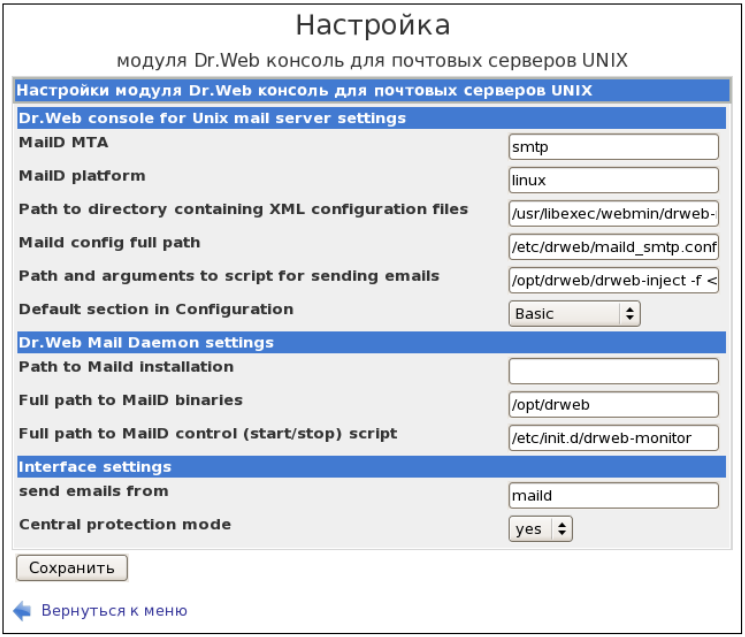


Рис. 21. Настройка модуля

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.

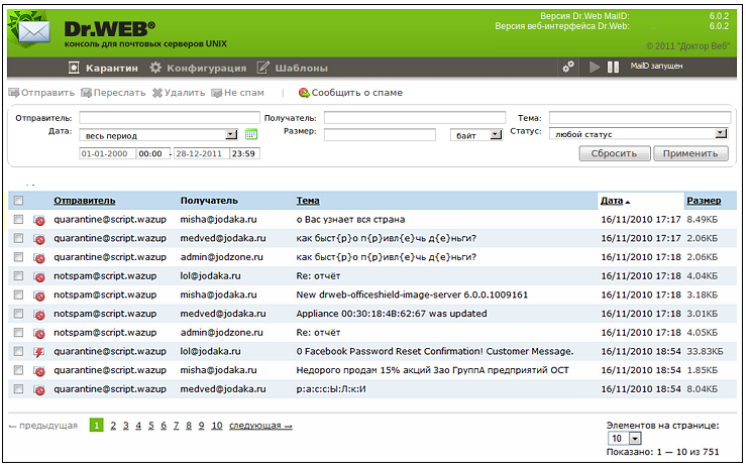


Рис. 22. Dr.Web консоль для почтовых серверов Unix

Справа от заголовка модуля вы найдете информацию о текущей версии **Dr.Web MailD** и веб-интерфейса **Dr.Web**.

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 центральной защиты кнопка **Остановить Dr.Web MailD** остановит также все прочие локальные сервисы **Dr.Web**, запущенные в режиме центральной защиты.

Если **Dr.Web MailD** работает в режиме центральной защиты, то после изменении прав доступа к настройкам в **Центре Управления Dr.Web ESS**, необходимо вручную перезагрузить страницу с веб-интерфейсом, чтобы изменения вступили в силу.

Карантин

В карантин перемещаются письма, отфильтрованные каким-либо из антивирусных и антиспам-плагинов **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.

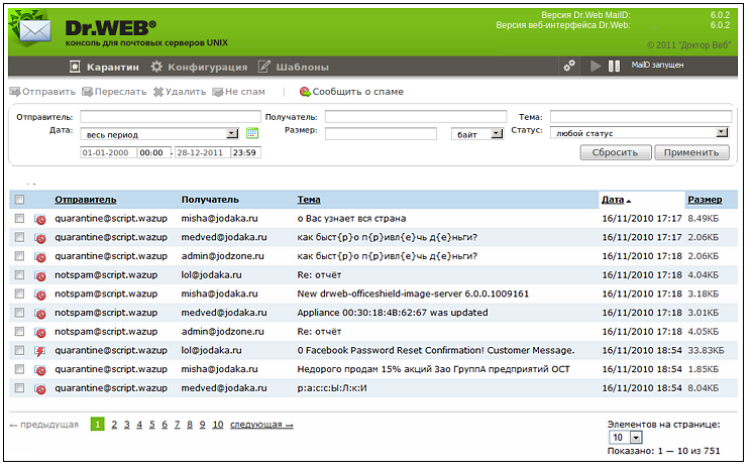


Рис. 23. Вкладка "Карантин"

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.

Панель инструментов

Элементы панели инструментов (за исключением кнопки **Пожаловаться на спам**) становятся активны только тогда, когда выделяется какое-либо из содержащихся в карантине писем.

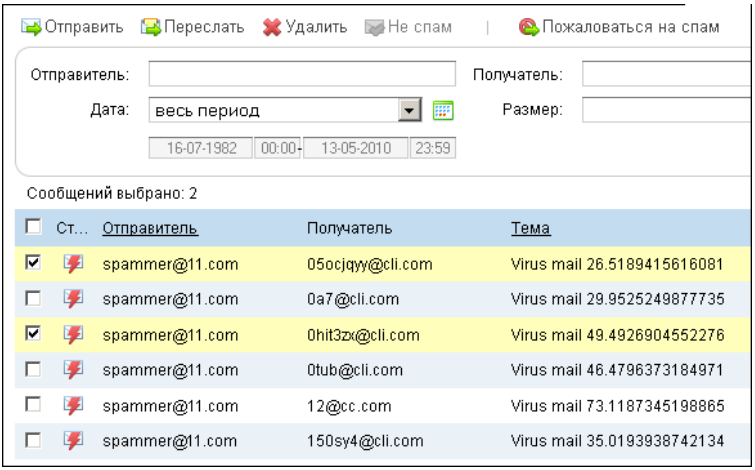


Рис. 24. Панель инструментов

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.
- убрать статус "спам" у писем, которые были помещены антиспам-плагином в карантин по ошибке. Для этого нужно выбрать письма со статусом "спам" из списка и нажать на кнопку **Не спам**. После этого соответствующие письма будут автоматически отправлены получателям и удалены из карантина.
- пожаловаться на спам.

Данная функция не предназначена для работы с элементами списка. Письмо, являющееся с точки зрения пользователя спамом, но не помещенное в карантин, должно быть предварительно сохранено в файловой системе.

После нажатия на кнопку **Пожаловаться на спам** откроется дополнительное окно с предложением выбрать файл, в котором содержится спам-письмо, и отправить его на проверку в соответствующий отдел компании **"Доктор Веб"**.

Панель фильтров

Панель фильтров предназначена для удобства при обработке писем, помещенных в карантин.

Рис. 25. 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.

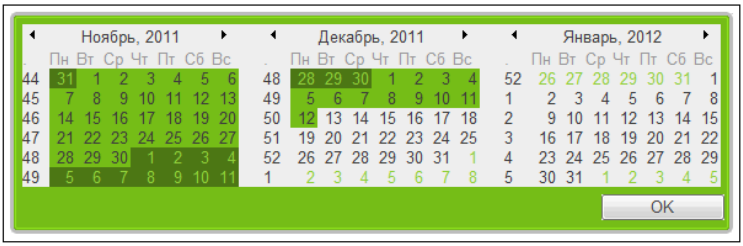


Рис. 26. Календарь

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 **OK**. Окно календаря закроется, и в соответствующих полях ввода появятся выбранные значения.

Можно также указать точное время или интервал времени для поиска писем.

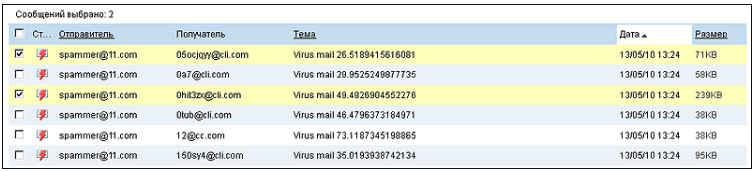
При указании интервала времени будут выбраны те письма, время помещения в карантин которых попадает в указанный интервал, включая его границы. Таким образом, если в качестве начала и конца интервала указать один и тот же момент времени, то будут выбраны только те письма, которые пришли строго в указанное 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 корреспонденции;
  - ✓ **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	0hll3zx@cli.com	Virus mail 49.4928904552276	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. 27. 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 корреспонденции, - 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.

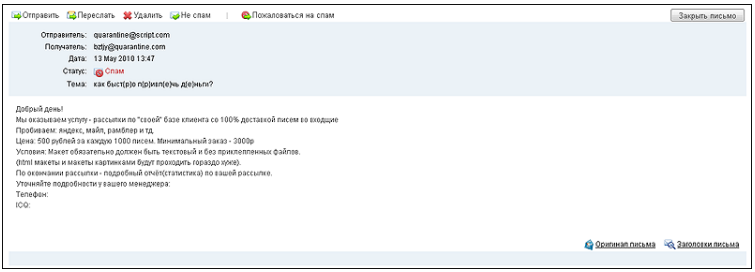


Рис. 28. Письмо

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. 29. 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 соответствующую ячейку. Если вы хотите внести дополнительные изменения, вы можете вернуться к предыдущей странице, нажав на кнопку **Продолжить Редактирование**. Нажмите **Сохранить** чтобы сохранить изменения или **Сохранить и применить** чтобы немедленно применить изменения.

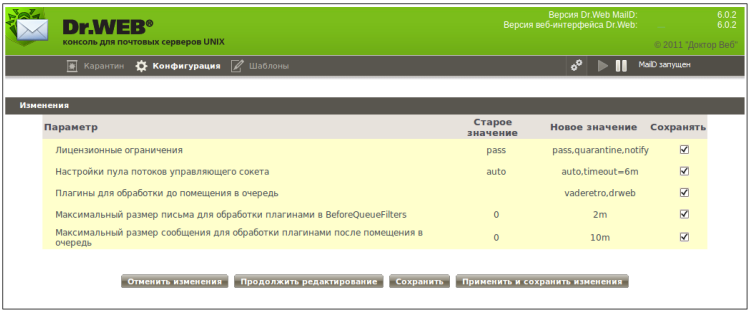


Рис. 30. Страница предпросмотра

Вкладка "Базовые настройки"

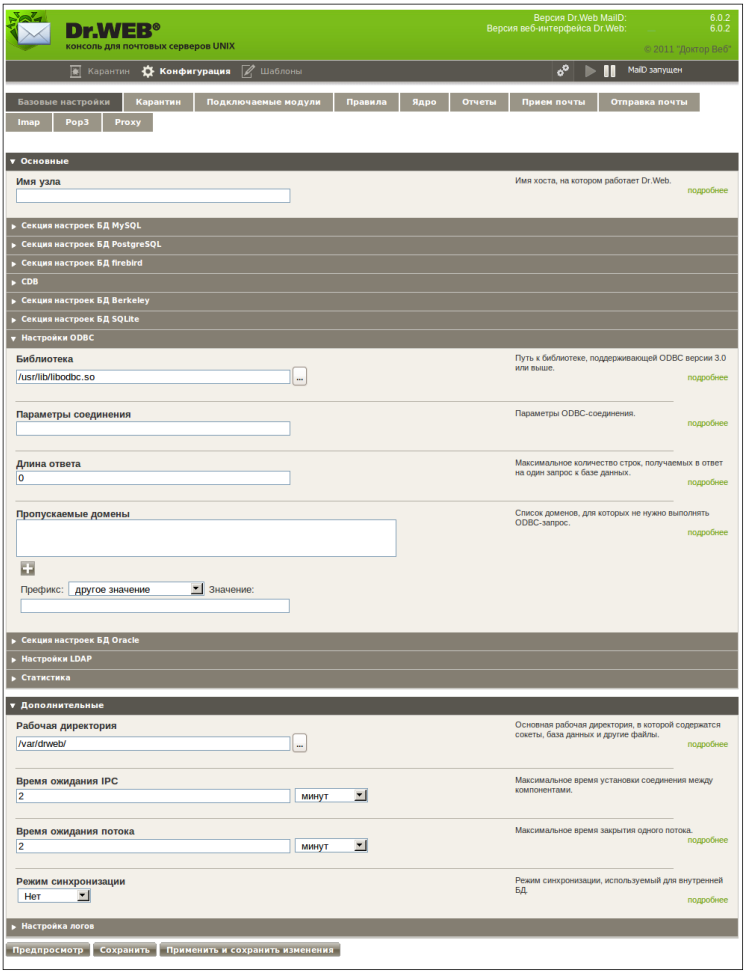


Рис. 31. Базовые настройки

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"

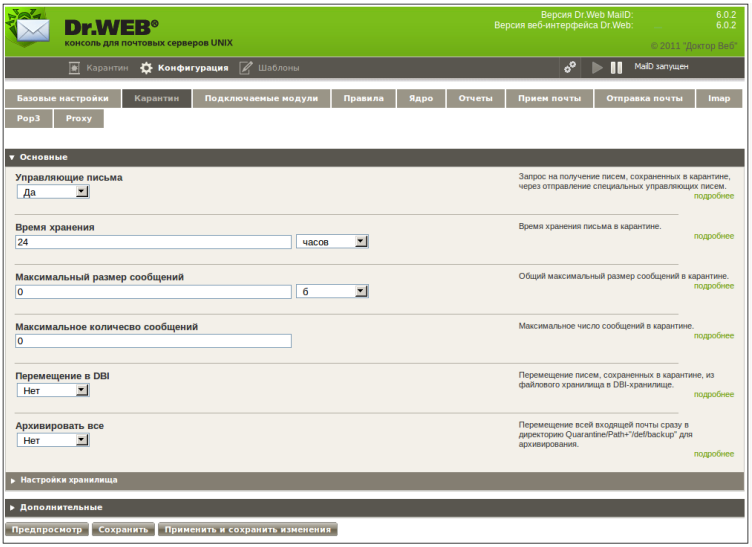


Рис. 32. Настройка карантина

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 DBI хранилищем.

Вкладка "Подключаемые модули"

На данной вкладке представлены общие настройки для всех подключаемых модулей, включенных в программный комплекс **Dr.Web для почтовых серверов UNIX**. Для настройки параметров каждого конкретного модуля необходимо перейти на соответствующую вкладку.

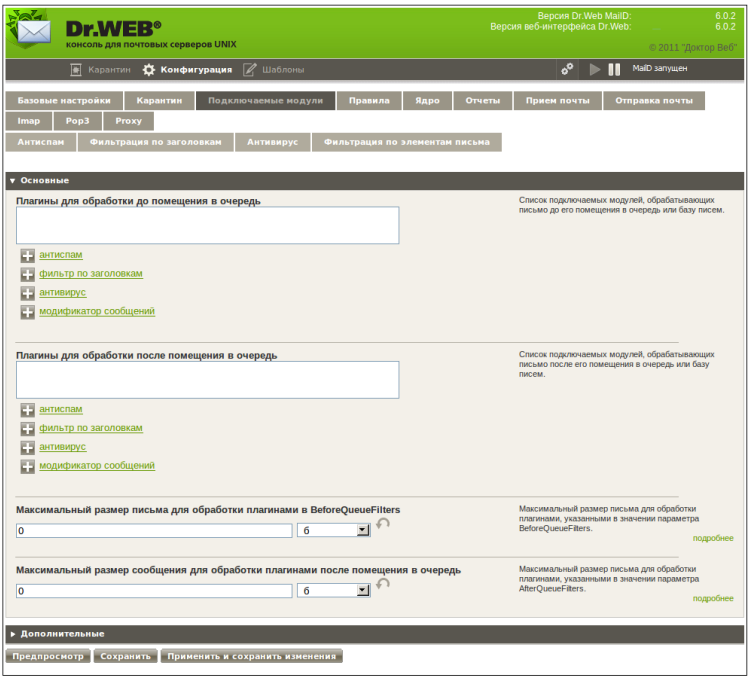


Рис. 33. Общие настройки работы подключаемых модулей

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. конфигурационные файлы).

Например, при добавлении слова "Infected" в качестве заголовка для зараженных файлов во вкладке "Антивирус", в конфигурационный файл плагина будет добавлено следующая строка:

```
Infected = cure,quarantine,notify,"add-header(infected!)"
```

На этой вкладке осуществляется настройка антиспам-модуля, входящего в программный комплекс **Dr.Web для почтовых серверов UNIX**.

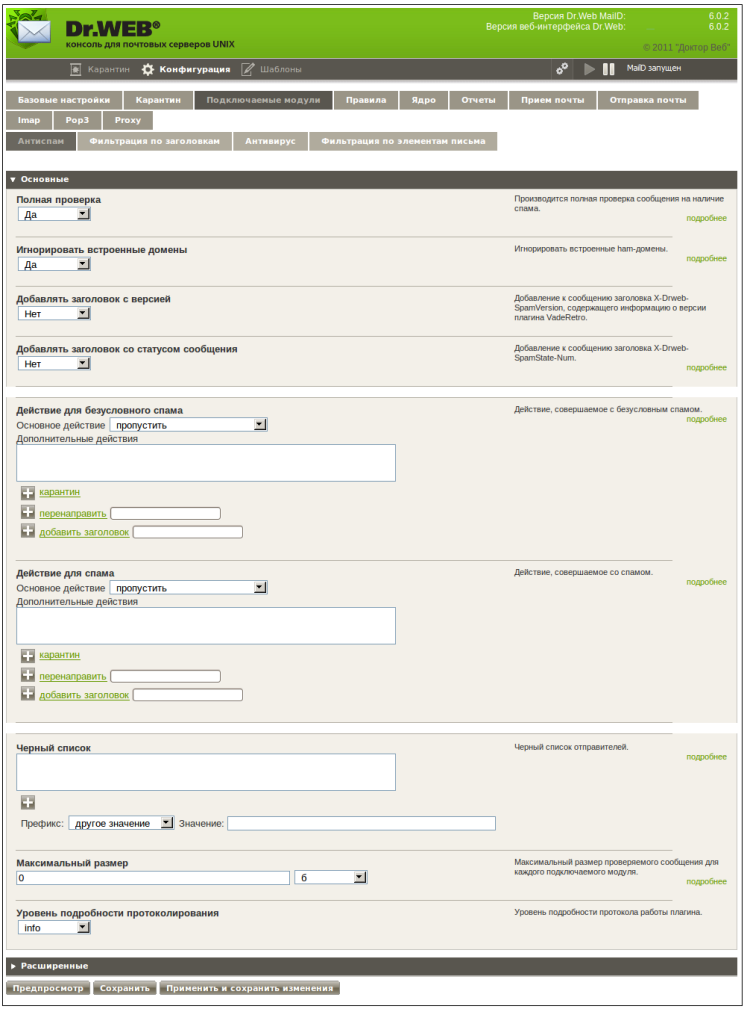


Рис. 34. Общие настройки антиспам-модуля

На данной вкладке осуществляется настройка модуля фильтрации по заголовкам, позволяющего осуществлять фильтрацию почтовых сообщений на основе их заголовков.

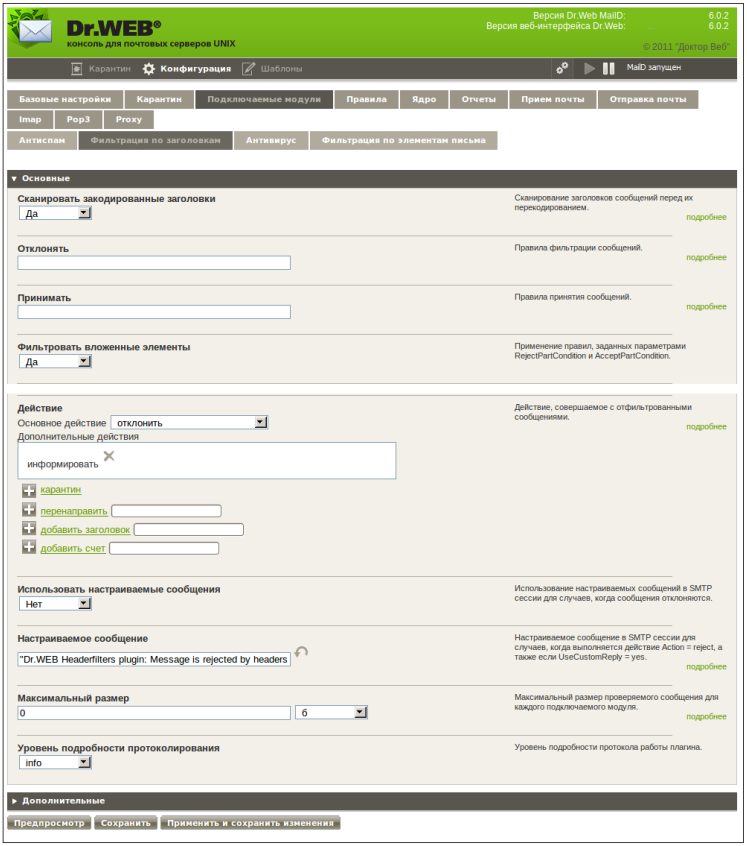


Рис. 35. Общие настройки модуля фильтрации по заголовкам

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.

На этой вкладке осуществляется настройка антивирусного

модуля, входящего в программный комплекс **Dr.Web для почтовых серверов UNIX**.

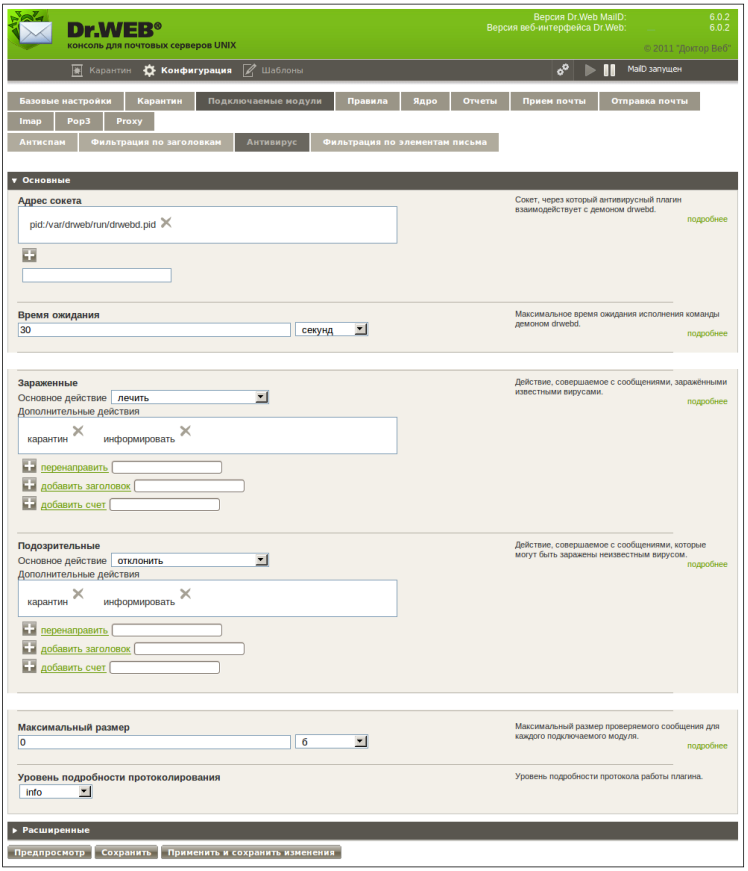


Рис. 36. Общие настройки антивирусного модуля

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.

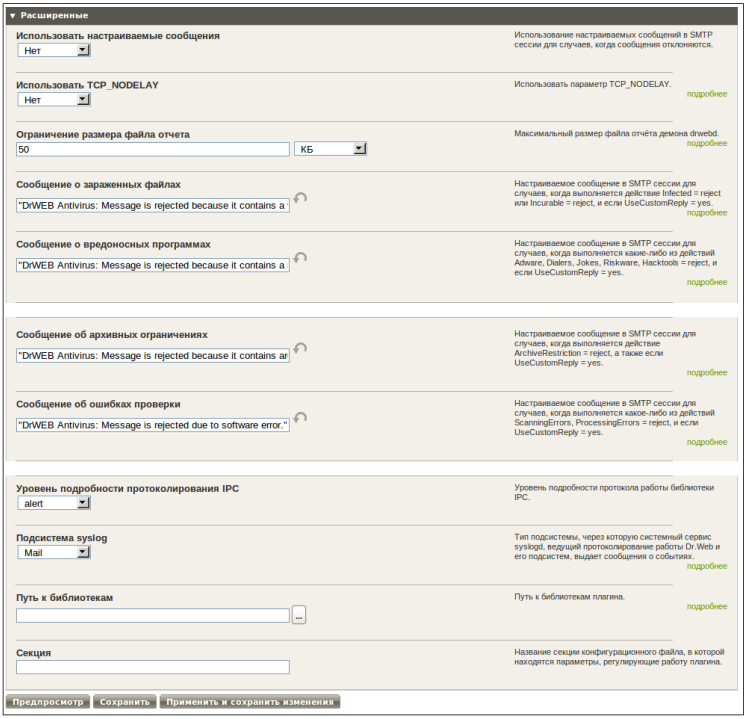


Рис. 37. Дополнительные настройки антивирусного модуля

На этой вкладке осуществляется настройка модуля фильтрации по элементам письма.

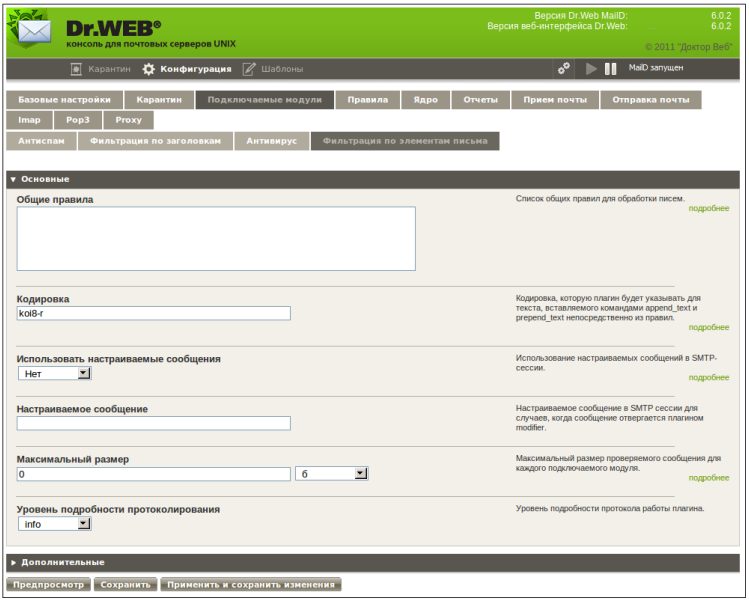


Рис. 38. Основные настройки модуля фильтрации по элементам письма

Вкладка "Правила"

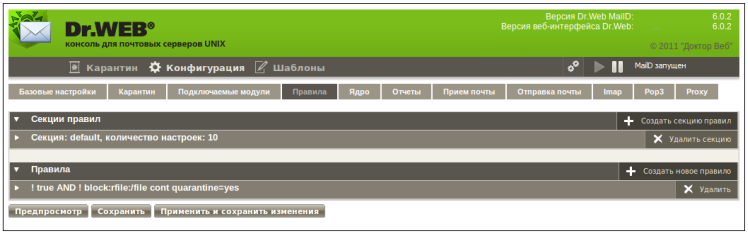


Рис. 39. Правила

Данная вкладка содержит настройки секции `[Rules]` конфигурационного файла **Dr.Web MailD**. С ее помощью можно создавать и отдельные правила, и пользовательские наборы

настроек для последующего использования их в правилах.

Чтобы создать правило, нажмите кнопку **Создать новое правило**. Чтобы отредактировать любое правило (как только что созданное, так и старое), достаточно кликнуть на него левой кнопкой мыши.

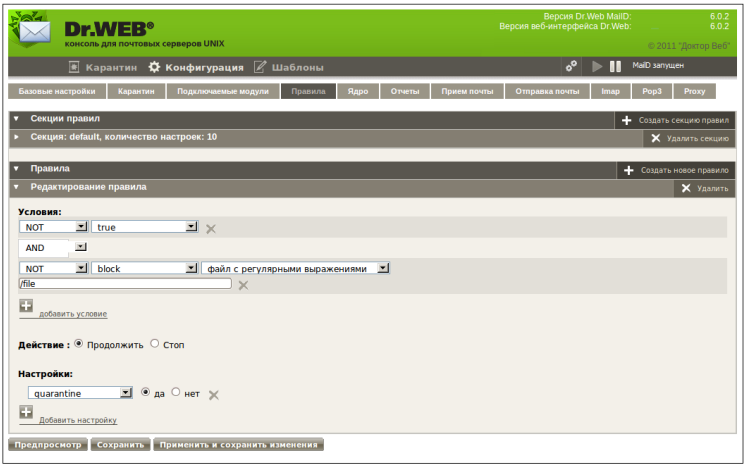


Рис. 40. Редактирование правила

При редактировании правила должны быть заданы значения во всех трех разделах: **Условия**, **Действие** и **Настройки**. При составлении условий могут использоваться логические операторы.

В текущей версии **Консоли** сложные правила, содержащие объединения, не поддерживаются. Работа с подобными правилами возможна только посредством правки конфигурационного файла.

Чтобы создать набор настроек, нажмите кнопку **Создать секцию правил**. Чтобы отредактировать любую группу настроек (как только что созданную, так и старую), достаточно кликнуть на нее левой кнопкой мыши.

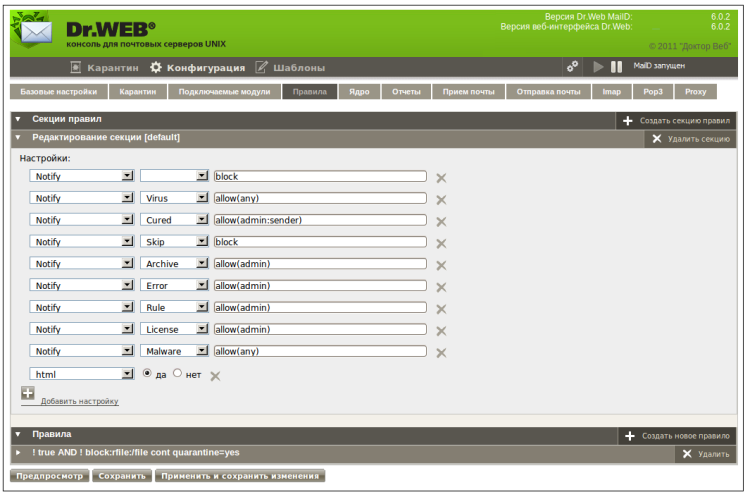


Рис. 41. Редактирование секции правил

Вкладка "Ядро"

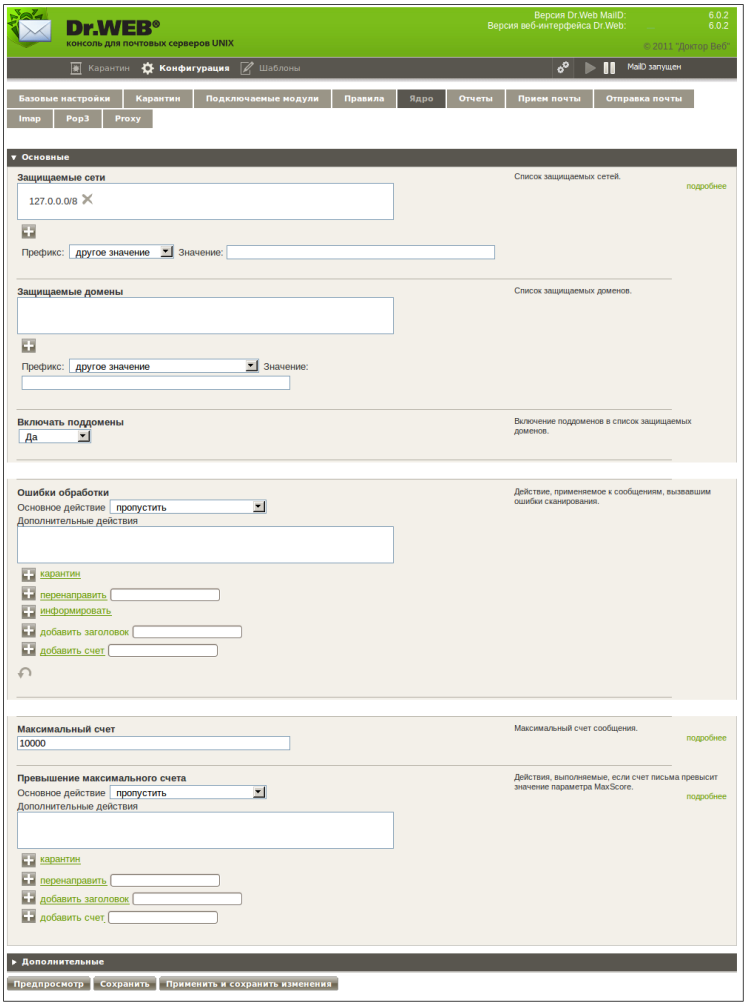


Рис. 42. Общие настройки Ядра

На этой вкладке вы можете указать адрес электронной почты,

используемый по умолчанию для перенаправления писем, отфильтрованных каким-либо плагином, 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.

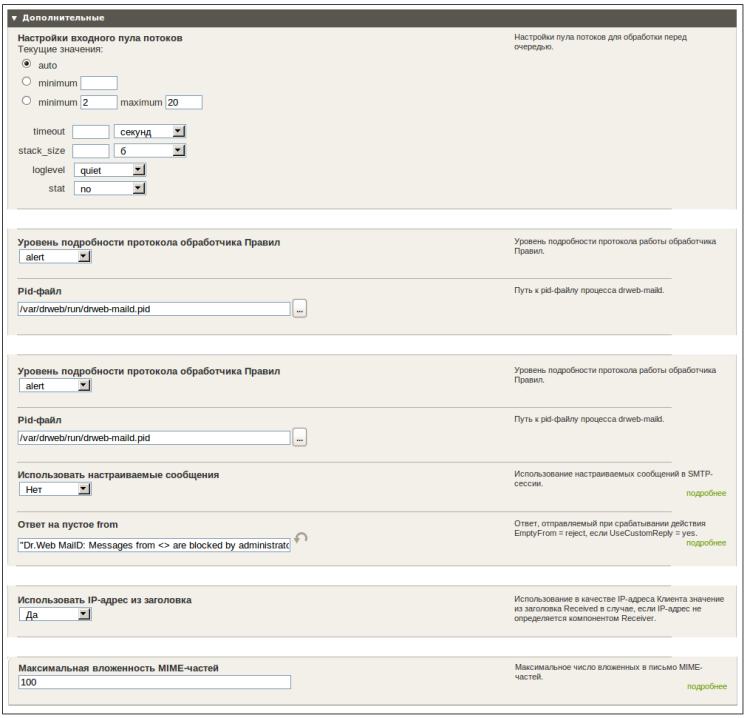


Рис. 43. Дополнительные настройки Ядра

В секции **Настройки базы писем** вы можете настроить работу с базой писем.

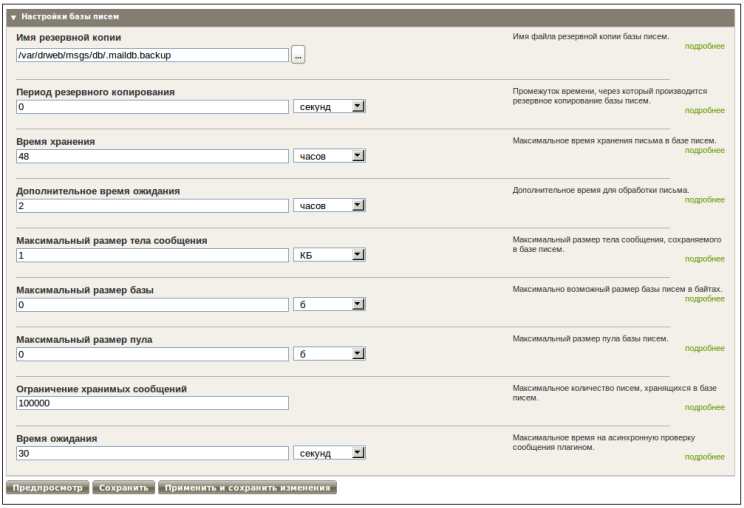


Рис. 44. Настройки секции MailBase

Вкладка "Отчеты"

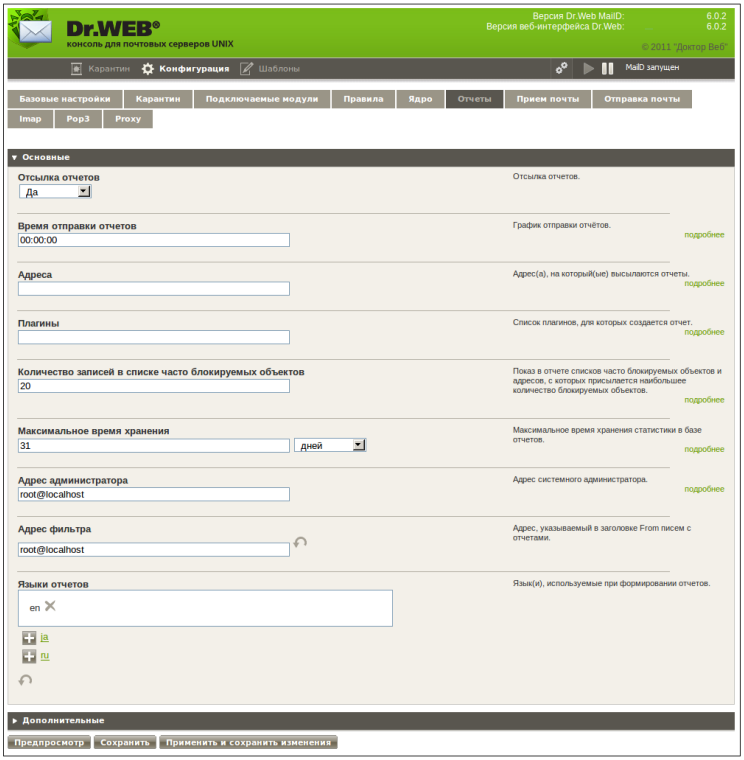


Рис. 45. Настройка отчетов

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"

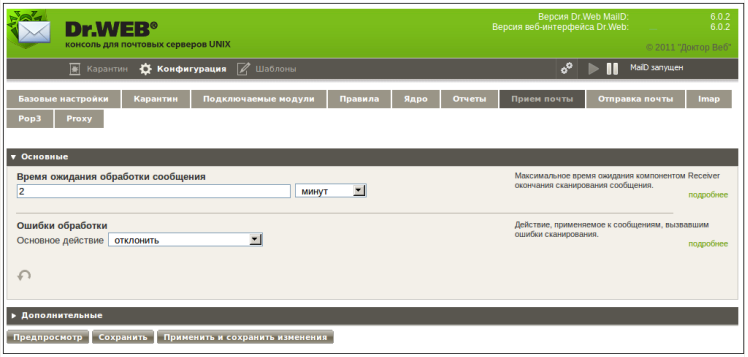


Рис. 46. Общие настройки приема почты

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.

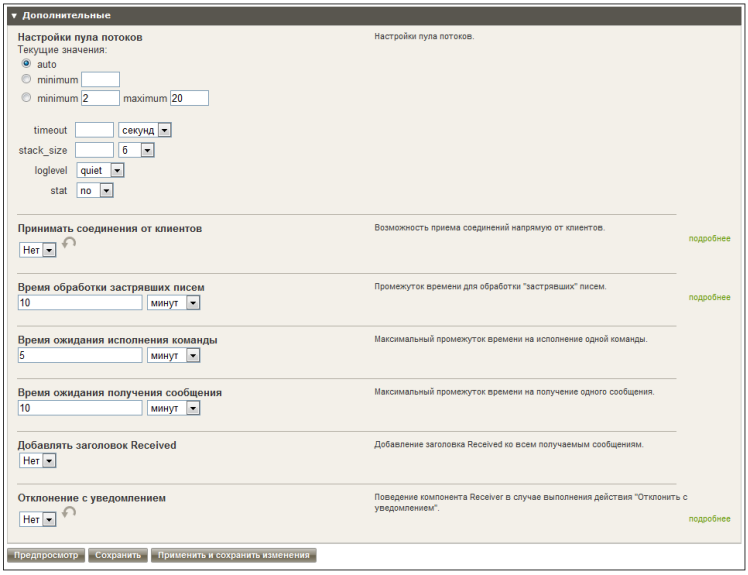


Рис. 47. Дополнительные настройки работы почтовой системы

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"

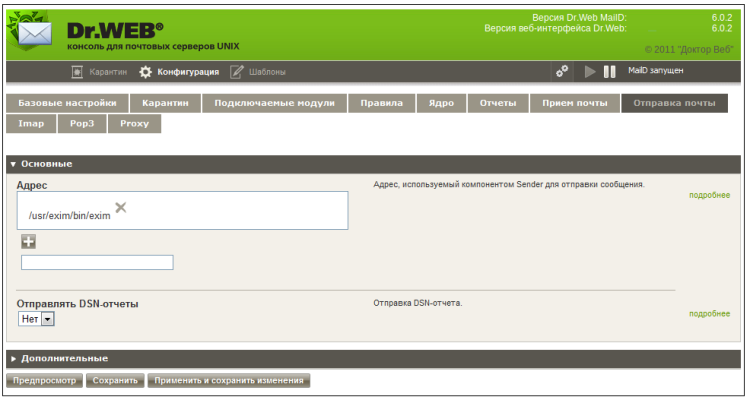


Рис. 48. Настройка отправки почты

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 **Демоном** и подключаемыми модулями.

Вкладка "IMAP"

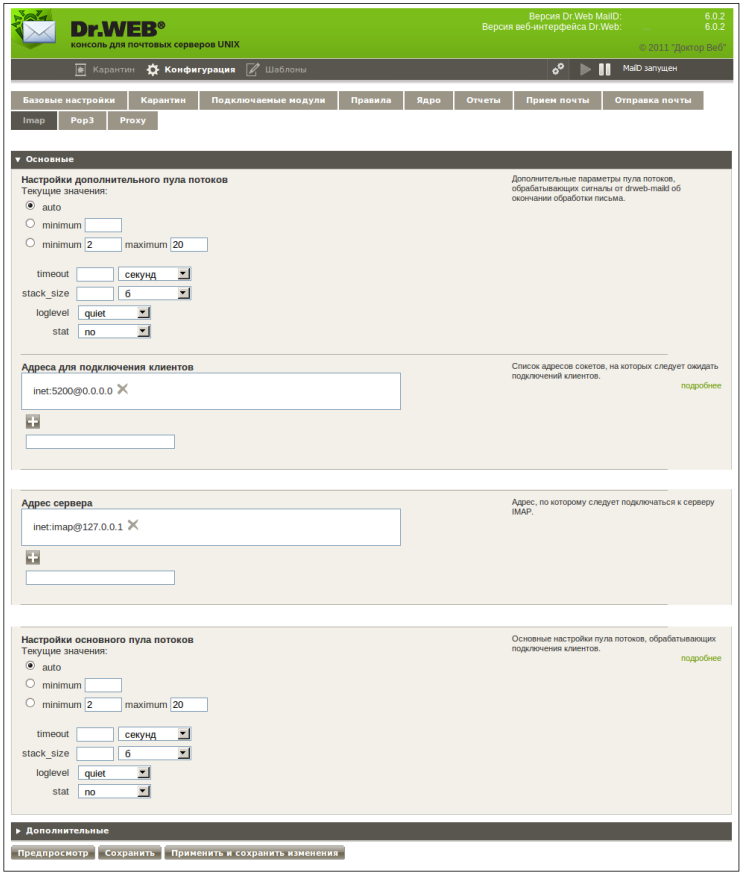


Рис. 49. Настройка фильтра IMAP

На данной вкладке вы можете указать настройки IMAP-фильтра для проверки почты по протоколу IMAP.

Вкладка "POP3"

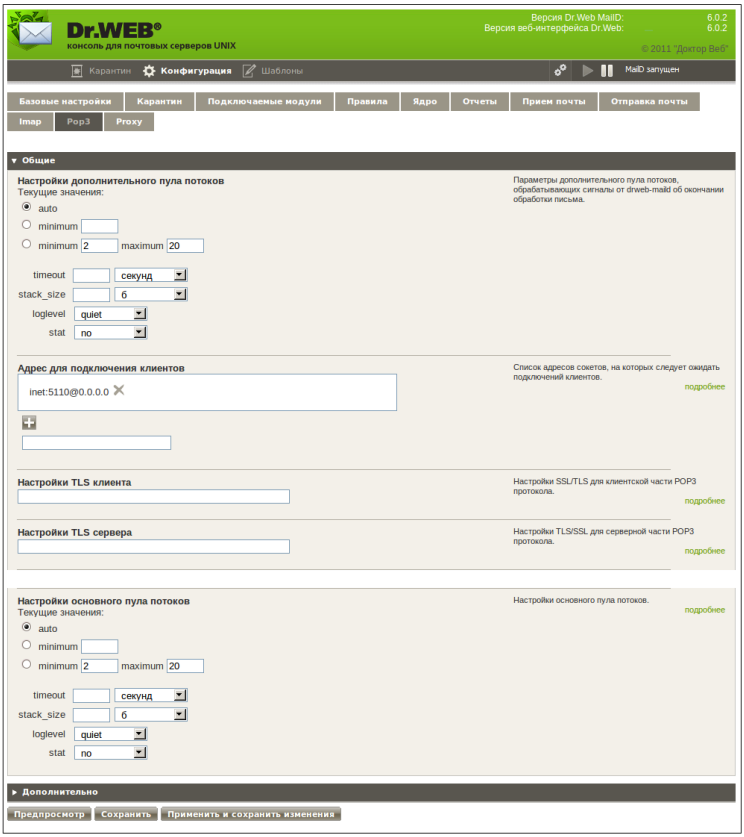


Рис. 50. Настройка фильтра POP3

На данной вкладке вы можете указать настройки POP3-фильтра для проверки почты по протоколу POP3.

Вкладка "Proxy"

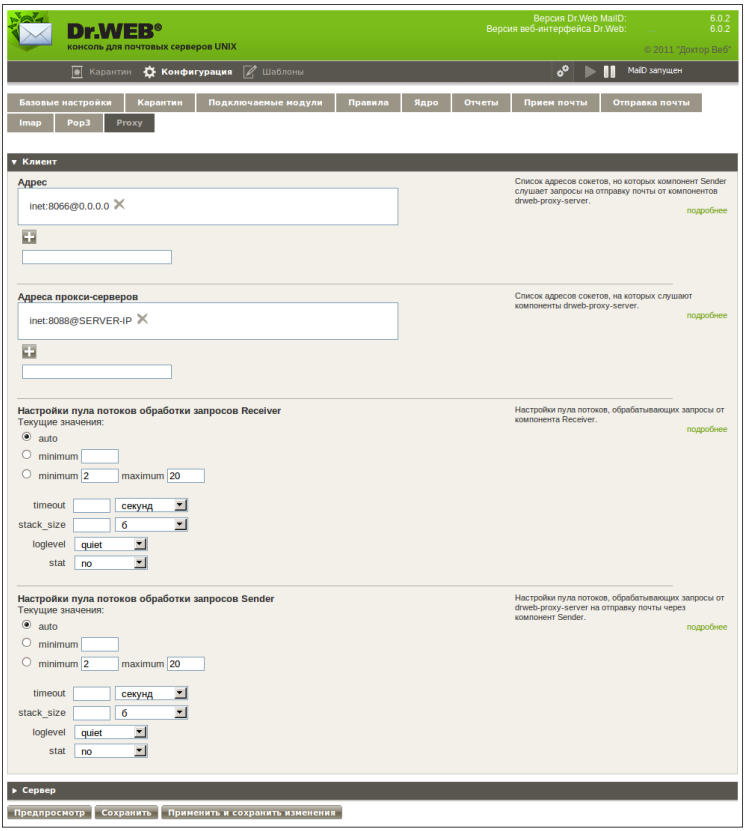


Рис. 51. Настройка прокси-сервера

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, которые генерируются и высылаются различным типам получателей при обнаружении в письме вредоносных объектов, а также при возникновении ошибок в работе **Демона** или подключаемых модулей.

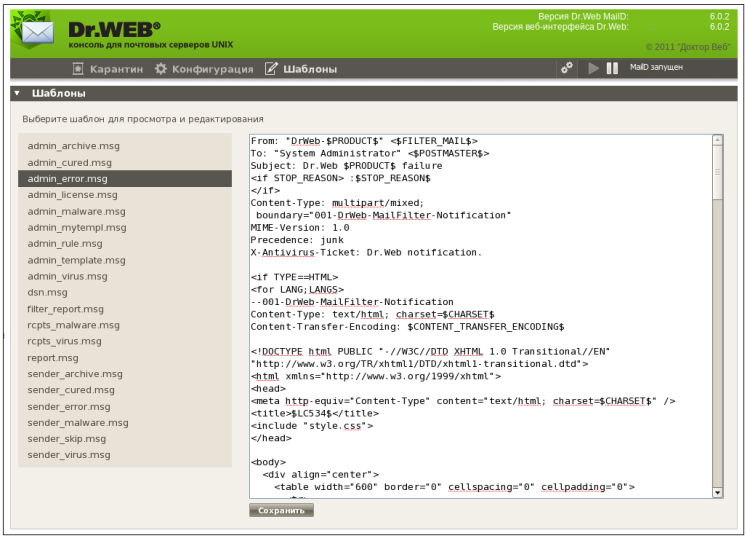


Рис. 52. Шаблоны

`ADMIN_ARCHIVE.msg` – шаблон с отчетом об ошибке сканирования письма с архивом во вложении, если на этот архив распространяются ограничения, заданные в главном конфигурационном файле `drweb32.ini`. Этот отчет направляется системному администратору.

`ADMIN_CURED.msg` – шаблон с отчетом об успешном излечении зараженного письма. Этот отчет направляется системному администратору.

`ADMIN_ERROR.msg` – шаблон с отчетом об ошибке в работе

Демона или подключаемого модуля. Этот отчет направляется системному администратору.

ADMIN_LICENSE.msg – шаблон с отчетом об ошибке сканирования письма, возникшей в связи с ограничениями, налагаемыми лицензией. Этот отчет направляется системному администратору.

ADMIN_MALWARE.msg – шаблон с отчетом об обнаружении в письме вредоносных программ. Этот отчет направляется системному администратору.

ADMIN_RULE.msg – шаблон с отчетом о блокировке письма в соответствии с заданным правилом. Этот отчет направляется системному администратору.

ADMIN_VIRUS.msg – шаблон с отчетом об обнаружении в письме вирусов. Этот отчет направляется системному администратору.

DSN.msg – шаблон с отчетом о доставке письма.

RCPTS_MALWARE.msg – шаблон с отчетом об обнаружении в письме вредоносных программ. Этот отчет направляется получателю.

RCPTS_VIRUS.msg – шаблон с отчетом об обнаружении в письме вирусов. Этот отчет направляется получателю.

REPORT.msg – шаблон для регулярных отчетов **Демона**.

SENDER_ARCHIVE.msg – шаблон с отчетом об ошибке сканирования письма с архивом во вложении, если на этот архив распространяются ограничения, заданные в главном конфигурационном файле drweb32.ini. Этот отчет направляется отправителю письма.

SENDER_CURED.msg – шаблон с отчетом об успешном излечении зараженного письма. Этот отчет направляется отправителю письма.

SENDER_ERROR.msg – шаблон с отчетом об ошибке в работе **Демона** или подключаемого модуля. Этот отчет направляется отправителю письма.

SENDER_MALWARE.msg – шаблон с отчетом об обнаружении в письме вредоносных программ. Этот отчет направляется отправителю письма.

SENDER_VIRUS.msg – шаблон с отчетом об обнаружении в письме вирусов. Этот отчет направляется отправителю письма.

SENDER_SKIP.msg – шаблон с отчетом 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 **Демона** или подключаемых модулей. Этот отчет направляется отправителю письма.

Работа в Enterprise режиме

Для начала работы **Консоли** в режиме централизованной защиты, необходимо произвести настройку **Агента**, описанную в соответствующем разделе. После внесения необходимых изменений, откройте базовые настройки **Консоли**, нажав кнопку  в верхнем меню навигации веб-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 **Агентом** от **Enterprise Сервера**. Изменения конфигурации, внесенные в данном режиме, вступят в силу только после перевода **Агента** в режим Standalone.

- `Yes` – **Консоль** получает конфигурационные данные из сокета **Агента**. В случае, если при этом **Агент** 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 **Агента**.

При возникновении проблем подключения к серверу, 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, **Агент** завершит свою работу. В этом случае проверьте настройки и попробуйте перезапустить **Агент** и **Консоль**.
- Если ранее вы уже подключались к серверу централизованной защиты, но в данный момент он недоступен (например, в случае проблем с соединением), **Агент** 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 `Enterprise` режим, в верхнем меню навигации страницы будет отображена надпись **(CPM)** (Аббревиатура от Central Protection Mode)**.**

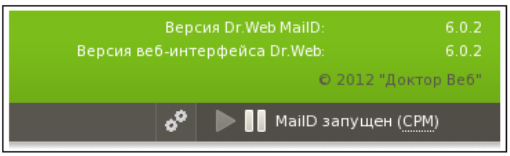


Рис. 53. Режим работы консоли Dr.Web для почтовых серверов Unix

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.

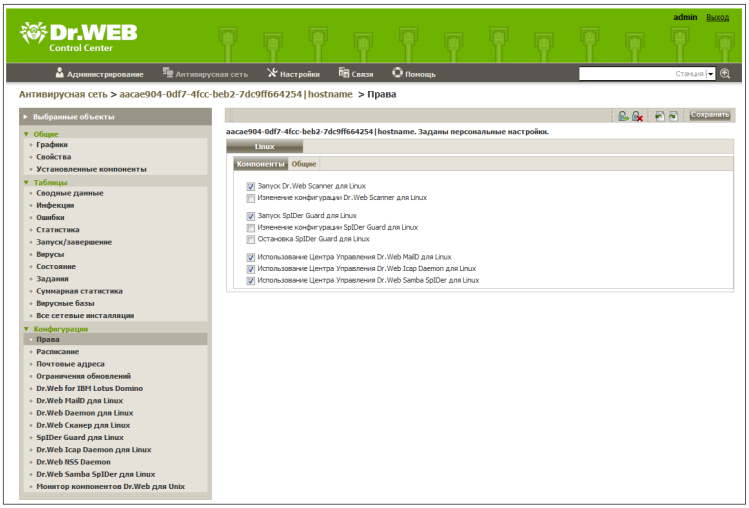


Рис. 54. Окно настройки прав пользователя рабочей станции

- 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.

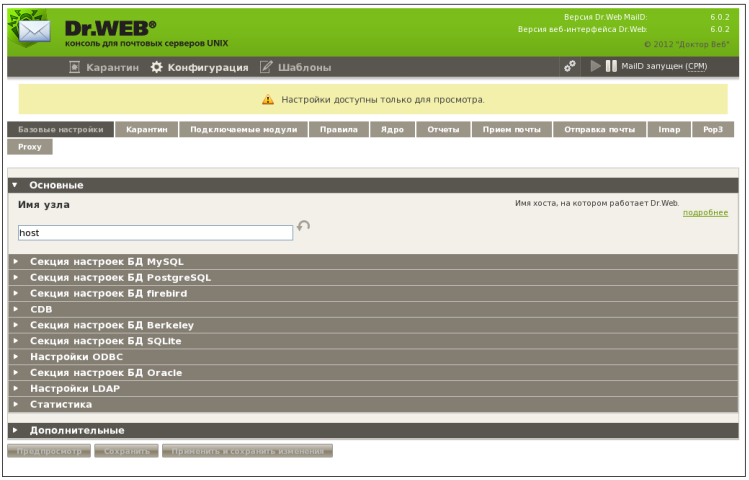


Рис. 55. Запрет на изменение конфигурации пользователем рабочей станции

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 следующее дерево:

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**. Для этого в интерфейсе **Центра Управления** 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.

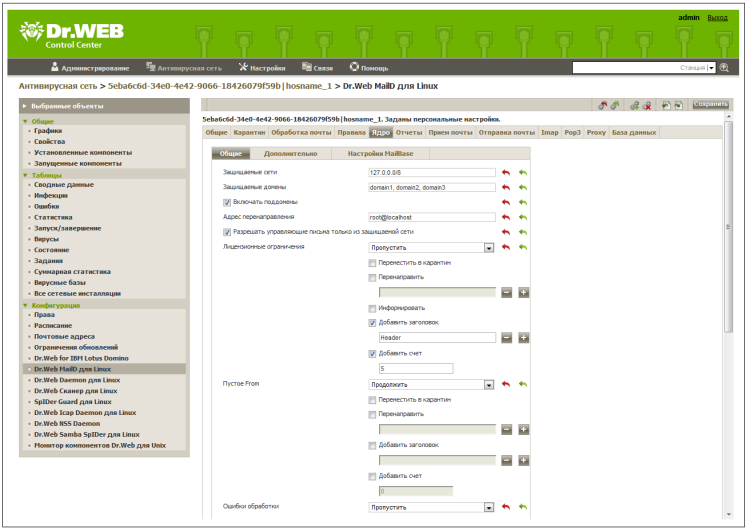


Рис. 56. Настройка Dr.Web MailD для Linux через интерфейс Центра Управления Dr.Web

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 **Enterprise Сервером** и 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 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 **Антивируса 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"* обладают правами "только для чтения" для просмотра доступных им групп.
- *Администраторы по умолчанию*. После установки **Enterprise Сервера** 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 отчета компонентов.