ViPNet Coordinator HW 4

Настройка с помощью командного интерпретатора

infotecs

ФРКЕ.00130-03 90 04

ОАО «ИнфоТеКС»

127287, г. Москва, Старый Петровско-Разумовский проезд, дом 1/23, строение 1

Тел: (495) 737-61-96 (горячая линия), 737-61-92, факс 737-72-78

Сайт компании «ИнфоТеКС»: http://www.infotecs.ru

Электронный адрес службы поддержки: hotline@infotecs.ru

Содержание

Введение

О документе

В документе описаны основные сценарии настройки ViPNet Coordinator HW с помощью командного интерпретатора, а также работа с журналами ViPNet Coordinator HW.

Для кого предназначен документ

Документ предназначен для администраторов, отвечающих за настройку и эксплуатацию ViPNet Coordinator HW.

Соглашения документа

Ниже перечислены соглашения, принятые в этом документе для выделения информации.

Таблица 1. Обозначения, используемые в примечаниях

Обозначение	Описание
	**Внимание!** Указывает на обязательное для исполнения или следования действие или информацию.
	**Примечание.** Указывает на необязательное, но желательное для исполнения или следования действие или информацию.
	**Совет.** Содержит дополнительную информацию общего характера.

Таблица 2. Обозначения, используемые для выделения информации в тексте

Обозначение	Описание
Название	Название элемента интерфейса. Например, заголовок окна, название поля, кнопки или клавиши.
Клавиша+Клавиша	Сочетание клавиш. Чтобы использовать сочетание клавиш, следует нажать первую клавишу и, не отпуская ее, нажать вторую клавишу.
Меню > Подменю > Команда	Иерархическая последовательность элементов. Например, пункты меню или разделы на панели навигации.
`Код`	Имя файла, путь, фрагмент текстового файла (кода) или команда, выполняемая из командной строки.

При описании команд в данном документе используются следующие условные обозначения:

- Команды, которые могут быть выполнены только в режиме администратора, содержат приглашение с символом «#». Например:

  ```
  hostname# команда
  ```

- Команды, которые могут быть выполнены в режиме и пользователя, и администратора, содержат приглашение с символом «>». Например:

  ```
  hostname> команда
  ```

- Параметры, которые должны быть заданы пользователем, заключены в угловые скобки. Например:

  ```
  команда <параметр>
  ```

- Необязательные параметры или ключевые слова заключены в квадратные скобки. Например:

  ```
  команда <обязательный параметр> [необязательный параметр]
  ```

- Если при вводе команды можно указать один из нескольких параметров, допустимые варианты заключены в фигурные скобки и разделены вертикальной чертой. Например:

  ```
  команда {вариант-1 | вариант-2}
  ```

Связанные документы

В таблице ниже перечислены документы, входящие в комплект документации ViPNet Coordinator HW помимо данного документа.

Таблица 3. Связанные документы

Документ	Содержание
«ViPNet Coordinator HW. Общее описание»	Описание общей информации по ViPNet Coordinator HW, а также существующих исполнений и характеристик аппаратных платформ
«ViPNet Coordinator HW. Подготовка к работе»	Описание подготовки ViPNet Coordinator HW к использованию, развертывания виртуального образа ViPNet Coordinator HW, работы со справочниками и ключами узла, обновления ПО, резервного копирования и восстановления настроек
«ViPNet Coordinator HW. Настройка с помощью веб-интерфейса»	Описание основных сценариев настройки ViPNet Coordinator HW с помощью веб-интерфейса
«ViPNet Coordinator HW. Сценарии работы»	Описание практических сценариев использования ViPNet Coordinator HW, которые требуют комплексного применения различных команд и базовых схем настройки ViPNet Coordinator HW
«ViPNet Coordinator HW. Справочное руководство по командному интерпретатору»	Описание команд ViPNet Coordinator HW
«ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам»	Описание конфигурационных файлов управляющего демона и системы защиты от сбоев
«ViPNet Coordinator HW. Лицензионные соглашения на компоненты сторонних производителей»	Лицензионные соглашения на компоненты сторонних производителей, которые использовались при разработке ПО для ViPNet Coordinator HW

О программно-аппаратном комплексе ViPNet Coordinator HW

Программно-аппаратный комплекс ViPNet Coordinator HW представляет собой интегрированное решение на базе специализированной аппаратной платформы и программного обеспечения ViPNet, которое функционирует под управлением адаптированной ОС GNU/Linux.

ViPNet Coordinator HW выступает в роли VPN-сервера и предназначен для использования в IP-сетях, защита которых организуется с применением комплекса программных продуктов ViPNet.

ViPNet Coordinator HW в сети ViPNet реализует функции координатора (см. глоссарий, стр. 269), а также ряд дополнительных функций.

Описание всех функций ViPNet Coordinator HW см. в документе «ViPNet Coordinator HW. Общее описание».

Обратная связь

Дополнительная информация

Сведения о продуктах и решениях ViPNet, распространенные вопросы и другая полезная информация собраны на сайте ОАО «ИнфоТеКС»:

- Веб-портал документации ViPNet http://docs.infotecs.ru.
- Описание продуктов ViPNet http://www.infotecs.ru/products/line/.
- Информация о решениях ViPNet http://www.infotecs.ru/solutions/.
- Сборник часто задаваемых вопросов (FAQ) http://www.infotecs.ru/support/faq/.
- Форум пользователей продуктов ViPNet http://www.infotecs.ru/forum.

Контактная информация

С вопросами по использованию продуктов ViPNet, пожеланиями или предложениями свяжитесь со специалистами ОАО «ИнфоТеКС». Для решения возникающих проблем обратитесь в службу технической поддержки.

- Техническая поддержка для пользователей продуктов ViPNet: hotline@infotecs.ru.
- Форма запроса в службу технической поддержки http://www.infotecs.ru/support/request/.
- Консультации по телефону для клиентов, имеющих расширенный уровень технического сопровождения:

  8 (495) 737-6196,

  8 (800) 250-0260 — бесплатный звонок из любого региона России (кроме Москвы).

Распространение информации об уязвимостях продуктов ОАО «ИнфоТеКС» регулируется политикой ответственного разглашения http://infotecs.ru/products/disclosure.php. Если вы обнаружили уязвимости в продуктах компании, сообщите о них по адресу security-notifications@infotecs.ru.

1

Работа с командным интерпретатором

Основные сведения по работе с командным интерпретатором

Все операции по администрированию ViPNet Coordinator HW вы можете выполнить с помощью командного интерпретатора ViPNet. Работа в командном интерпретаторе возможна при различных способах подключения к ViPNet Coordinator HW:

- Локальное подключение к ViPNet Coordinator HW с помощью консоли (на стр. 20).
- Удаленное подключение к ViPNet Coordinator HW с помощью протокола SSH (на стр. 22).

Командный интерпретатор независимо от способа подключения к ViPNet Coordinator HW запускается автоматически после успешной аутентификации пользователя (см. «Аутентификация пользователя на ViPNet Coordinator HW» на стр. 25). После запуска командный интерпретатор находится в режиме пользователя (обозначается символом «>» в приглашении интерпретатора). В режиме пользователя недоступны некоторые команды, требующие прав администратора.

Чтобы перейти в режим администратора, требуется пройти аутентификацию администратора (см. «Аутентификация администратора на ViPNet Coordinator HW» на стр. 27). В режиме администратора доступны все возможные настройки.

В случае одновременной работы нескольких командных интерпретаторов (независимо от типа подключения — локального или удаленного) только один из них может находиться в режиме администратора. При попытке перейти в некотором командном интерпретаторе в режим администратора проверяется режим работы остальных запущенных интерпретаторов. Если остальные интерпретаторы находятся в режиме пользователя, то данный интерпретатор переходит в режим администратора. Если один из интерпретаторов находится в режиме администратора, то выводится информация об узле, на котором он запущен, и предложение принудительно завершить работу этого интерпретатора. В случае согласия работа интерпретатора, находящегося в режиме администратора, завершается, после чего данный интерпретатор переходит в режим администратора.

Чтобы просмотреть информацию обо всех запущенных сессиях командного интерпретатора, выполните команду `who`.

Чтобы принудительно завершить любую сессию командного интерпретатора, выполните команду `admin kick`. Выполнить команду можно только в режиме администратора.

Чтобы вернуться в режим пользователя, используйте сочетание клавиш **Ctrl+D** или выполните команду `exit`.

Чтобы выйти из командного интерпретатора, используйте сочетание клавиш **Ctrl+D** или выполните команду `exit`. В случае выхода из командного интерпретатора на консоли снова появится приглашение для аутентификации пользователя.

Интерфейс командного интерпретатора

Командный интерпретатор ViPNet принимает от вас текстовые команды, состоящие из слов, разделенных пробелами. Для облегчения ввода команд в этом интерпретаторе предусмотрены следующие средства:

- сокращенный ввод команд (на стр. 17);
- автодополнение команд (на стр. 17);
- контекстная справка (на стр. 18);
- сочетания клавиш (на стр. 18).

Сокращенный ввод команд

Слова команды распознаются по минимальному числу символов, позволяющих отличить ее от других команд, ввод которых возможен в текущей ситуации.

Например:

- Если у двух команд первое слово начинается с символа «i» (`inet` и `iplir`), то для указания этих команд достаточно ввести первые две буквы соответствующего слова (`in` и `ip`).
- После слова `iplir` в качестве второго слова возможны только слова `stop` и `start`. Поэтому для указания соответствующих команд достаточно ввести `sto` и `sta`.

Таким образом, вместо полной команды `iplir start` достаточно ввести `ip sta`.

Если среди введенных символов есть ошибочные, то вся команда считается ошибочной, даже если она может быть однозначно определена по первым правильно введенным символам (то есть в приведенном выше примере команда `iplor sta` будет ошибочной).

Автодополнение команд

Если введенные символы однозначно определяют какую-либо команду или параметр команды, то вы можете выполнить автоматическое дополнение недостающих символов, нажав клавишу **Tab**:

```
hostname> en<Tab>
hostname> enable _
```

Автодополнение работает только для слова, на котором находится курсор. Автодополнение можно использовать при вводе любых команд и параметров, кроме параметров, значение которых задается пользователем произвольно (например, имена сетевых интерфейсов).

Контекстная справка

Контекстная справка позволяет вам просмотреть информацию о командах и параметрах, ввод которых возможен в текущей ситуации. Контекстная справка вызывается с помощью символа «?».

Чтобы просмотреть список всех доступных вам групп команд, в приглашении интерпретатора ViPNet введите символ «?»:

```
hostname> ?
inet                Set of commands for routing, interfaces and network tools
failover            Control command for Failover daemon
iplir               Control command for IpLir daemon
mftp                Control command for MFTP daemon
enable              go to administrator mode
exit                leave command interpreter
version             view the versions of the appliance
who                 show vipnet sessions
machine             display or change machine settings
debug
ups
firewall            firewall management
alg                 control of the properties of the application-level gateway
hostname> _
```

Левая колонка списка содержит первое слово группы команд, правая — краткое описание ее назначения.

В случае ввода символа «?» в процессе набора команд, интерпретатор ViPNet предложит вам варианты завершения текущего или следующего слова команды, в зависимости от положения курсора:

```
hostname> machi?
machine             halt or reboot the machine
hostname> machi_
hostname> machine ?
halt                switch the machine off
reboot              reboot the machine
show                display statistics
hostname> machine _
```

После информации о вариантах завершения команды отображается приглашение интерпретатора ViPNet с ранее введенной командой для редактирования. Редактировать команды можно как обычно: стирать символы клавишей **Backspace** или **Delete**, перемещаться по тексту с помощью клавиш со стрелками влево и вправо.

Сочетания клавиш

Командный интерпретатор поддерживает стандартные сочетания клавиш, перечисленные в таблице ниже.

Таблица 4. Поддерживаемые интерпретатором сочетания клавиш

Сочетание клавиш	Действие
Ctrl+U	стереть всю команду
Ctrl+K	стереть все от курсора до конца строки
Ctrl+A	перейти в начало строки
Ctrl+E	перейти в конец строки
Ctrl+B	перейти на один символ назад
Ctrl+F	перейти на один символ вперед
Ctrl+H	стереть символ перед курсором
Ctrl+W	стереть слово перед курсором

Локальное подключение к ViPNet Coordinator HW с помощью консоли

Чтобы локально подключиться к ViPNet Coordinator HW, выполните следующие действия:

1. Если этого не было сделано ранее, выберите и настройте консоль для подключения к ViPNet Coordinator HW (см. «Выбор консоли при локальном подключении к ViPNet Coordinator HW» на стр. 20).
2. Выполните аутентификацию пользователя (см. «Аутентификация пользователя на ViPNet Coordinator HW» на стр. 25).
3. Если требуется выполнить настройку параметров ViPNet Coordinator HW, выполните аутентификацию администратора (см. «Аутентификация администратора на ViPNet Coordinator HW» на стр. 27).

Примечание. Если на ViPNet Coordinator HW истекают или уже истекли персональный ключ пользователя или ключи защиты ключей обмена (см. глоссарий, стр. 268), то на консоли появится соответствующее сообщение. Обратитесь к администратору ViPNet Coordinator HW для обновления ключей (см. «Обновление ключей при истечении срока их действия» на стр. 47).

Выбор консоли при локальном подключении к ViPNet Coordinator HW

При локальном подключении к ViPNet Coordinator HW в качестве консоли вы можете использовать следующее оборудование:

- монитор и клавиатура (обычная консоль);
- любое устройство, подключенное к COM-порту ViPNet Coordinator HW (COM-консоль).

Примечание. На некоторых аппаратных платформах ViPNet Coordinator HW вместо COM-порта RS-232 присутствует служебный Ethernet-порт RJ45. Этот порт также может использоваться для подключения к ViPNet Coordinator HW через COM-консоль. Подробнее об аппаратных платформах см. документ «ViPNet Coordinator HW. Общее описание».

При подключении COM-консоли к ViPNet Coordinator HW (например, с помощью программы PuTTY) установите значения параметров COM-порта, приведенные в таблице ниже.

Таблица 5. Требования к параметрам COM-порта для подключения консоли

Параметр	Значение
Speed (скорость обмена данными)	38400
Data (размер данных)	8
Parity (четность)	None
Stopbits (стоповые биты)	1
Тип терминала	VT100+

Примечание. В ViPNet Coordinator HW поддерживается только кодировка KOI8-R. Поэтому при подключении к компьютеру, на котором используется другая кодировка, возможны проблемы при вводе с клавиатуры и отображении на консоли символов нелатинского алфавита

К ViPNet Coordinator HW локально можно подключиться с помощью нескольких консолей. Одновременная работа во всех запущенных консолях возможна только в режиме пользователя. В режиме администратора при этом можно работать только в одной консоли. Поскольку возможна одновременная работа сразу в нескольких консолях при каждом включении ViPNet Coordinator HW, независимо от количества подключенных консолей, предлагается выбрать консоль для дальнейшей работы следующим образом:

1. На все консоли, подключенные к ViPNet Coordinator HW, выводится приглашение нажать любую клавишу. Если в течение 10 секунд вы не нажмете клавишу ни на одной из консолей, то считается, что клавиша нажата на обычной консоли.
2. На консоль, на которой нажата клавиша, выводится список возможных консолей и приглашение выбрать нужную. Если в течение 5 секунд вы не выберите консоль, то автоматически выбирается обычная консоль.

Все последующие сообщения о загрузке ОС и приглашение для входа в систему будут выводиться на выбранной консоли.

Удаленное подключение к ViPNet Coordinator HW с помощью протокола SSH

Удаленное подключение к командному интерпретатору следует осуществлять только с выделенных рабочих мест по каналу, защищенному средствами ПО ViPNet. Вы можете подключаться к ViPNet Coordinator HW с других защищенных узлов ViPNet, связанных с ним (связи между узлами сети ViPNet задаются в программе ViPNet Центр управления сетью (ЦУС) (см. глоссарий, стр. 265)).

Внимание! Предоставлять удаленный доступ к ViPNet Coordinator HW с незащищенных узлов запрещено. С помощью фильтров защищенной сети следует ограничить соединения между ViPNet Coordinator HW и рабочими местами администраторов, разрешив только удаленное управление и передачу данных по служебным протоколам ViPNet.

Для удаленного подключения вы можете использовать любой SSH-клиент с парольным типом аутентификации.

Примечание. В ViPNet Coordinator HW поддерживается только кодировка KOI8-R. Поэтому при подключении к компьютеру, на котором используется другая кодировка, возможны проблемы при вводе с клавиатуры и отображении на консоли символов нелатинского алфавита

Чтобы удаленно подключиться к ViPNet Coordinator HW, выполните следующие действия:

1. Если администратор сети или администратор ViPNet Coordinator HW задал способ аутентификации «Устройство», то на узле, с которого вы будете удаленно подключаться к ViPNet Coordinator HW, выполните аутентификацию пользователя в ПО ViPNet с использованием устройства (см. «Аутентификация пользователя на ViPNet Coordinator HW» на стр. 25).
2. С помощью SSH-клиента подключитесь к ViPNet Coordinator HW, указав необходимые параметры подключения и пароль.
3. Если в момент вашего подключения с ViPNet Coordinator HW установлены удаленные сессии других пользователей, и количество удаленных сессий максимальное, появится сообщение с предложением подключиться к ViPNet Coordinator HW в режиме администратора (см. «Ограничение количества удаленных сессий» на стр. 23). В случае согласия на такое подключение выполните аутентификацию администратора (см. «Аутентификация администратора на ViPNet Coordinator HW» на стр. 27).

   Если запущена другая удаленная сессия в режиме администратора, то прервите ее. В противном случае ваша попытка удаленного подключения будет завершена.

Если в течение 30 секунд вы не подтвердите подключение к ViPNet Coordinator HW в режиме администратора, то ваша удаленная сессия также будет завершена.

Примечание. Если во время удаленной сессии с ViPNet Coordinator HW вы не будете выполнять никакие действия в течение 30 минут, то сессия будет прервана. Подробнее см. Мониторинг неактивных удаленных сессий (на стр. 23).

Если на ViPNet Coordinator HW истекают или уже истекли персональный ключ пользователя или ключи защиты ключей обмена (см. глоссарий, стр. 268), то на консоли появится соответствующее сообщение. Обратитесь к администратору ViPNet Coordinator HW для обновления ключей (см. «Обновление ключей при истечении срока их действия» на стр. 47).

Ограничение количества удаленных сессий

На ViPNet Coordinator HW установлено ограничение на количество одновременно запущенных удаленных сессий пользователей. В зависимости от исполнения ViPNet Coordinator HW разрешено:

- 5 удаленных сессий одновременно, если используется ViPNet Coordinator HW всех исполнений HW50 и HW100.
- 30 удаленных сессий одновременно, если используется ViPNet Coordinator HW остальных исполнений.

Установленное ограничение изменить нельзя, так как увеличение одновременных удаленных подключений может отрицательно сказаться на выполнении основных функций ViPNet Coordinator HW.

При удаленном подключении пользователя к ViPNet Coordinator HW проверяется количество открытых удаленных сессий пользователей. Если количество сессий максимальное, то пользователю выдается сообщение об этом с предложением подключиться в режиме администратора. Если в течение 30 секунд от пользователя не будет получен ответ, подключение будет завершено. В случае ввода пользователем корректного пароля администратора будет открыта сессия. Возможна только одна удаленная сессия в режиме администратора. Поэтому если на момент запуска вашей сессии в режиме администратора запущена другая сессия, то ее потребуется прервать. После завершения работы администратора при выполнении команды `exit` сессия завершается без возможности перейти в режим пользователя.

Мониторинг неактивных удаленных сессий

При одновременном удаленном подключении по протоколу SSH большого количества пользователей нагрузка на ViPNet Coordinator HW значительно увеличивается, что может привести к сбою в работе. Во избежание такой ситуации производится мониторинг активности пользователей в удаленных сессиях.

Проверка активности пользователя в каждой удаленной сессии осуществляется один раз в 5 минут. Если пользователь не выполняет никаких действий в течение 30 минут, его сессия принудительно завершается.

Для просмотра текущего допустимого времени неактивности сессии пользователя выполните команду:

```
hostname> machine show session-timeout
```

Для установки максимально допустимого времени неактивности сессии пользователя в удаленной сессии выполните команду:

```
hostname# machine set session-timeout <время>
```

При работе ViPNet Coordinator HW в режиме кластера горячего резервирования (см. глоссарий, стр. 268) указанные настройки резервируются.

Аутентификация пользователя на ViPNet Coordinator HW

В ViPNet Coordinator HW предусмотрено два типа аутентификации пользователя:

- «Пароль». При аутентификации требуется ввести имя учетной записи и пароль пользователя. Каждый раз при вводе пароля вычисляется парольный ключ, который используется для доступа к вашему персональному ключу.
- «Устройство». При аутентификации требуется ввести имя учетной записи, подключить устройство, на котором сохранен персональный ключ, и ввести ПИН-код доступа к устройству.

Примечание. По умолчанию способ аутентификации задается администратором сети ViPNet при создании дистрибутива ключей (см. глоссарий, стр. 267). Администратор ViPNet Coordinator HW при необходимости может изменить способ аутентификации. Подробнее см. раздел Изменение способа аутентификации пользователя (на стр. 41).

Чтобы выполнить аутентификацию пользователя одним из указанных способов, выполните следующие действия:

1 Введите имя учетной записи пользователя `user`.

2 Для аутентификации с помощью пароля — введите пароль пользователя. Пароль будет совпадать с паролем дистрибутива ключей, если он не изменялся в процессе работы с ViPNet Coordinator HW (см. «Изменение пароля пользователя» на стр. 40). При вводе пароля на экране ничего не отображается. Если введен неверный пароль, на консоли появляется соответствующее сообщение и приглашение для повторной аутентификации.

Примечание. Если вы ввели неверный пароль, чтобы сделать очередную попытку ввода пароля подождите несколько секунд. Задержка реализована для предотвращения возможности подбора пароля методом перебора. С каждой новой неуспешной попыткой ввода пароля задержка увеличивается. Если вы ввели неверный пароль 10 раз подряд, задержка составит 25 минут, но после нее вы также сможете повторить попытку ввода пароля. При успешной попытке ввода пароля счетчик, который фиксирует неуспешные попытки, обнуляется. Также счетчик обнуляется после десятой неуспешной попытки ввода пароля.

События обо всех неуспешных попытках ввода пароля фиксируются в журнале устранения неполадок.

---

3 Для аутентификации с помощью устройства:

3.1 Подключите к одному из USB-разъемов ViPNet Coordinator HW (или компьютера, на котором развернут виртуальный образ ViPNet Coordinator HW) внешнее устройство, на которое сохранен ваш персональный ключ.

3.2 Введите ПИН-код доступа к подключенному устройству в ответ на сообщение `Enter PIN`.

Если введен неверный ПИН-код, аутентификация не будет выполнена.

В случае успешной аутентификации интерпретатор будет запущен в режиме пользователя. Вы можете приступать к работе с ViPNet Coordinator HW. О том, как перейти в режим администратора, см. раздел Аутентификация администратора на ViPNet Coordinator HW (на стр. 27).

Аутентификация администратора на ViPNet Coordinator HW

Чтобы перейти в режим администратора, выполните следующие действия:

1 Выполните команду `enable`.

2 Введите пароль администратора сетевого узла ViPNet (см. глоссарий, стр. 270). При вводе пароля на экране ничего не отображается. Если введен неверный пароль, на консоли появляется соответствующее сообщение и приглашение для повторной аутентификации.

Примечание. Если вы ввели неверный пароль, чтобы сделать очередную попытку ввода пароля подождите несколько секунд. Задержка реализована для предотвращения возможности подбора пароля методом перебора. С каждой новой неуспешной попыткой ввода пароля задержка увеличивается. Если вы ввели неверный пароль 10 раз подряд, задержка составит 25 минут, но после нее вы также сможете повторить попытку ввода пароля. При успешной попытке ввода пароля счетчик, который фиксирует неуспешные попытки, обнуляется. Также счетчик обнуляется после десятой неуспешной попытки ввода пароля.

События обо всех неуспешных попытках ввода пароля фиксируются в журнале устранения неполадок.

В случае ввода верного пароля интерпретатор перейдет в режим администратора — на консоли появится символ «#» в приглашении интерпретатора. В этом режиме вы можете выполнить те настройки, которые были недоступны в режиме пользователя.

2

Настройка системных параметров

Настройка даты и времени

Чтобы компьютер с программным обеспечением ViPNet Coordinator HW корректно взаимодействовал с другими защищенными узлами ViPNet, необходимо правильно настроить системные дату и время.

Внимание! Если системные дата и время заданы неверно, защищенные соединения с другими узлами ViPNet могут блокироваться.

Рекомендуется настроить синхронизацию системного времени по протоколу NTP (см. «Настройка параметров NTP-сервера» на стр. 163).

Если требуется настроить системные дату и время вручную, выполните следующие действия:

1 Для просмотра текущего времени выполните команду:

```
hostname> machine show date
```

2 Если требуется изменить часовой пояс, выполните команду:

```
hostname# machine set timezone {<Континент/Пояс> | <UTC>}
```

Примечание. Название континента и часового пояса должны начинаться с прописной буквы. Например, чтобы установить часовой пояс города Москвы, выполните команду:

```
hostname# machine set timezone Europe/Moscow
```

Чтобы посмотреть список всех существующих часовых поясов, введите данную команду без параметра.

3 Если требуется изменить системное время, выполните следующие действия:

3.1 Остановите демоны iplircfg, failoverd и mftpd с помощью команд:

```
hostname> iplir stop
hostname> failover stop
hostname> mftp stop
```

3.2 Настройте дату и время с помощью команды:

```
hostname# machine set date <YYYY-MM-DD> <hh:mm:ss>,
```

где `YYYY` — год, `MM` — месяц, `DD` — день, `hh` — часы, `mm` — минуты, `ss` — секунды.

3.3 Запустите демоны iplircfg, failoverd и mftpd с помощью команд:

```
hostname> iplir start
hostname> failover start
hostname> mftp start
```

Настройка параметров использования файла подкачки

Чтобы увеличить производительность ViPNet Coordinator HW за счет оптимизации скорости его процессов, рекомендуется настроить использование на нем файла подкачки.

Внимание! Использование файла подкачки в исполнениях с одним дисковым накопителем (ViPNet Coordinator HW50 и ViPNet Coordinator HW100 на аппаратных платформах HW100 X1, X8) невозможно.

Чтобы настроить параметры файла подкачки, выполните следующие действия:

1. Задайте размер файла подкачки с помощью команды:

   `hostname# machine swap set <size>`,

   указав с помощью параметра `<size>` нужный размер файла подкачки в мегабайтах.

   Если будет задан размер файла подкачки, превышающий размер доступного пространства на диске, появится соответствующее сообщение.

   

   Внимание! После создания файла подкачки на диске должно остаться не менее 256 Мбайт свободного пространства.

2. Включите использование файла подкачки с помощью команды:

   `hostname# machine swap mode on`

3. Чтобы отключить использование файла подкачки, выполните команду:

   `hostname# machine swap mode off`

   После выполнения данной команды файл подкачки будет удален.

4. Чтобы просмотреть сведения об использовании оперативной памяти и файле подкачки, выполните команду:

   `hostname# machine show memory`

Управление копиями конфигурации VPN

Виды копий конфигурации VPN

В ViPNet Coordinator HW существует возможность создания копий конфигурации VPN, которые позволяют при необходимости осуществлять возврат соответствующего программного обеспечения к другому состоянию (например, к состоянию до выполнения настроек или до приема обновленных справочников и ключей).

Копия конфигурации VPN — это совокупность настроек ViPNet Coordinator HW. Существуют два вида копий конфигурации VPN:

- частичная — включает в себя следующие конфигурационные файлы ПО ViPNet Coordinator HW: `iplir.conf` и все файлы с именем, соответствующим маске `iplir.conf-<интерфейс или группа интерфейсов>`, `failover.ini`, `mftp.conf`, файл с настройками апплета SGA;
- полная — включает в себя все файлы частичной копии конфигурации VPN, а также справочники и ключи.

Копии конфигурации VPN могут создаваться следующими способами:

- Автоматически (перед обновлением ПО или справочников и ключей). В этом случае по умолчанию создаются частичные копии конфигурации VPN. Вы можете изменить вид копии конфигурации VPN, создаваемой автоматически, или выключить автоматическое создание копий конфигурации VPN, отредактировав значение параметра `confsave` в секции `[upgrade]` в файле `mftp.conf`.

  Кроме того, с помощью параметра `maxautosaves` этой же секции вы можете задать максимальное число автоматически созданных копий конфигурации VPN, которое может храниться на ViPNet Coordinator HW. По умолчанию может храниться не более 10 автоматически созданных копий конфигурации VPN.

  Описание параметров секции `[upgrade]` см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам», в разделе «Секция [upgrade]».
- Вручную по команде. В этом случае всегда создаются только полные копии конфигурации VPN.

Создание копии конфигурации VPN

Внимание! По умолчанию перед обновлением ПО или справочников и ключей автоматически создается частичная копия конфигурации VPN (см. «Виды копий конфигурации VPN» на стр. 31). Вы можете изменить эту настройку в секции `[upgrade]` файла `mftp.conf` (подробнее см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам»).

Чтобы вручную создать копию текущей конфигурации VPN, выполните следующие действия:

1 Завершите работу демонов iplircfg, failoverd и mftpd с помощью команд:

```
hostname> iplir stop
hostname> failover stop
hostname> mftp stop
```

2 Выполните команду:

```
hostname# admin config save <имя>,
```

указав имя копии конфигурации VPN, используя только буквы латинского алфавита, цифры, символы «–» и «_». Если имя копии конфигурации VPN состоит из нескольких слов, заключите его в кавычки (например: `"settings copy_1"`).

3 Если копия конфигурации VPN с выбранным именем уже существует, будет предложено ее перезаписать:

```
Config '<имя копии конфигурации VPN>' (version <версия ViPNet Coordinator HW>)
exists. Overwrite?
```

Выполните одно из действий:

- Чтобы подтвердить перезапись, введите символ `y` и нажмите клавишу **Enter**.
- Чтобы создать конфигурацию с другим именем, введите символ `n` и нажмите клавишу **Enter**.

4 Запустите демоны iplircfg, failoverd и mftpd с помощью команд:

```
hostname> iplir start
hostname> failover start
hostname> mftp start
```

Созданную копию конфигурации VPN вы можете использовать для восстановления настроек ViPNet Coordinator HW (см. «Восстановление настроек из копии конфигурации VPN» на стр. 33).

При необходимости вы также можете просмотреть список всех созданных копий конфигурации VPN (см. «Просмотр списка копий конфигурации VPN» на стр. 33).

Просмотр списка копий конфигурации VPN

Чтобы просмотреть список ранее созданных копий конфигурации VPN (см. «Создание копии конфигурации VPN» на стр. 31), выполните команду:

```
hostname# admin config list
```

В результате отобразится список, каждая строка которого содержит следующую информацию:

- имя;
- версия ViPNet Coordinator HW, в которой была создана копия конфигурации VPN (после слова `version`);
- вид копии конфигурации VPN — полная (`full`) или частичная (`part`);
- дата и время создания копии конфигурации VPN (после слова `saved`).

Если копия конфигурации VPN была создана автоматически, то ее имя начинается со слова `autosave` и содержит дату создания. Например: `autosave-2015-11-12`.

Если копия конфигурации VPN уже использовалась для восстановления настроек ViPNet Coordinator HW, дата и время использования отображаются после даты и времени создания копии конфигурации VPN (после слова `loaded`). Если копия конфигурации VPN ни разу не использовалась, после даты и времени создания будет указано `never loaded`.

Восстановление настроек из копии конфигурации VPN

Чтобы восстановить настройки ViPNet Coordinator HW из копии конфигурации VPN, выполните следующие действия:

1 Завершите работу всех демонов и драйверов ViPNet Coordinator HW с помощью команды:

```
hostname# vpn stop
```

2 Выполните команду:

```
hostname# admin config load <имя> [<версия>],
```

указав имя копии конфигурации VPN и версию ViPNet Coordinator HW, в которой она была создана. Если имя копии конфигурации VPN состоит из нескольких слов, заключите его в кавычки.

3 Если текущая версия ViPNet Coordinator HW выше версии ViPNet Coordinator HW, в которой была создана выбранная для восстановления настроек копия конфигурации VPN, подтвердите возврат к более ранней версии.

4 Будет предложено сохранить все текущие настройки ViPNet Coordinator HW:

```
Save current configuration [Y/n]?
```

Выполните одно из действий:

- Чтобы сохранить все текущие настройки, введите символ `Y` и нажмите клавишу **Enter**. Затем укажите имя соответствующей полной копии конфигурации VPN.
- Чтобы не сохранять текущие настройки ViPNet Coordinator HW, введите символ `n` и нажмите клавишу **Enter**. Для подтверждения действия введите фразу `Yes, do as I say`.

5 Дождитесь завершения процесса восстановления настроек ViPNet Coordinator HW.

6 Запустите демоны и драйверы ViPNet Coordinator HW с помощью команды:

```
hostname# vpn start
```

Теперь вы можете продолжить работу с ViPNet Coordinator HW.

Удаление копии конфигурации VPN

Чтобы удалить ненужные копии конфигурации VPN, выполните команду:

```
hostname# admin config delete <имя> [<версия>],
```

указав имя копии конфигурации VPN и версию ViPNet Coordinator HW, в которой она была создана.

Чтобы не писать имя копии конфигурации VPN полностью или удалить сразу несколько копий конфигурации VPN, имена которых содержат определенные символы, вместо имени укажите маску имени, используя символ «*».

Примечание. Если имя копии конфигурации VPN состоит из нескольких слов либо если для указания имени используется маска, заключите его в кавычки.

Например, чтобы удалить все копии конфигурации VPN, созданные автоматически в августе 2015 года, выполните команду:

```
hostname# admin config delete "rollback-2015-08-??-*"
```

Если несколько копий конфигурации VPN имеют имя, соответствующие указанной маске, то будет отображено количество таких копий конфигурации VPN:

```
<количество копий конфигурации VPN> configs match that pattern. Delete? (y/n)
```

Выполните одно из действий:

- Чтобы подтвердить удаление всех копий конфигурации VPN, введите символ `y` и нажмите клавишу **Enter**.
- Чтобы отменить удаление всех копий конфигурации VPN, введите символ `n` и нажмите клавишу **Enter**.

Чтобы удалить копии конфигурации VPN, созданные в определенной версии ViPNet Coordinator HW, укажите номер этой версии. Например:

```
hostname# admin config delete "rollback-2015-08-??-*" 4.1.0
```

Контроль целостности файлов ViPNet Coordinator HW

Во избежание возникновения сбоев ПО целостность критически важных файлов ViPNet Coordinator HW регулярно проверяется с помощью встроенных скриптов. Контролируются следующие компоненты:

- ядро Linux и образы модулей ViPNet Coordinator HW, используемые при загрузке ПО;
- системные демоны и библиотеки;
- конфигурационные файлы демонов ViPNet Coordinator HW;
- справочники и ключи ViPNet.

Указанные файлы защищены контрольными суммами, которые хранятся в файлах с расширением `*.crg`. Проверка целостности выполняется путем вычисления текущей контрольной суммы файла и ее сравнения с контрольной суммой из файла `*.crg`.

Каждый файл `*.crg` содержит также собственную контрольную сумму. Проверка контрольной суммы файла `*.crg` выполняется перед проверкой целостности остальных файлов, контрольная сумма которых в нем указана.

Проверка целостности файлов автоматически выполняется в следующих случаях:

- Установка ПО ViPNet Coordinator HW.
- Обновление ПО ViPNet Coordinator HW.
- Загрузка ПО ViPNet Coordinator HW.
- Периодическая проверка с интервалом 12 часов (в этом случае конфигурационные файлы ViPNet Coordinator HW, а также справочники и ключи ViPNet не проверяются).

Примечание. При каждой периодической проверке в системный журнал заносится запись, содержащая время и результаты этой проверки. Чтобы просмотреть информацию о последней периодической проверке, выполните команду:

```
hostname# admin show check integrity status
```

Кроме того, вы можете проверить целостность конфигурационных файлов вручную с помощью команды:

```
hostname# machine self-test
```

Примечание. Команду проверки целостности конфигурационных файлов нельзя использовать при удаленном администрировании с помощью SSH (подробнее см. в документе «ViPNet Coordinator HW. Общее описание»).

Нарушением целостности файлов также считается отсутствие хотя бы одного файла конфигурации или файла с контрольными суммами.

При несовпадении контрольных сумм в ходе периодической проверки ViPNet Coordinator HW автоматически перезагружается.

При несовпадении контрольных сумм в ходе проверки во время установки, обновления или загрузки ПО, а также в ходе проверки, выполняемой вручную, выводится сообщение о нарушении целостности файла. Затем выполняется одно из следующих действий:

- Если для искаженного файла имеется резервная копия (см. «Использование резервного раздела в исполнениях ViPNet Coordinator HW с двумя накопителями» на стр. 38), файл будет автоматически восстановлен. При этом будет выведено соответствующее сообщение. При восстановлении ViPNet Coordinator HW может быть перезагружен.
- Если резервная копия искаженного конфигурационного файла не найдена, будет выведено соответствующее сообщение. После этого демоны, отвечающие за работу в сети ViPNet, не будут запущены либо будут остановлены и сетевые интерфейсы будут выключены.

Совет. Если у вас есть файл экспорта справочников, ключей и настроек с расширением `*.vbe`, попробуйте восстановить систему, импортировав этот файл (подробнее см. в документе «ViPNet Coordinator HW. Подготовка к работе», в главе «Резервное копирование и восстановление настроек»). Если восстановить работу ViPNet Coordinator HW не удалось, обратитесь в службу поддержки ОАО «ИнфоТеКС».

- Если при загрузке обнаружено нарушение целостности ядра Linux или одного из образов модулей ViPNet Coordinator HW, и соответствующая резервная копия не найдена, то будет выведено соответствующее сообщение и загрузка ViPNet Coordinator HW завершится. Для выключения ViPNet Coordinator HW нажмите любую клавишу.

Использование резервного раздела в исполнениях ViPNet Coordinator HW с двумя накопителями

Чтобы обеспечить возможность оперативного восстановления критически важных компонентов ViPNet Coordinator HW, целостность которых регулярно проверяется (см. «Контроль целостности файлов ViPNet Coordinator HW» на стр. 36), такие файлы периодически копируются на специальный резервный раздел накопителя ViPNet Coordinator HW.

Примечание. Данная функция не предусмотрена в исполнениях ViPNet Coordinator HW с одним дисковым накопителем (HW 50 N1, N2, N3 и HW 100 X1).

Кроме того, данная функция доступна только на устройствах, на которых первоначально было установлено ПО ViPNet Coordinator HW версии не ниже 4.1.1. При обновлении ПО с более ранней версии данная функция не добавляется.

Копии файлов сохраняются в резервный раздел `/mnt/reserv` того накопителя ViPNet Coordinator HW, с которого не происходит загрузка ПО. При проверке целостности критически важных файлов искаженные файлы автоматически восстанавливаются из своих резервных копий.

Копирование файлов в резервный раздел выполняется в следующих случаях:

- Успешная проверка целостности файлов при первоначальной установке и обновлении ПО либо при выполнении команды `machine self-test`.
- Успешная проверка целостности при загрузке ViPNet Coordinator HW.
- Обновление справочников и ключей.
- Изменение пароля пользователя.

Примечание. В первом случае копируются все конфигурационные файлы, а в остальных — только часть файлов.

Если два ViPNet Coordinator HW работают в режиме кластера горячего резервирования (см. глоссарий, стр. 268), то резервные копии файлов регулярно копируются с активного сервера на пассивный. При изменении справочников и ключей на кластере, копии справочников и ключей обновляются как на активном, так и на пассивном сервере.

3

Настройка параметров безопасности

Изменение пароля пользователя

Пароль пользователя ViPNet Coordinator HW (см. глоссарий, стр. 270) назначается администратором сети ViPNet. После первичной установки справочников и ключей рекомендуется его изменить. Это дополнительно повысит безопасность, поскольку пароль не будет известен администратору сети ViPNet.

Чтобы изменить пароль пользователя, выполните следующие действия:

1 Выполните команду:

```
hostname# admin passwd
```

2 Введите текущий пароль пользователя или пароль администратора сетевого узла.

Примечание. При вводе паролей на экране ничего не отображается, введенные символы отредактировать нельзя.

---

3 Введите новый пароль пользователя и подтвердите его. Появится сообщение об успешном изменении пароля пользователя.

В результате с помощью нового пароля будут перешифрованы ключи пользователя, в том числе резервный набор персональных ключей при наличии.

Совет. Мы рекомендуем запомнить пароль, назначенный администратором сети ViPNet. Он может потребоваться вам при установке нового дистрибутива ключей (см. глоссарий, стр. 267) или РНПК, поскольку они будут защищены этим паролем и без него не смогут быть установлены на ViPNet Coordinator HW. Если вы забудете пароль, запросите его у администратора сети ViPNet вместе с дистрибутивом или РНПК.

Изменение способа аутентификации пользователя

Способ аутентификации (см. «Аутентификация пользователя на ViPNet Coordinator HW» на стр. 25) первоначально задается администратором сети в ПО ViPNet Удостоверяющий и ключевой центр. Информация о заданном способе аутентификации на ViPNet Coordinator HW попадает в составе справочников при первом развертывании дистрибутива ключей. Если требуется узнать, какой способ аутентификации используется на ViPNet Coordinator HW, введите команду:

```
hostname# iplir show authentication-type
```

В процессе работы с ViPNet Coordinator HW способ аутентификации может быть изменен администратором в командном интерпретаторе. Причем изменить способ можно только на «Устройство». Если вы используете способ аутентификации «Устройство», то изменить его на «Пароль» невозможно.

Изменение способа аутентификации на «Устройство» может потребоваться в том случае, если в целях дополнительной безопасности аутентификация пользователя на ViPNet Coordinator HW должна производиться с использованием устройства, на котором сохранен его персональный ключ.

Примечание. После изменения способа аутентификации на «Устройство» вы не сможете изменить его обратно на «Пароль».

Изменить способ аутентификации можно только при локальном подключении к ViPNet Coordinator HW. в удаленной SSH-сессии изменять способ аутентификации запрещено.

Чтобы изменить способ аутентификации на «Устройство», выполните следующие действия:

1 Обратитесь к администратору сети ViPNet с запросом изменить способ аутентификации пользователя ViPNet Coordinator HW на «Устройство».

Администратор сети после изменения способа аутентификации должен выслать на ViPNet Coordinator HW обновление справочников и ключей и предоставить вам внешнее устройство, на котором сохранен персональный ключ пользователя.

Внимание! В текущей версии ViPNet Coordinator HW для аутентификации могут использоваться только внешние устройства Рутокен Lite производства компании «Актив».

На устройстве не должны содержаться другие контейнеры ключей. При наличии других контейнеров на устройство не сможет быть найден персональный ключ пользователя, вследствие чего работа с ViPNet Coordinator HW будет невозможна.

2 Убедитесь, что на ViPNet Coordinator HW принято поступившее обновление справочников и ключей. Подробнее об этом см. в документе «ViPNet Coordinator HW. Подготовка к работе», в разделе «Обновление и удаление справочников и ключей».

3 В командном интерпретаторе выполните команду:

```
hostname# admin authentication-type-token
```

4 В ответ на предложение изменить способ аутентификации для усиления безопасности `In version 4.2, you can enhance security by two-factor authentication. As you switch to the two-factor authentication, you will not be able to return to the previous settings [Yes/No]` введите `Yes`.

5 Подключите к одному из USB-разъемов ViPNet Coordinator HW (или компьютера, на котором развернут виртуальный образ ViPNet Coordinator HW) внешнее устройство, на котором сохранен персональный ключ пользователя.

6 Введите ПИН-код доступа к подключенному устройству в ответ на сообщение `Enter PIN`.

7 Если введенный ПИН-код верен, появится сообщение об успешном изменении способа аутентификации.

Теперь для прохождения аутентификации на ViPNet Coordinator HW пользователю потребуется подключать данное устройство и вводить ПИН-код к нему.

Внимание! При работе ViPNet Coordinator HW в режиме кластера горячего резервирования (см. глоссарий, стр. 268) указанные настройки в командном интерпретаторе достаточно выполнить на активном сервере кластера. Эти настройки передаются с активного сервера на пассивный в ходе резервирования.

Просмотр информации об установленных ключах

При необходимости вы можете просмотреть информацию о следующих ключах, установленных на ViPNet Coordinator HW:

- персональный ключ пользователя;
- набор персональных ключей (РНПК);
- ключи узла (см. глоссарий, стр. 268).

Получение сведений об установленных ключах может требоваться вам перед добавлением РНПК на ViPNet Coordinator HW — чтобы убедиться, что РНПК отсутствует на узле (см. «Добавление резервного набора персональных ключей (РНПК)» на стр. 45). Остальная информация может быть востребована сотрудниками технического сопровождения при возникновении проблем с ключевой системой ViPNet Coordinator HW (например, после некорректного обновления справочников и ключей на узле или в других случаях).

Для просмотра информации о ключах выполните команду:

```
hostname# iplir show key-info
```

В результате выполнения команды отобразятся списки с информацией по типам ключей. Пример выводимой информации о ключах представлен на рисунке ниже.

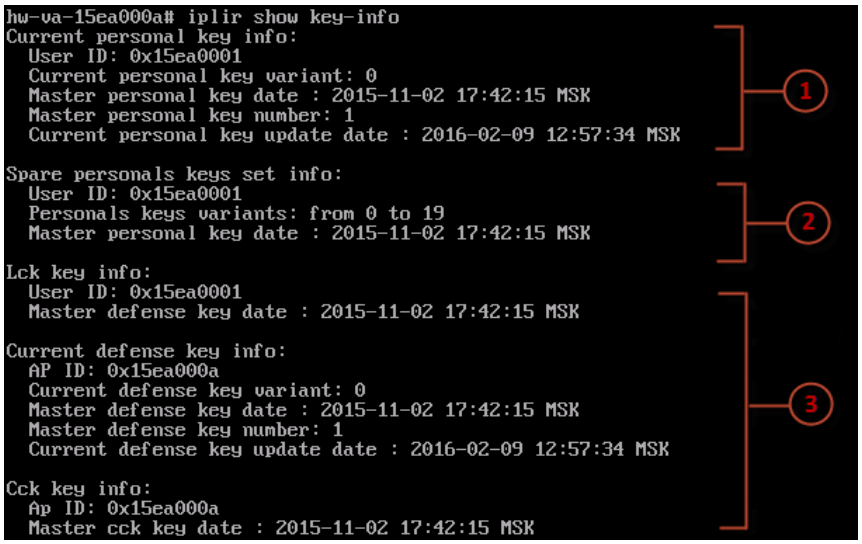

Рисунок 1: Просмотр информации о ключах

Цифрами на рисунке обозначены:

1 Информация о текущем персональном ключе пользователя.

2 Информация о РНПК.

Примечание. В этом списке может быть указано `Spare key set is not present`. Это означает, что РНПК отсутствует на узле. О том, как добавить РНПК на ViPNet Coordinator HW, см. в разделе Добавление резервного набора персональных ключей (РНПК) (на стр. 45).

3 Информация о ключах узла.

Добавление резервного набора персональных ключей (РНПК)

Когда требуется добавлять РНПК

В процессе эксплуатации сети ViPNet администратор может выполнять обновление ключей узлов в случае их компрометации и при смене мастер-ключа персональных ключей. В процессе данных операций меняется персональный ключ пользователя — берется новый ключ из резервного набора персональных ключей (РНПК) пользователя и с его помощью выполняется перешифрование всех остальных ключей. В итоге при централизованном обновлении ключи на узел отправляются зашифрованными на новом персональном ключе. Чтобы ключи могли быть приняты, на узле должен присутствовать РНПК. В противном случае расшифрование ключей будет невозможно, и их придется устанавливать вручную. Таким образом, РНПК, будучи заранее переданным пользователю, позволяет в указанных случаях дистанционно обновить ключи без необходимости высылать пользователю новый персональный ключ по скомпрометированному каналу.

Как правило, на ViPNet Coordinator HW файл с РНПК попадает в составе дистрибутива ключей (см. глоссарий, стр. 267). Но может возникнуть ситуация, когда его может не быть на узле. Например, если вы для восстановления работоспособности узла или обновления справочников и ключей повторно развернули дистрибутив ключей, в котором не было РНПК. В этом случае вы можете добавить файл с РНПК на ViPNet Coordinator HW вручную.

Примечание. В ViPNet Administrator версии 4.5 и ниже РНПК помещаются только в состав первого дистрибутива ключей. В составе последующих дистрибутивов ключей он отсутствует.

Начиная с ViPNet Administrator версии 4.6, РНПК помещается в состав каждого дистрибутива ключей, независимо от того, создается он впервые или повторно.

Порядок добавления РНПК

Для добавления резервного набора персональных ключей на сервер ViPNet Coordinator HW выполните следующие действия:

1 Перед добавлением убедитесь, что на ViPNet Coordinator HW отсутствует файл с РНПК (`*.pk`). Выполните просмотр информации об установленных ключах с помощью команды (см. «Добавление резервного набора персональных ключей (РНПК)» на стр. 45):

```
hostname# iplir show key-info
```

При отсутствии файла с РНПК на узле будет выдана следующая информация:

```
Spare key set is not present.
```

2 Запросите РНПК у администратора сети ViPNet. Администратор передаст вам файл с РНПК на отдельном внешнем устройстве лично в руки либо другим доверенным способом. Файл РНПК на устройстве будет защищен паролем пользователя ViPNet Coordinator HW, который задан в программе ViPNet Administrator.

Внимание! Если в процессе использования ViPNet Coordinator HW вы изменяли пароль пользователя (см. глоссарий, стр. 270) и не помните пароль, заданный администратором сети ViPNet в программе ViPNet Administrator, запросите его у администратора. Без данного пароля вы не сможете добавить файл с РНПК на узел.

3 Выполните команду:

```
hostname# admin add spare keys
```

4 Будет дополнительно выполнена проверка наличия файла с РНПК. Если файл отсутствует, будет автоматически предложено указать внешнее устройство, на котором содержится файл РНПК.

В зависимости от того, на каком внешнем устройстве вам был передан файл с РНПК, выполните следующие действия:

- подключите USB-носитель к USB-разъему ViPNet Coordinator HW или компьютера, на котором развернут виртуальный образ ViPNet Coordinator HW;
- вставьте CD-диск в оптический привод ViPNet Coordinator HW или компьютера, на котором развернут виртуальный образ ViPNet Coordinator HW.

После этого подтвердите подключение.

5 Будет произведен поиск файлов с РНПК и проверка идентификатора пользователя в каждом найденном файле:

- если подходящих файлов нет, появляется соответствующее сообщение;
- если файлы найдены, и идентификатор пользователя в файлах и на узле совпадает, то выводится список файлов для выбора.

6 Выберите нужный файл с РНПК. Для этого введите номер файла из предложенного списка и нажмите клавишу **Enter**.

7 Введите пароль пользователя ViPNet Coordinator HW, которым защищен файл с РНПК, и нажмите клавишу **Enter**. Пароль к РНПК будет отличаться от пароля пользователя, если вы меняли последний.

8 При успешном вводе пароля файл РНПК будет сохранен на ViPNet Coordinator HW и перешифрован на парольном ключе, отличном от пароля пользователя. Это сделано для того, чтобы в дальнейшем при обращении к РНПК не требовалось вводить пароль каждый раз.

Обновление ключей при истечении срока их действия

Персональный ключ пользователя, который входит в состав ключей пользователя, а также ключи защиты ключей обмена (см. глоссарий, стр. 268), которые входят в состав ключей узла, имеют ограниченный срок действия — 12 месяцев. В ViPNet Coordinator HW реализован механизм, который проверяет срок действия этих ключей с момента их установки на узел при первоначальном развертывании дистрибутива ключей или после обновления. За месяц, неделю и по истечении срока действия ключей производится специальное оповещение пользователя. В командном интерпретаторе и веб-интерфейсе появляются соответствующие сообщения. Кроме этого, информация об истечении ключей добавляется в журнал устранения неполадок.

Совет. Если требуется узнать, какого числа истекает срок действия персонального ключа или ключей защиты, вы можете просмотреть сведения об установленных ключах с помощью команды `iplir show key-info` (см. «Просмотр информации об установленных ключах» на стр. 43).

Как правило, персональный ключ и ключи защиты устанавливаются и обновляются на узле одновременно, поэтому даты начала и окончания срока их действия чаще всего будут совпадать.

При истечении срока действия ключей никакие ограничения на работу ViPNet Coordinator HW не накладываются. Но настоятельно рекомендуется обновить ключи, для чего следует обратиться к администратору сети ViPNet. Кроме того, перед обновлением ключей вы должны убедиться, что на узле присутствует резервный набор персональных ключей пользователя (РНПК). При отсутствии РНПК его следует добавить на узел. Подробнее см. раздел Добавление резервного набора персональных ключей (РНПК) (на стр. 45).

Внимание! При отсутствии РНПК на ViPNet Coordinator HW обновление ключей не сможет быть выполнено корректно. Кроме этого, по завершении такого обновления ключей потребуется повторная установка справочников и ключей для восстановления работоспособности ViPNet Coordinator HW.

Обновление ключей производится по следующему алгоритму:

1. После вашего обращения администратор сети должен выполнить следующие действия:
   - Поднять вариант ключей пользователя при обновлении персонального ключа, ключей узла — при обновлении ключей защиты ключей обмена.
   - После изменения варианта ключей сформировать и выслать обновления ключей на узел ViPNet Coordinator HW.
   - В случае обновления персонального ключа — перезаписать персональный ключ на внешнем устройстве, если пользователь ViPNet Coordinator HW производит аутентификацию с помощью устройства.

2 На ViPNet Coordinator HW вы должны принять поступившие обновления ключей и убедиться в их успешной установке.

4

Настройка подключения к сети

Настройка сетевых интерфейсов Ethernet

Сетевым интерфейсам Ethernet, установленным в системе, присваиваются имена `eth0`, `eth1` и так далее (по количеству таких интерфейсов в системе). Чтобы настроить подключение к сети, задайте параметры сетевых интерфейсов Ethernet. Для этого выполните следующие действия:

1. По умолчанию сетевому интерфейсу назначен класс `access` (см. глоссарий, стр. 268). При необходимости смените класс с помощью команды:

   ```
   hostname# inet ifconfig <имя интерфейса> class {access | trunk | slave}
   ```

   Если требуется, чтобы данный интерфейс Ethernet обрабатывал трафик из нескольких VLAN (см. «Организация обработки трафика из нескольких VLAN» на стр. 53), назначьте ему класс `trunk`.

   Если вы хотите объединить данный интерфейс Ethernet с другими в составе агрегированного интерфейса (см. «Использование агрегированных сетевых интерфейсов» на стр. 61), назначьте ему класс `slave`.

2. Выполните одно из действий:

   - Чтобы включить на сетевом интерфейсе режим автоматического получения параметров от DHCP-сервера, выполните команду:

     ```
     hostname# inet ifconfig <имя интерфейса> dhcp
     ```

     Также для режима DHCP вы можете задать дополнительные параметры:

     - включить или выключить автоматическое получение адресов DNS-серверов командой:

       ```
       hostname# inet ifconfig <имя интерфейса> dhcp dns {on | off}
       ```

     - включить или выключить автоматическое получение NTP-серверов командой:

       ```
       hostname# inet ifconfig <имя интерфейса> dhcp ntp {on | off}
       ```

     - включить или выключить автоматическое получение маршрутов от DHCP-сервера командой:

       ```
       hostname# inet ifconfig <имя интерфейса> dhcp route {on | off}
       ```

     По умолчанию всем дополнительным параметрам режима DHCP установлено значение `on`.

     Вы можете задать ряд параметров, которые будут присваиваться маршрутам DHCP-сервера (см. «Настройка параметров динамических маршрутов от DHCP/PPP-протокола» на стр. 190). При необходимости вы можете просмотреть параметры, которые заданы для режима DHCP на всех сетевых интерфейсах (см. «Просмотр настроек DHCP в режиме клиента» на стр. 193).

   - Чтобы присвоить сетевому интерфейсу статический IP-адрес, выполните команду:

     ```
     hostname# inet ifconfig <имя интерфейса> address <IP-адрес> netmask <маска сети>
     ```

Если до выполнения этой команды интерфейс работал в режиме DHCP, данные о DNS- и NTP-серверах, полученные по протоколу DHCP, будут потеряны.

3 Включите сетевой интерфейс с помощью команды:

```
hostname# inet ifconfig <имя интерфейса> up
```

Внимание! Максимальное количество интерфейсов в ViPNet Coordinator HW (включая физические, агрегированные, виртуальные, VLAN и localhost) не может превышать 128.

Назначение дополнительных IP-адресов

Назначение дополнительных IP-адресов (alias) для сетевых интерфейсов Ethernet ViPNet Coordinator HW удобно при развертывании сетей некоторых топологий. Например, с помощью дополнительных IP-адресов вы можете в рамках своей сети организовать логическую подсеть, узлам которой, в отличие от узлов остальной сети, будет разрешен доступ в Интернет.

Чтобы добавить дополнительный статический IP-адрес на сетевой интерфейс Ethernet, убедитесь, что на этом интерфейсе задан статический основной IP-адрес, и выполните команду:

```
hostname# inet ifconfig <имя интерфейса> address add <IP-адрес> netmask <маска сети>
```

Организация обработки трафика из нескольких VLAN

Вы можете использовать ViPNet Coordinator HW для обработки трафика в разветвленной сети, состоящей из нескольких независимых виртуальных локальных сетей VLAN (см. глоссарий, стр. 266). Это возможно благодаря поддержке виртуальных интерфейсов и тегированию (маркировке) трафика в соответствии со стандартом IEEE 802.1 Q.

Для организации обработки трафика из нескольких VLAN подключите один из сетевых интерфейсов ViPNet Coordinator HW (или компьютера, на котором развернут виртуальный образ ViPNet Coordinator HW) к коммутатору, объединяющему виртуальные сети, и выполните следующие действия:

1. Завершите работу управляющего демона с помощью команды:

   ```
   hostname> iplir stop
   ```

2. Измените класс интерфейса (см. глоссарий, стр. 268), к которому подключен коммутатор, с помощью команды:

   ```
   hostname# inet ifconfig <физический интерфейс> class trunk
   ```

3. Задайте номера виртуальных интерфейсов, которые будут соответствовать виртуальным сетям за коммутатором, с помощью следующих команд:

   ```
   hostname# inet ifconfig <физический интерфейс> vlan add <номер виртуального интерфейса>
   ```

4. Задайте параметры созданных виртуальных интерфейсов. Для этого выполните одно из действий:

   - Чтобы включить на сетевом интерфейсе режим автоматического получения параметров от DHCP-сервера, выполните команду:

     ```
     hostname# inet ifconfig <имя интерфейса> dhcp
     ```

     Также для режима DHCP вы можете задать дополнительные параметры:

     - включить или выключить автоматическое получение адресов DNS-серверов командой:

       ```
       hostname# inet ifconfig <имя интерфейса> dhcp dns {on | off}
       ```

     - включить или выключить автоматическое получение NTP-серверов командой:

       ```
       hostname# inet ifconfig <имя интерфейса> dhcp ntp {on | off}
       ```

     - включить или выключить автоматическое получение маршрутов от DHCP-сервера командой:

       ```
       hostname# inet ifconfig <имя интерфейса> dhcp route {on | off}
       ```

     По умолчанию всем дополнительным параметрам режима DHCP установлено значение `on`.

     Вы можете задать ряд параметров, которые будут присваиваться маршрутам DHCP-сервера (см. «Настройка параметров динамических маршрутов от DHCP/PPP-протокола» на стр. 190). При необходимости вы можете просмотреть параметры, которые заданы для

режима DHCP на всех сетевых интерфейсах (см. «Просмотр настроек DHCP в режиме клиента» на стр. 193).

- Чтобы присвоить сетевому интерфейсу статический IP-адрес, выполните команду:

  ```
  hostname# inet ifconfig <имя интерфейса> address <IP-адрес> netmask <маска сети>
  ```

  Если до выполнения этой команды интерфейс работал в режиме DHCP, данные о DNS- и NTP-серверах, полученные по протоколу DHCP, будут потеряны.

5 Для редактирования конфигурационного файла `iplir.conf` выполните команду:

```
hostname# iplir config
```

6 Добавьте секции `[adapter]`, описывающие созданные виртуальные интерфейсы (см. документ «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам», главу «Файл iplir.conf», «Секция [adapter]»). В каждой секции укажите следующие параметры:

```
name = <физический интерфейс>.<номер виртуального интерфейса>
type = internal
```

7 В секции `[adapter]` с описанием интерфейса, к которому подключен коммутатор, присвойте параметру `allowtraffic` значение `off`.

8 Чтобы сохранить изменения в конфигурационном файле, нажмите сочетание клавиш **Ctrl+O**. Затем нажмите клавишу **Enter**.

9 Чтобы закрыть файл, нажмите сочетание клавиш **Ctrl+X**.

10 Включите физический интерфейс, к которому подключен коммутатор, с помощью команды:

```
hostname# inet ifconfig <физический интерфейс> up
```

При этом автоматически будут включены созданные виртуальные интерфейсы.

11 Запустите управляющий демон с помощью команды:

```
hostname> iplir start
```

Теперь ViPNet Coordinator HW сможет обрабатывать трафик из виртуальных сетей на соответствующих виртуальных интерфейсах.

Пример организации обработки трафика из нескольких VLAN с помощью ViPNet Coordinator HW см. в документе «ViPNet Coordinator HW. Сценарии работы», в главе «Использование сервисных функций ViPNet Coordinator HW», в разделе «Организация обработки трафика из нескольких VLAN».

Внимание! Максимальное количество интерфейсов в ViPNet Coordinator HW (включая физические, агрегированные, виртуальные, VLAN и localhost) не может превышать 128.

Подключение к мобильной сети 3G, 4G

Внимание! Подключение к мобильной сети возможно только в исполнениях ViPNet Coordinator HW со встроенными 3G-модемами: ViPNet Coordinator HW50 A, B на аппаратной платформе HW50 N3 и ViPNet Coordinator HW100 A, B на аппаратной платформе HW100 N3.

Подключение к сети 3G, 4G выполняется по протоколу PPP. 3G-, 4G-модем представлен в операционной системе сетевым интерфейсом с именем `pppX`.

Для подключения к Интернету вы можете пользоваться услугами любого оператора мобильной связи. Для этого приобретите SIM-карту и установите ее в соответствующий слот ViPNet Coordinator HW. Если требуется, подключите необходимые услуги мобильного оператора. Подробную информацию об условиях подключения к Интернету можно получить у оператора мобильной связи.

Перед организацией подключения к мобильной сети 3G, 4G при необходимости выполните предварительную настройку ViPNet Coordinator HW (см. «Предварительная настройка ViPNet Coordinator HW» на стр. 55), а затем задайте параметры подключения к Интернету через 3G-, 4G-модем (см. «Настройка ViPNet Coordinator HW для работы с 3G-, 4G-модемом» на стр. 56).

Предварительная настройка ViPNet Coordinator HW

Перед организацией подключения к сети 3G, 4G при необходимости выполните следующие действия:

- По умолчанию при первом подключении модема к сети 3G, 4G на ViPNet Coordinator HW добавляется новый маршрут по умолчанию, в котором в качестве шлюза указан адрес узла сети вашего мобильного оператора. Этот адрес 3G-, 4G-модем получает при подключении автоматически. Первоначально для нового маршрута задана метрика по умолчанию (см. «Изменение метрики по умолчанию для маршрутов от DHCP/PPP-протокола» на стр. 193). Если вы хотите задать для маршрута другое значение метрики, выполните команду:

  ```
  hostname# inet usb-modem set route-metric <метрика>,
  ```

  указав в параметре `<метрика>` число из интервала 1–255.

- Если вы не хотите добавлять маршрут по умолчанию, в котором используется шлюз, получаемый при первом подключении модема к сети 3G, 4G, выполните команду:

  ```
  hostname# inet usb-modem set route off
  ```

- При необходимости вы можете также отменить добавление адреса DNS-сервера, получаемого от вашего мобильного оператора при первом подключении 3G-, 4G-модема. Для этого выполните команду:

  ```
  hostname# inet usb-modem set dns off
  ```

После выполнения всех необходимых предварительных настроек задайте параметры подключения к Интернету через 3G-, 4G-модем (см. «Настройка ViPNet Coordinator HW для работы с 3G-, 4G-модемом» на стр. 56).

Настройка ViPNet Coordinator HW для работы с 3G-, 4G-модемом

Для подключения к мобильной сети 3G, 4G после выполнения необходимых предварительных настроек (см. «Предварительная настройка ViPNet Coordinator HW» на стр. 55) выполните следующие действия:

1. Задайте настройки для подключения к сети вашего мобильного оператора:
   - Настройки для подключения к сетям мобильных операторов МТС, Билайн, Мегафон или SkyLink предустановлены в ViPNet Coordinator HW, поэтому, если вы будете использовать услуги одного из этих мобильных операторов, укажите это с помощью команды:

     ```
     hostname# inet usb-modem set provider {mts | beeline | megafon | skylink}
     ```

   - Если вы будете пользоваться услугами другого мобильного оператора, настройте параметры 3G-, 4G-подключения. Для этого выполните следующие действия:
     - Узнайте у вашего мобильного оператора телефонный номер точки доступа в Интернет, а также имя пользователя и пароль, необходимые для авторизации.
     - Задайте имя нового оператора с помощью команды:

       ```
       hostname# inet usb-modem add provider <имя оператора>
       ```

     - Укажите IP-адрес или DNS-имя точки доступа в Интернет с помощью команды:

       ```
       hostname# inet usb-modem set connection address {<IP-адрес> | <DNS-имя>}
       ```

     - Задайте номер телефона (в формате USSD-запроса) точки доступа в Интернет с помощью команды:

       ```
       hostname# inet usb-modem set phone <номер телефона>
       ```

     - При необходимости укажите имя пользователя с помощью команды:

       ```
       hostname# inet usb-modem set user <имя пользователя>
       ```

     - При необходимости укажите пароль для подключения с помощью команды:

       ```
       hostname# inet usb-modem set password <пароль>
       ```

     - Укажите, что для подключения к сети следует использовать параметры вновь заданного оператора с помощью команды:

       ```
       hostname# inet usb-modem set provider <имя оператора>
       ```

Например:

```
hostname# inet usb-modem add provider xtelecom
hostname# inet usb-modem set connection address internet.xtelecom.it
hostname# inet usb-modem set phone *99***1#
hostname# inet usb-modem set user xtelecom
hostname# inet usb-modem set password xtelecom
hostname# inet usb-modem set pin 1234
hostname# inet usb-modem set provider xtelecom
```

2 Если ваша SIM-карта защищена ПИН-кодом, укажите этот ПИН-код с помощью команды:

```
hostname# inet usb-modem set pin <ПИН-код>
```

Примечание. Вы можете отменить использование ПИН-кода с помощью команды:
`hostname# inet usb-modem reset pin`

3 Для проверки текущих параметров подключения к сети 3G, 4G выполните команду:

```
hostname> inet show usb-modem
```

4 Включите 3G-, 4G-модем с помощью команды:

```
hostname# inet usb-modem mode on
```

Примечание. При использовании некоторых моделей 3G-, 4G-модемов подключение к Интернету может занять несколько минут.

В результате подключение к мобильной 3G-, 4G-сети будет установлено.

Подключение к сети Wi-Fi

Внимание! Подключение к сети Wi-Fi возможно только в исполнениях ViPNet Coordinator HW со встроенными адаптерами Wi-Fi: ViPNet Coordinator HW50 A, B на аппаратной платформе HW50 N2 и ViPNet Coordinator HW100 A, B на аппаратной платформе HW100 N2.

Адаптер Wi-Fi отображается как сетевой интерфейс с именем `wlan0`.

Все настройки подключения в сети Wi-Fi должны выполняться при выключенном сетевом интерфейсе `wlan0`. Выключить интерфейс можно с помощью команды:

```
hostname# inet wifi mode off
```

Примечание. ViPNet Coordinator HW может также работать в качестве точки доступа к сети Wi-Fi (см. «Настройка параметров точки доступа к сети Wi-Fi» на стр. 175). Использование ViPNet Coordinator HW одновременно в качестве клиента и точки доступа Wi-Fi не поддерживается.

Чтобы подключить ViPNet Coordinator HW к сети Wi-Fi, выполните следующие действия:

1. Для просмотра списка доступных сетей Wi-Fi выполните команду:

   ```
   hostname> inet wifi scan
   ```

   Примечание. ViPNet Coordinator HW можно подключить только к сети Wi-Fi, которая присутствует в списке доступных сетей. К скрытым Wi-Fi-сетям подключение ViPNet Coordinator HW невозможно.

2. Укажите параметры доступа к сети Wi-Fi, к которой вы хотите подключиться. Для этого используйте одну из следующих команд:

   - Если аутентификация в сети не требуется, выполните команду:

     ```
     hostname# inet wifi client authentication open
     ```

     Затем по запросу введите имя сети.

   - Если для аутентификации используется режим WPA-PSK или WPA2-PSK, выполните команду:

     ```
     hostname# inet wifi client authentication {wpa-psk | wpa2-psk}
     ```

     Затем по запросу введите имя сети и пароль. Пароль можно узнать у администратора сети Wi-Fi, к которой вы подключаетесь.

3. Сетевой интерфейс `wlan0` всегда работает в режиме автоматического получения параметров от DHCP-сервера, при этом вы можете задать следующие дополнительные параметры:

   - включить или выключить автоматическое получение адресов DNS-серверов командой:

     ```
     hostname# inet ifconfig <имя интерфейса> dhcp dns {on | off}
     ```

- включить или выключить автоматическое получение NTP-серверов командой:

  ```
  hostname# inet ifconfig <имя интерфейса> dhcp ntp {on | off}
  ```

- включить или выключить автоматическое получение маршрутов от DHCP-сервера командой:

  ```
  hostname# inet ifconfig <имя интерфейса> dhcp route {on | off}
  ```

По умолчанию всем указанным дополнительным параметрам установлено значение `on`.

Вы можете задать ряд параметров, которые будут присваиваться маршрутам DHCP-сервера (см. «Настройка параметров динамических маршрутов от DHCP/PPP-протокола» на стр. 190). При необходимости вы можете просмотреть параметры, которые заданы для режима DHCP на всех сетевых интерфейсах (см. «Просмотр настроек DHCP в режиме клиента» на стр. 193).

4 Включите сетевой интерфейс `wlan0` с помощью команды:

```
hostname# inet wifi mode on
```

5 Для проверки параметров подключения к сети Wi-Fi выполните команду:

```
hostname> inet show wifi
```

Использование динамических интерфейсов

При подключении к мобильной сети 3G или 4G (см. «Подключение к мобильной сети 3G, 4G» на стр. 55) или беспроводной сети Wi-Fi (см. «Подключение к сети Wi-Fi» на стр. 58) на ViPNet Coordinator HW автоматически добавляются и удаляются динамические интерфейсы (см. глоссарий, стр. 267). По умолчанию разрешена работа динамических интерфейсов, которые входят в одну из следующих групп:

- `ppp` — группа интерфейсов для подключения к мобильной сети через встроенный модем;
- `wifi` — группа интерфейсов для подключения к беспроводной сети Wi-Fi.

Если созданный динамический интерфейс входит в одну из перечисленных групп, он автоматически становится активным и начинает пропускать IP-трафик. Дополнительные настройки динамических интерфейсов, аналогичные настройкам статических интерфейсов через файл `iplir.conf`, не требуются. Для каждой группы динамических интерфейсов на ViPNet Coordinator HW создан отдельный журнал IP-пакетов (файл `iplir.conf-<интерфейс или группа интерфейсов>`), в котором регистрируются записи обо всех IP-пакетах, разрешенных и заблокированных на интерфейсах этой группы.

Вы можете просмотреть список активных динамических сетевых интерфейсов с помощью команды:

```
hostname> iplir show adapters groups
```

Активные динамические интерфейсы также отображаются в списке интерфейсов при выполнении команды:

```
hostname> iplir show adapters
```

Если требуется изменить настройки журнала IP-пакетов для группы интерфейсов, выполните команду:

```
hostname# iplir config <группа интерфейсов>
```

Использование агрегированных сетевых интерфейсов

Если пропускной способности отдельных сетевых интерфейсов ViPNet Coordinator HW недостаточно для ваших задач или если требуется повысить надежность ваших каналов передачи данных, вы можете объединить несколько физических сетевых интерфейсов ViPNet Coordinator HW в один логический — агрегированный интерфейс (см. глоссарий, стр. 266). При этом соответствующие каналы связи объединяются на канальном уровне сетевой модели OSI.

Например, если вам нужен канал связи с пропускной способностью выше 1 Гбит/с, а на вашем исполнении ViPNet Coordinator HW есть только гигабитные интерфейсы, вы можете объединить два или три из них в один агрегированный. При этом даже если один из объединенных интерфейсов выйдет из строя, агрегированный канал продолжит работать.

Кроме того, вы можете использовать агрегированный интерфейс только для резервирования канала связи, без увеличения его пропускной способности. В этом случае весь трафик будет передаваться через один из подчиненных физических интерфейсов, остальные же начинают работать только при его сбое.

Агрегированные каналы целесообразно использовать в отказоустойчивых сетях, по каналам которых передаются большие объемы данных, например в сетях центров обработки данных.

Чтобы задать, как будет распределяться нагрузка по подчиненным физическим интерфейсам, необходимо выбрать один из режимов работы агрегированного интерфейса (см. «Режимы работы агрегированного интерфейса» на стр. 63).

Создание агрегированного интерфейса

Чтобы добавить новый агрегированный сетевой интерфейс, выполните следующие действия:

1. Выберите физические интерфейсы ViPNet Coordinator HW, которые вы хотите объединить, и установите для них класс `slave` (см. глоссарий, стр. 268) с помощью команды:

   ```
   hostname# inet ifconfig <имя интерфейса> class slave
   ```

2. Выберите режим работы создаваемого агрегированного интерфейса (см. «Режимы работы агрегированного интерфейса» на стр. 63), затем создайте агрегированный канал с помощью команды:

   ```
   hostname# inet bonding add <номер> mode <режим> slaves <интерфейс 1>
   [<интерфейс 2>] [<интерфейс 3>]
   ```
   , где:

   - `<номер>` — номер агрегированного интерфейса. Вы можете создать до трех агрегированных интерфейсов с номерами 0, 1 или 2. В результате выполнения команды создается агрегированный интерфейс с именем `bond<номер>`.

- `<режим>` — режим работы агрегированного интерфейса.
- `<интерфейс 1>`, `<интерфейс 2>`, `<интерфейс 3>` — физические интерфейсы, подчиненные создаваемому агрегированному интерфейсу. Вы можете задать до трех подчиненных интерфейсов. При создании агрегированного интерфейса необходимо задать не менее одного подчиненного интерфейса. В дальнейшем вы можете добавлять подчиненные интерфейсы с помощью команды `inet ifconfig bonding add` и удалять подчиненные интерфейсы с помощью команды `inet ifconfig bonding delete`.

3 По умолчанию соединение на подчиненных физических интерфейсах проверяется каждые 0,1 секунды. Вы можете изменить это значение и задать частоту от 1 до 1000 миллисекунд с помощью следующей команды:

```
hostname# inet ifconfig <имя агрегированного интерфейса> bonding miimon <частота в миллисекундах>
```

4 Для режимов, в которых это требуется, задайте также дополнительные параметры с помощью команд, указанных в разделе Режимы работы агрегированного интерфейса (на стр. 63).

5 Выполните одно из действий:

- Чтобы включить на агрегированном сетевом интерфейсе режим автоматического получения параметров от DHCP-сервера, выполните команду:

  ```
  hostname# inet ifconfig <имя агрегированного интерфейса> dhcp
  ```

  Также для режима DHCP вы можете задать дополнительные параметры:

  - включить или выключить автоматическое получение адресов DNS-серверов командой:

    ```
    hostname# inet ifconfig <имя агрегированного интерфейса> dhcp dns {on | off}
    ```

  - включить или выключить автоматическое получение адресов NTP-серверов командой:

    ```
    hostname# inet ifconfig <имя агрегированного интерфейса> dhcp ntp {on | off}
    ```

  - включить или выключить автоматическое получение маршрутов от DHCP-сервера командой:

    ```
    hostname# inet ifconfig <имя агрегированного интерфейса> dhcp route {on | off}
    ```

  По умолчанию всем дополнительным параметрам режима DHCP установлено значение `on`.

  Вы можете задать ряд параметров, которые будут присваиваться маршрутам DHCP-сервера (см. «Настройка параметров динамических маршрутов от DHCP/PPP-протокола» на стр. 190). При необходимости вы можете просмотреть параметры, которые заданы для режима DHCP на всех сетевых интерфейсах (см. «Просмотр настроек DHCP в режиме клиента» на стр. 193).

- Чтобы присвоить агрегированному сетевому интерфейсу статический IP-адрес, выполните команду:

  ```
  hostname# inet ifconfig <имя агрегированного интерфейса> address <IP-адрес> netmask <маска сети>
  ```

  Если до выполнения этой команды агрегированный интерфейс работал в режиме DHCP, данные о DNS- и NTP-серверах, полученные про протоколу DHCP, будут потеряны.

6 Отредактируйте файл конфигурации `iplir.conf`. Для этого выполните следующие действия:

6.1 Остановите управляющий демон с помощью команды:

```
hostname> iplir stop
```

6.2 Чтобы открыть файл конфигурации `iplir.conf` для редактирования, выполните команду:

```
hostname# iplir config
```

6.3 Добавьте секцию `[adapter]`, описывающую созданный агрегированный интерфейс (см. документ «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам», главу «Файл iplir.conf»):

```
[adapter]
name= bond<номер агрегированного интерфейса>
allowtraffic= on
type= internal
```

6.4 Запустите управляющий демон с помощью команды:

```
hostname> iplir start
```

7 По умолчанию созданному интерфейсу назначается класс `access`. Если вы хотите, чтобы интерфейс обрабатывал трафик из нескольких VLAN, назначьте ему класс `trunk` (см. «Организация обработки трафика из нескольких VLAN» на стр. 53).

8 Последовательно включите все подчиненные физические интерфейсы с помощью команды:

```
hostname# inet ifconfig <подчиненный интерфейс> up
```

9 Включите агрегированный сетевой интерфейс с помощью команды:

```
hostname# inet ifconfig <агрегированный интерфейс> up
```

В результате будет создан агрегированный интерфейс с именем `bond<номер>`. В дальнейшем вы можете работать с агрегированным интерфейсом так же, как с обычным физическим.

Внимание! Максимальное количество интерфейсов в ViPNet Coordinator HW (включая физические, агрегированные, виртуальные, VLAN и localhost) не может превышать 128.

Режимы работы агрегированного интерфейса

При создании агрегированного интерфейса необходимо указать режим его работы, наиболее подходящий для решения ваших задач. В ViPNet Coordinator HW предусмотрено несколько режимов, позволяющих по-разному распределять нагрузку между подчиненными интерфейсами. Эти режимы и их параметры описаны в таблице ниже.

Таблица 6. Режимы работы агрегированного интерфейса

Режим	Описание
`balance-rr`	Режим, подходящий как для балансировки нагрузки на подчиненных интерфейсах, так и для защиты от сбоев. Может применяться в сетях с простой топологией. В этом режиме исходящие пакеты, попадающие на агрегированный интерфейс, отправляются через подчиненные физические интерфейсы поочередно: первый пакет отправляется через один подчиненный интерфейс, второй пакет — через следующий подчиненный интерфейс и так далее.
`balance-xor`	Режим, предназначенный для защиты от сбоев и распределения нагрузки таким образом, чтобы пакеты от одного и того же отправителя к одному и тому же получателю всегда отправлялись через один и тот же подчиненный интерфейс. В этом режиме для определения подчиненного физического интерфейса, через который отправляется пакет, используется специальная хэш-функция, алгоритм вычисления которой можно задать с помощью команды: `inet ifconfig <имя агрегированного интерфейса> bonding xmit-hash-policy <алгоритм>` Вы можете задать один из следующих алгоритмов: • `layer2` — в алгоритме используются MAC-адреса отправителя и получателя пакета, таким образом, одинаковыми считаются сетевые узлы с одинаковыми MAC-адресами; • `layer2+3` — в алгоритме используются MAC-адреса отправителя и получателя, а также IP-адреса отправителя и получателя (для протокола IPv4), таким образом, одинаковыми считаются сетевые узлы с одинаковыми MAC-адресами и IP-адресами; • `layer3+4` — в алгоритме используются IP-адреса отправителя и получателя, а также номера портов TCP и UDP (при наличии), таким образом, одинаковыми считаются сетевые узлы с одинаковыми MAC-адресами, IP-адресами, а также портами TCP или UDP.
`balance-tlb`	Режим, предназначенный для балансировки нагрузки на подчиненных интерфейсах и рекомендуемый к использованию при передаче большого числа пакетов разного размера, когда более простые режимы не распределяют нагрузку равномерно. В этом режиме ведется подсчет размера исходящих пакетов, переданных через каждый из подчиненных физических интерфейсов, и на основе этого выполняется выбор интерфейса, через который будет передан пакет, попавший на агрегированный интерфейс. **Примечание.** Для работы агрегированного интерфейса в режиме `balance-tlb` необходимо, чтобы все подчиненные физические интерфейсы были подключены к сети через коммутатор.

Режим	Описание

`802.3ad`	Режим динамического агрегирования с использованием протокола LACP, предназначен для комплексной балансировки нагрузки. В этом режиме агрегированный интерфейс работает следующим образом:
	• Среди подчиненных физических интерфейсов формируются группы — «агрегаторы», скорость передачи данных на интерфейсах которых одинакова (например, группа гигабитных интерфейсов и группа 10-гигабитных интерфейсов).
	• Один из агрегаторов выбирается активным в соответствии с алгоритмом, задаваемым с помощью команды:
	`inet ifconfig <интерфейс> bonding ad-select <алгоритм>`
	Вы можете выбрать один из приведенных ниже алгоритмов:
	`stable` — алгоритм, при котором первоначально выбирается агрегатор с наибольшей суммарной пропускной способностью подчиненных физических интерфейсов, а в дальнейшем выбор нового агрегатора выполняется только в случае сбоя всех подчиненных интерфейсов текущего агрегатора.
	`bandwidth` — режим, при котором первоначально выбирается агрегатор с наибольшей пропускной способностью подчиненных физических интерфейсов, а в дальнейшем, при добавлении, удалении или сбое подчиненных физических интерфейсов, в агрегаторах производится перегруппировка подчиненных физических интерфейсов и выполняется выбор нового агрегатора.
	`count` — режим, при котором первоначально выбирается агрегатор с наибольшим количеством подчиненных физических интерфейсов, а в дальнейшем, при добавлении, удалении или сбое подчиненных физических интерфейсов, в агрегаторах производится перегруппировка подчиненных физических интерфейсов и выполняется выбор нового агрегатора.
	• Внутри агрегатора подчиненный физический интерфейс, через который отправляются исходящие пакеты, выбирается аналогично режиму `balance-xor`.
	• С другим сетевым оборудованием происходит обмен пакетами LACP с периодичностью, задаваемой с помощью команды:
	`inet ifconfig <интерфейс> bonding lacp-rate <slow | fast>`
	В случае выбора параметра `slow` обмен пакетами по протоколу LACP выполняется каждые 30 секунд, в случае выбора параметра `fast` — каждую секунду.
	Обмен пакетами позволяет определить сбой подчиненного интерфейса даже в том случае, если этот интерфейс подключен к другому сетевому узлу не напрямую.

`active-backup`	Режим, предназначенный для защиты от сбоев, но не для балансировки нагрузки на подчиненных физических интерфейсах. В этом режиме один из подчиненных физических интерфейсов назначается основным (автоматически или явно с помощью команды `inet ifconfig bonding primary`), и все исходящие пакеты отправляются через него. При этом, в случае сбоя на основном подчиненном интерфейсе пакеты будут отправляться через другие подчиненные интерфейсы.

Режим	Описание
`broadcast`	Режим предоставляет наибольшую защиту от сбоев. В этом режиме пакеты, попадающие на агрегированный интерфейс, отправляются через все подчиненные физические интерфейсы одновременно.

Примечание. Если в процессе функционирования агрегированного канала вы измените режим работы агрегированного интерфейса, возможно кратковременное пропадание соединения (до 1 секунды).

5

Настройка VPN

Общие принципы настройки VPN

Настройка параметров VPN производится путем редактирования конфигурационного файла `iplir.conf`. Большинство параметров в этом файле задается автоматически при установке справочников и ключей на узле — параметрам присваиваются значения, заданные администратором при настройке сети ViPNet и отдельных сетевых узлов в программе ViPNet Центр управления сетью (ЦУС) (см. глоссарий, стр. 265).

Помимо настройки режимов работы узла через межсетевой экран (см. «Настройка режимов работы через межсетевой экран» на стр. 69), с помощью параметров файла `iplir.conf` вы можете выполнить следующие настройки VPN:

- Настройка параметров видимости узлов (на стр. 77).
- Настройка параметров виртуальных адресов (на стр. 79).
- Настройка туннелируемых адресов (на стр. 81).
- Настройка IP-адресов доступа к узлу и их приоритета (на стр. 86).

Настройка режимов работы через межсетевой экран

Если ViPNet Coordinator HW работает во внутренней сети с частными IP-адресами и на границе этой сети установлен межсетевой экран или другое устройство, осуществляющее трансляцию адресов, то для нормальной работы с другими узлами, находящимися во внешней сети или стоящими за другими межсетевыми экранами, необходимо настроить один из режимов работы через межсетевой экран (см. «Режимы подключения к сети через межсетевой экран» на стр. 69).

Настройка режима выполняется путем редактирования файла конфигурации `iplir.conf`. Перед редактированием этого файла необходимо завершить работу управляющего демона командой `iplir stop`, а после окончания редактирования, чтобы все изменения вступили в силу, — вновь запустить его командой `iplir start`.

Режимы подключения к сети через межсетевой экран

Чтобы обеспечить возможность подключения ViPNet Coordinator HW к внешней сети при использовании различных типов межсетевого экрана, в программном обеспечении ViPNet предусмотрено четыре режима работы через межсетевой экран:

1. «Без использования межсетевого экрана» — ViPNet Coordinator HW имеет прямое подключение к внешней сети.
2. «Координатор» — подключение ViPNet Coordinator HW через другой координатор (при каскадной схеме установки координаторов).
3. «Со статической трансляцией адресов» — подключение через внешний межсетевой экран (устройство) с трансляцией адресов (NAT), на котором возможна настройка статических правил трансляции адресов.
4. «С динамической трансляцией адресов» — подключение через внешний межсетевой экран (устройство), на котором осуществляется динамическая трансляция адресов (наиболее распространенный способ подключения к WAN).

Если ViPNet Coordinator HW имеет непосредственное подключение к внешней сети, то есть может быть доступен напрямую со стороны любых других сетевых узлов (например, имеет публичный адрес), то для него следует выбрать режим «Без использования межсетевого экрана».

Если ViPNet Coordinator HW имеет частный IP-адрес, который может быть недоступен для других сетевых узлов в соответствии с общими правилами маршрутизации (то есть на выходе во внешнюю сеть установлен межсетевой экран или иное устройство, выполняющее преобразование адресов), то необходимо выбрать один из режимов подключения к сети через межсетевой экран.

Режим «Координатор» выбирается в случае, если на выходе во внешнюю сеть уже установлен другой координатор, выполняющий функции межсетевого экрана. В этом случае установленный координатор служит межсетевым экраном для ViPNet Coordinator HW.

Режим «Со статической трансляцией адресов» выбирается в случае, если на выходе во внешнюю сеть установлен межсетевой экран или иное NAT-устройство, на котором можно настроить статические правила трансляции адресов, обеспечивающие взаимодействие с определенным внутренним адресом сети по протоколу UDP с заданным портом. Данный режим задан по умолчанию.

Также режим «Со статической трансляцией адресов» следует использовать, если в локальной сети имеется несколько координаторов, связанных между собой как по внутренней, так и по внешней сети, и к ним необходим доступ из внешней сети со стороны удаленных узлов, работающих в режиме динамической трансляции адресов. Если же на координаторах будет установлен режим «Без использования межсетевого экрана», то для удаленного узла, использующего один из этих координаторов как сервер соединений, в качестве адреса доступа к другому узлу может зарегистрироваться некорректный адрес (при этом доступ к нему будет потерян).

Режим «С динамической трансляцией адресов» выбирается в случае, если на выходе во внешнюю сеть установлен межсетевой экран или иное NAT-устройство, на котором затруднительно настроить статические правила трансляции адресов. Данный режим наиболее универсален и ViPNet Coordinator HW в этом режиме будет работоспособен и при других способах подключения к сети.

Примечание. Если узлы находятся в одной локальной сети и могут обмениваться широковещательными пакетами, то независимо от выбранного типа межсетевого экрана взаимодействие между ними всегда осуществляется напрямую по IP-адресу узла.

Настройка режима «Без использования межсетевого экрана»

Для настройки работы узла ViPNet Coordinator HW без использования межсетевого экрана выполните следующие действия:

1. Завершите работу управляющего демона с помощью команды:

   ```
   hostname> iplir stop
   ```

2. Откройте файл `iplir.conf` для редактирования с помощью команды:

   ```
   hostname# iplir config
   ```

3. В собственной секции `[id]` установите параметр `usefirewall` в значение `off`.
4. В секции `[dynamic]` установите параметр `dynamic_proxy` в значение `off`.
5. Для всех используемых сетевых интерфейсов в секциях `[adapter]` установите параметр `type` в значение `internal`.

6 Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

```
hostname> iplir start
```

Настройка режима «Координатор»

ViPNet Coordinator HW может устанавливаться за координатор, только если между ними нет никаких межсетевых экранов. При этом по умолчанию один из сетевых интерфейсов этого координатора доступен ViPNet Coordinator HW по реальному IP-адресу.

Для настройки работы ViPNet Coordinator HW через координатор выполните следующие действия:

1 Завершите работу управляющего демона с помощью команды:

```
hostname> iplir stop
```

2 Откройте файл `iplir.conf` для редактирования с помощью команды:

```
hostname# iplir config
```

3 В секции `[id]` выбранного координатора:

- в параметре `ip` укажите любой реальный IP-адрес этого координатора, доступный данному узлу, если он не указан;
- в параметре `port` укажите номер порта назначения, на который будут посылаться пакеты для выбранного координатора.

4 В собственной секции `[id]` установите параметр `usefirewall` в значение `on` и в параметре `proxyid` укажите идентификатор выбранного координатора.

5 В секции `[dynamic]` установите параметр `dynamic_proxy` в значение `off`.

6 В секции `[adapter]` сетевого интерфейса, со стороны которого находится выбранный координатор, установите параметр `type` в значение `external`.

7 Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

```
hostname> iplir start
```

После соединения с координатором и при его правильной настройке в секции `[id]` этого координатора будут установлены значения параметров `firewallip`, `port` и `proxyid`. Кроме того, в соответствии со значениями этих параметров координатора могут измениться значения параметров `firewallip` и `port` собственной секции `[id]`.

Настройка режима «Со статической трансляцией адресов»

Если ViPNet Coordinator HW установлен за межсетевым экраном (устройством) с трансляцией адресов (NAT), на котором можно настроить статические правила NAT, то на ViPNet Coordinator HW нужно произвести настройки режима работы со статическим NAT.

В этом случае IP-адрес ViPNet Coordinator HW и порт доступа к нему должны быть жестко заданы на межсетевом экране. Кроме того, на ViPNet Coordinator HW необходимо настроить маршрутизацию на внешний межсетевой экран (шлюз по умолчанию или маршруты для удаленных подсетей).

На межсетевом экране (или NAT-устройстве) должны быть заданы следующие статические правила NAT:

- Пропускать исходящие UDP-пакеты с IP-адресом и портом ViPNet Coordinator HW (порт источника) на любой внешний адрес и порт (с подменой адреса источника на внешний адрес NAT-устройства).
- Пропускать и перенаправлять входящие UDP-пакеты с портом назначения, заданным в собственной секции `[id]` ViPNet Coordinator HW, на IP-адрес ViPNet Coordinator HW.

Для настройки подключения ViPNet Coordinator HW к сети через межсетевой экран со статическим NAT выполните следующие действия:

1. Завершите работу управляющего демона с помощью команды:

   ```
   hostname> iplir stop
   ```

2. Откройте файл `iplir.conf` для редактирования с помощью команды:

   ```
   hostname# iplir config
   ```

3. В собственной секции `[id]` установите следующие параметры:
   - параметр `usefirewall` в значение `on`;
   - параметр `proxyid` в значение `0`;
   - параметр `fixfirewall` в значение:
     - `on` — если на внешнем межсетевом экране с NAT есть возможность настроить статические правила только для входящих пакетов, предназначенных узлу ViPNet Coordinator HW, то есть обеспечить пропуск пакетов, имеющих заданные адрес и порт назначения, а также их перенаправление на адрес узла ViPNet Coordinator HW. В этом случае внешний IP-адрес межсетевого экрана с NAT и порт доступа к нему могут быть жестко заданы на межсетевом экране, и их необходимо указать в параметрах `firewallip` и `port` собственной секции `[id]`.
     - `off` — если внешний IP-адрес межсетевого экрана с NAT в процессе работы может меняться. В этом случае на других узлах IP-адрес доступа к узлу ViPNet Coordinator HW будет регистрироваться по внешним параметрам пакета. Параметр `firewallip` в

данном режиме определяется автоматически по информации, полученной от узлов, находящихся во внешней сети, поэтому редактировать его вручную не следует.

4. Если вы установили параметр `fixfirewall` в значение `on`, то в секции `[adapter]` сетевого интерфейса, со стороны которого находится внешний межсетевой экран, установите параметр `type` в значение `external`.
5. В секции `[dynamic]` установите параметр `dynamic_proxy` в значение `off`.
6. Если в локальной сети через один межсетевой экран (или NAT-устройство) работает несколько узлов с программным обеспечением ViPNet, задайте для таких узлов разные номера портов. Для этого в собственной секции `[id]` узла измените значение параметра `port`, определяющего номер порта, через который преобразованные в UDP-формат пакеты уходят с узла ViPNet Coordinator HW (порт источника) и приходят на данный узел (порт назначения).
7. Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

   ```
   hostname> iplir start
   ```

Информация об IP-адресе межсетевого экрана и порте доступа сообщается программой всем остальным узлам, с которыми связан ViPNet Coordinator HW.

Настройка режима «С динамической трансляцией адресов»

Если в локальной сети подключение происходит через некоторое устройство, выполняющее трансляцию адресов (NAT), на котором затруднительно настроить статические правила трансляции, и есть необходимость во взаимодействии с другими узлами, находящимися во внешней относительно этого устройства сети, то на узле нужно настроить режим работы с динамической трансляцией адресов.

Режим подключения с использованием такого типа межсетевого экрана наиболее универсален и может использоваться практически во всех случаях. Однако основное его назначение — обеспечить надежное двустороннее соединение с узлами, подключенными к сети через NAT-устройства, на которых настройка статических правил трансляции затруднена или невозможна. Такая ситуация типична, например, для простых NAT-устройств с минимумом настроек: DSL-модемов, беспроводных устройств и других. Затруднительно также произвести настройки на NAT-устройствах, установленных у провайдера (в домашних сетях, GPRS и других сетях, где провайдер предоставляет частный IP-адрес).

Для обеспечения возможности двусторонней работы на узле, подключенном к внешней сети через NAT-устройство, и всех его клиентов (если узел является координатором) на узле устанавливается режим работы через межсетевой экран с динамической трансляцией адресов. Одновременно во внешней сети должен присутствовать постоянно доступный координатор, который будет являться сервером соединений.

Рисунок 2. Подключение координатора через межсетевой экран с динамической трансляцией адресов

Организация соединений между узлами осуществляется в этом случае следующим образом:

- Узел в режиме работы через межсетевой экран с динамическим NAT после подключения к сети периодически отправляет IP-пакеты на свой сервер соединений. Период отправки таких IP-пакетов по умолчанию составляет 25 секунд. Этого, как правило, достаточно для работы через большинство NAT-устройств. При необходимости вы можете изменить этот период (тайм-аут).
- После того как связь между узлом и его сервером соединений будет установлена, узел начинает устанавливать соединение с другим узлом. Он начинает передавать тестовые IP-пакеты удаленному узлу через свой сервер соединений. Одновременно с этим узел передает тестовые IP-пакеты на сервер соединений удаленного узла и напрямую на удаленный узел.
- Если тестовые IP-пакеты дошли до удаленного узла, то удаленный узел регистрирует соединение и начинает передавать ответный IP-трафик напрямую. Узел при получении ответного IP-трафика от удаленного узла свой последующий IP-трафик ему также начинает передавать напрямую.

  Если тестовые IP-пакеты дошли только до сервера соединений удаленного узла, то сервер соединений регистрирует это соединение и отправляет напрямую узлу ответные IP-пакеты удаленного узла.

  То есть с удаленным узлом устанавливается прямое соединение или соединение через его сервер соединений. Если ответный IP-трафик так и не поступил от удаленного узла или его сервера соединений, то узел по-прежнему осуществляет соединение с удаленным узлом через свой сервер соединений.
- Возможность прямого соединения с удаленным узлом, находящимся за устройством с динамической трансляцией адресов, сохраняется по умолчанию в течение 75 секунд (трех интервалов отправки IP-пакетов) с момента окончания предыдущего соединения.

Для настройки работы ViPNet Coordinator HW в данном режиме выполните следующие действия:

1 Завершите работу управляющего демона с помощью команды:

```
hostname> iplir stop
```

2 Откройте файл `iplir.conf` для редактирования с помощью команды:

```
hostname# iplir config
```

3 В собственной секции `[id]`:

- параметр `usefirewall` установите в значение `on`;
- в параметре `port` укажите значение из диапазона 1–65535 (по умолчанию — 55777).

4 В секции `[dynamic]`:

- параметр `dynamic_proxy` установите в значение `on`;
- в параметре `forward_id` укажите идентификатор внешнего координатора (сервера соединений), с помощью которого ViPNet Coordinator HW будет устанавливать соединение с другими узлами.

Внимание! Выбранный сервер соединений должен быть доступен напрямую или через межсетевой экран со статической трансляцией адресов.

При подключении через межсетевой экран с динамическим NAT параметры `firewallip` и `port` секции `[dynamic]` определяются автоматически по информации, полученной от внешних узлов, редактировать их вручную не следует.

5 При необходимости в секции `[dynamic]`:

- измените значение параметра `timeout` (период отправки IP-пакетов серверу соединений);
- параметр `always_use_server` установите в значение `on`, если требуется направлять весь IP-трафик через сервер соединений. При этом учтите, что передача всего IP-трафика через сервер соединений может привести к снижению скорости обмена данными между узлами.

6 Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

```
hostname> iplir start
```

Примечание. На сервере соединений узла, работающего в режиме «С динамической трансляцией адресов», в секции `[id]` этого узла параметру `proxyid` будет присвоено значение `0xFFFFFFFE`.

Для всех остальных координаторов в секции `[id]` этого узла в параметр `proxyid` будет указан идентификатор узла, выполняющего функции его сервера соединений.

Принципы назначения виртуальных адресов

Виртуальные адреса назначаются сетевым узлам, которые подключаются к внешней сети через межсетевой экран. Необходимость использования виртуальных адресов обусловлена тем, что узлы, стоящие за разными межсетевыми экранами, могут иметь одинаковые адреса в своих частных сетях, и в случае обращения к ним по реальным адресам могла бы возникнуть неоднозначность. Для узлов, не находящихся за внешним межсетевым экраном, виртуальные адреса также назначаются, но не используются.

Параметры, необходимые для назначения виртуальных адресов, задаются в секции `[virtualip]` файла `iplir.conf`.

При появлении каждого сетевого узла в списке связей ему назначается виртуальный адрес, который привязан к уникальному идентификатору этого узла и соответствует его первому реальному адресу. Такой виртуальный адрес называется базовым, и он является точкой отсчета при назначении виртуальных адресов для каждого из реальных адресов узла.

Остальные виртуальные адреса сетевого узла, характеризующие каждый из реальных адресов узла, называются вторичными виртуальными адресами или для простоты — виртуальными адресами, и указываются в параметре `ip` соответствующей секции `[id]` через запятую после реального адреса.

Четыре октета, составляющие IP-адрес, используются при назначении виртуальных адресов следующим образом:

- Первый октет всех виртуальных адресов имеет одинаковое значение, соответствующее первому октету стартового виртуального адреса `startvirtualip` секции `[virtualip]`.
- Второй октет характеризует один из реальных адресов сетевого узла.
- Третий и четвертый октеты характеризуют сетевой узел. Они одинаковы для всех виртуальных адресов данного узла и различны для виртуальных адресов, принадлежащих разным узлам.

То есть при назначении виртуального адреса для следующего сетевого узла сначала увеличивается четвертый октет, а когда четвертый октет достигает максимального значения, увеличивается третий октет. Например:

Сетевой узел №1: 192.168.0.1 11.0.12.1

...

Сетевой узел №254: 192.168.0.254 11.0.12.254

Сетевой узел №255: 192.168.1.1 11.0.13.1

Если какой-либо сетевой узел имеет несколько реальных адресов, то при назначении виртуального адреса для следующего реального адреса того же сетевого узла увеличивается второй октет:

Ethernet 0: 192.168.0.1 11.44.12.1

Ethernet 1: 88.88.88.88 11.45.12.1

Ethernet 2: 200.0.0.1 11.46.12.1

Текущий адрес доступа к сетевому узлу определяется автоматически и содержится в параметре `accessip` соответствующей секции `[id]`. Если в данный момент узел виден по виртуальному адресу, то его адресом доступа считается базовый виртуальный адрес.

Вторичные виртуальные адреса могут использоваться, если нужно обратиться к конкретному реальному адресу данного сетевого узла, который виден по виртуальным адресам, а не к узлу вообще. Такая необходимость может возникнуть, например, если на данном сетевом узле работает приложение, которое ожидает сетевые запросы только по одному из адресов сетевого узла.

На схеме показан защищенный сервер с двумя сетевыми интерфейсами Ethernet 0 и Ethernet 1. На защищенном сервере настроен веб-сервис, который обрабатывает подключения по внешнему адресу 88.88.88.88 на Ethernet 1.

Рисунок 3. Схема использования вторичного виртуального адреса

Если настроена виртуальная видимость защищенного сервера, то обращение с сетевого узла ViPNet Client к защищенному серверу будет происходить по виртуальным адресам. При этом 11.0.0.1 — базовый виртуальный адрес защищенного сервера, соответствующий реальному адресу 10.0.0.2, а 11.1.0.1 — второй виртуальный адрес защищенного сервера, соответствующий реальному адресу 88.88.88.88, к которому ViPNet Client подключается для взаимодействия с веб-сервисом.

Пакет, отправленный с ViPNet Client на 11.1.0.1, после обработки драйвером iplir на ViPNet Client пойдет на реальный адрес 10.0.0.2 сетевого интерфейса Ethernet 0, затем, после обработки драйвером `iplir`, он будет направлен на адрес 88.88.88.88 и таким образом будет передан веб-сервису.

При обновлениях адресных справочников, а также изменениях в списке реальных адресов для какого-либо узла сетевые узлы сохраняют свои виртуальные адреса, а вновь добавленные узлы и реальные IP-адреса получают новые свободные виртуальные адреса (с учетом ограничения на максимально возможный адрес, заданного параметром `maxvirtualip` секции `[virtualip]`).

Настройка параметров видимости узлов

Для связи ViPNet Coordinator HW с защищенными узлами могут использоваться реальные или виртуальные IP-адреса. По умолчанию для связи с узлами, от которых ViPNet Coordinator HW получает широковещательные пакеты, используются их реальные адреса. Вы можете изменить настройки адресов видимости с помощью параметра `visibility` в секциях `[id]` и параметров в секции `[visibility]` файла `iplir.conf`.

Примечание. О настройках видимости туннелируемых узлов см. в разделе Настройка туннелируемых адресов (на стр. 81).

Видимость каждого защищенного узла определяется следующим образом:

1. Если в секции `[id]` узла, присутствует параметр `visibility`, то видимость узла определяется значением этого параметра.
2. Если в секции `[id]` узла нет параметра `visibility`, но при этом сеть, которой принадлежит узел, указана в параметре `subnet_real` или `subnet_virtual` секции `[visibility]`, то видимость узла определяется значением этого параметра.
3. В иных случаях видимость узла определяется значением параметра `default` секции `[visibility]`.

То есть индивидуальные настройки видимости узлов приоритетнее настроек видимости сетей, а настройки видимости сетей приоритетнее настроек видимости по умолчанию.

Чтобы настроить параметры видимости узлов, выполните следующие действия:

1. Завершите работу управляющего демона с помощью команды:

   ```
   hostname> iplir stop
   ```

2. Откройте файл `iplir.conf` для редактирования с помощью команды:

   ```
   hostname# iplir config
   ```

3. Чтобы настроить параметры видимости узлов по умолчанию и параметры видимости сетей, в секции `[visibility]`:
   - параметр `default` установите в одно из значений:
     - `auto` (по умолчанию) — видимость узлов определяется автоматически;
     - `real` — для доступа к узлам используются их реальные IP-адреса;
     - `virtual` — для доступа к узлам используются их виртуальные IP-адреса.
   - при необходимости укажите идентификаторы вашей сети ViPNet и доверенных сетей в одном из параметров:
     - `subnet_real` — идентификаторы сетей ViPNet, для которых необходимо настроить видимость узлов по реальным IP-адресам;
     - `subnet_virtual` — идентификаторы сетей ViPNet, для которых необходимо настроить видимость узлов по виртуальным IP-адресам.

     В каждом из этих параметров можно указать один либо несколько идентификаторов через запятую.
4. Чтобы настроить параметры видимости отдельных узлов, в соответствующих секциях `[id]` параметр `visibility` установите в одно из значений:
   - `auto` — определять видимости узла автоматически, в зависимости от его текущего адреса видимости.
   - `real` — всегда обращаться к узлу по его реальному адресу.

- `virtual` — всегда обращаться к узлу по его виртуальному адресу.

Внимание! Настраивать параметры видимости отдельных узлов нужно с осторожностью, так как если для двух узлов с одинаковыми реальными адресами, но разными виртуальными, установить параметр `visibility` в значение `real`, возникнет конфликт IP-адресов.

5 Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

```
hostname> iplir start
```

Настройка параметров виртуальных адресов

Каждый сетевой узел ViPNet автоматически формирует один или несколько виртуальных IP-адресов для каждого сетевого узла ViPNet и туннелируемого узла, с которым он связан. Каждому реальному адресу узла ставится в соответствие виртуальный IP-адрес. То есть число формируемых виртуальных адресов зависит от числа реальных адресов узла и числа адресов, туннелируемых этим узлом.

Чтобы настроить параметры виртуальных адресов ViPNet Coordinator HW, выполните следующие действия:

1 Завершите работу управляющего демона с помощью команды:

```
hostname> iplir stop
```

2 Откройте файл `iplir.conf` для редактирования с помощью команды:

```
hostname# iplir config
```

3 В секции `[virtualip]` при необходимости измените значения следующих параметров:

- `startvirtualip` — стартовый адрес для формирования базовых виртуальных адресов защищенных узлов (по умолчанию — `11.0.0.1`). При изменении данного параметра учитывайте, что назначение всех базовых виртуальных адресов производится заново, как при начальном формировании файлов конфигурации.
- `maxvirtualip` — максимальный адрес для формирования базовых виртуальных адресов (по умолчанию — `11.0.254.254`). Данный параметр используется для ограничения диапазона назначаемых базовых виртуальных адресов. Вы можете его уменьшить, при этом необходимо следить за тем, чтобы оно было больше значения параметра `endvirtualip`.
- `starttunnelvirtualip` — стартовый адрес для формирования виртуальных адресов туннелируемых узлов в автоматическом режиме (по умолчанию для диапазонов адресов туннелируемых узлов — `12.0.0.1`, для адресов одиночных туннелируемых узлов — `11.0.0.1`).

Примечание. Подробнее о режимах задания параметров туннелирования узлов см. в разделе Настройка туннелируемых адресов (на стр. 81).

4 Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

```
hostname> iplir start
```

Настройка туннелируемых адресов

Технология туннелирования позволяет защищать трафик открытых узлов корпоративной сети на потенциально опасном участке сети или включать открытые узлы в защищенную сеть без установки на эти узлы программного обеспечения ViPNet. Параметры туннелирования узлов задаются на координаторах сети ViPNet, которые будут выполнять туннелирование этих узлов.

Обычно параметры туннелирования рассылаются на узлы в составе справочников и ключей. Если туннелируемые адреса координатора заданы в программе ViPNet Центр управления сетью, то другие узлы получат информацию об этом автоматически. Если туннелируемые адреса заданы на координаторе вручную, эти адреса также необходимо указать вручную на каждом узле, который будет работать с этими туннелируемыми узлами посредством сети ViPNet.

В случае, если настройки были сначала заданы вручную на узлах, а затем пришли настройки из ЦУС, то настройки сравниваются и при этом:

- если настройки туннелирования для ViPNet Coordinator HW не менялись в ЦУС, то заданные локально настройки остаются без изменений;
- если с настройками туннелирования из ЦУС получены новые адреса туннелируемых узлов, то они добавляются в iplir.conf;
- если заданные на узлах адреса туннелируемых узлов были удалены из настроек туннелирования, полученных из ЦУС, то они будут удалены.

Задание виртуальных адресов для туннелируемых узлов может производиться в одном из следующих режимов:

- в автоматическом режиме (по умолчанию) (см. «Задание виртуальных адресов для туннелируемых узлов в автоматическом режиме» на стр. 82) — при этом задаются реальные адреса туннелируемых узлов, а виртуальные адреса для туннелируемых узлов назначаются автоматически;
- в ручном режиме (см. «Задание виртуальных адресов для туннелируемых узлов вручную» на стр. 83) — при этом реальные адреса туннелируемых узлов и виртуальные адреса для туннелируемых узлов задаются на узлах вручную.

Настройка видимости туннелируемых узлов (на стр. 84) по реальным или виртуальным адресам доступна при любом режиме задания виртуальных адресов для туннелируемых узлов.

Внимание! В случае обновления ViPNet Coordinator HW до версии 4.2 на узле, на котором вручную были заданы виртуальные адреса для туннелируемых узлов, эти настройки сохраняются. В случае перехода в автоматический режим все виртуальные адреса для туннелируемых узлов будут переназначены.

Задание виртуальных адресов для туннелируемых узлов в автоматическом режиме

Примечание. Мы рекомендуем настраивать туннелирование централизованно через ViPNet Центр управления сетью, в этом случае параметры туннелирования будут рассылаться на сетевые узлы в составе справочников и ключей.

Настраивать туннелирование на координаторе рекомендуется только когда централизованная настройка туннелирования в ViPNet Центр управления сетью не используется, в противном случае настройки туннелирования на координаторе будут заменены настройками из ЦУСа.

Вы можете задать реальные адреса туннелируемых узлов на координаторе вручную, а соответствующие виртуальные адреса для туннелируемых узлов будут заданы в автоматическом режиме. Для этого выполните следующие действия:

1 Завершите работу управляющего демона с помощью команды:

```
hostname> iplir stop
```

2 Откройте файл `iplir.conf` для редактирования (подробнее см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам»).

3 Убедитесь, что в секции `[misc]` параметр `tunnel_virt_assignment` установлен в значение `auto`.

4 В секции `[id]` измените значение параметра `tunnel` следующим образом:

`tunnel = <ip1>-<ip2>`, где:

`ip1` и `ip2` — начальный и конечный реальные адреса диапазона туннелируемых узлов. Параметры `ip3` и `ip4` указывать не нужно, они будут заданы автоматически.

Внимание! Если собственный узел туннелирует какие-либо компьютеры, то на всех туннелируемых компьютерах необходимо указать собственный узел в качестве шлюза по умолчанию.

Например, чтобы координатор туннелировал адреса с 192.168.201.5 по 192.168.201.10, а виртуальные адреса для этого диапазона реальных адресов были назначены автоматически, укажите:

```
tunnel = 192.168.201.5-192.168.201.10
```

5 Адреса `<ip1>-<ip2>`, указанные в параметре `tunnel`, также укажите на каждом узле, который будет работать с этими туннелируемыми узлами посредством сети ViPNet.

6 При необходимости в секции `[virtualip]` задайте начальный и конечный адреса для формирования виртуальных адресов (по умолчанию для одиночных туннелируемых адресов: `startvirtualip = 11.0.0.1`, `maxvirtualip = 11.0.255.254`, для диапазонов туннелируемых

узлов параметр определяется автоматически: `starttunnelvirtualip = <x+1>.0.0.1`, где `x` — первый октет `maxvirtualip`).

7 Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

8 `hostname> iplir start`

При переходе в автоматический режим назначения виртуальных адресов все виртуальные адреса для туннелируемых узлов будут переназначены. Чтобы обращение к узлам, туннелируемым ViPNet Coordinator HW, происходило по виртуальным адресам, настройте на каждом узле, который будет работать с этими туннелируемыми узлами, видимость по виртуальным адресам (см. «Настройка видимости туннелируемых узлов» на стр. 84).

Задание виртуальных адресов для туннелируемых узлов вручную

Примечание. Мы рекомендуем настраивать туннелирование централизованно через ViPNet Центр управления сетью, в этом случае параметры туннелирования будут рассылаться на сетевые узлы в составе справочников и ключей.

Настраивать туннелирование на координаторе рекомендуется только когда централизованная настройка туннелирования в ViPNet Центр управления сетью не используется, в противном случае настройки туннелирования на координаторе будут заменены настройками из ЦУСа.

Вы можете вручную задать реальные и виртуальные адреса туннелируемых узлов на координаторе. Для этого выполните следующие действия:

1 Завершите работу управляющего демона с помощью команды:

`hostname> iplir stop`

2 Откройте файл `iplir.conf` для редактирования (подробнее см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам»).

3 Убедитесь, что в секции `[misc]` параметр `tunnel_virt_assignment` установлен в значение `manual`.

Внимание! Настройка реальных и виртуальных адресов для туннелируемых узлов требует особой внимательности, поскольку в ручном режиме диапазоны виртуальных адресов не проверяются на наличие конфликтов.

4 В секции `[id]` измените значение параметра `tunnel` следующим образом:

`tunnel = <ip1>-<ip2> to <ip3>-<ip4>`, где `ip1` и `ip2` — начальный и конечный реальные адреса диапазона туннелируемых узлов, `<ip3>-<ip4>` — диапазон виртуальных адресов, которые соответствуют реальным адресам из диапазона `<ip1>-<ip2>`, и которые будут использоваться вместо реальных адресов туннелируемых узлов, если на узле, который к ним

обращается, настроена видимость по виртуальным адресам (см. «Настройка видимости туннелируемых узлов» на стр. 84). Например в случае, когда диапазон `<ip1>-<ip2>` принадлежит к частной сети и такие же адреса уже есть в локальной сети данного координатора. Параметр `ip4` можно не указывать, он формируется путем прибавления к `ip3` разницы между `ip2` и `ip1`.

Внимание! Если собственный узел туннелирует какие-либо компьютеры, то на всех туннелируемых компьютерах необходимо указать собственный узел в качестве шлюза по умолчанию.

Например, чтобы координатор туннелировал адреса с 192.168.201.5 по 192.168.201.10, а виртуальными адресами для этого диапазона реальных были адреса с 192.168.111.5 по 192.168.111.10, укажите:

```
tunnel = 192.168.201.5-192.168.201.10 to 192.168.111.5
```

5 Адреса `<ip1>-<ip2>`, указанные в параметре `tunnel`, будут туннелироваться другими координаторами. Также укажите `<ip3>-<ip4>`, туннелируемые ViPNet Coordinator HW, на каждом узле, который будет работать с этими туннелируемыми узлами посредством сети ViPNet.

6 Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

```
hostname> iplir start
```

Примеры настройки туннелей в типовых схемах работы ViPNet Coordinator HW см. в документе «ViPNet Coordinator HW. Сценарии работы».

Настройка видимости туннелируемых узлов

Вы можете настроить видимость туннелируемых узлов по реальным или виртуальным адресам для ViPNet Coordinator HW. Для этого выполните следующие действия:

1 Завершите работу управляющего демона с помощью команды:

```
hostname> iplir stop
```

2 Откройте файл `iplir.conf` для редактирования (подробнее см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам»).

3 В секции `[id]`, которая относится к туннелирующему узлу, в параметре `tunnelvisibility` укажите тип видимости туннелируемых узлов: `real` — реальные адреса или `virtual` — виртуальные адреса.

Внимание! Если вы не укажете значение параметра `tunnelvisibility` в секции `[id]`, которая относится к туннелирующему узлу, то видимость туннелируемых узлов будет определяться параметром `tunneldefault` секции `[visibility]`.

4 Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

```
hostname> iplir start
```

Настройка IP-адресов доступа к узлу и их приоритета

Каждый узел ViPNet взаимодействует с другим узлом, с которым у него есть связь, по определенному адресу доступа. Однако нередко некоторые узлы подключены к нескольким независимым каналам связи и имеют несколько адресов доступа, каждый из которых может использоваться для связи с другими узлами.

При наличии разных каналов для связи между узлами может возникнуть ситуация, когда для взаимодействия будет выбран менее предпочтительный канал (более дорогостоящий или менее быстродействующий). Например, если координатор, находящийся в центральном офисе организации, для выхода в Интернет использует одновременно два канала, предоставляемых разными провайдерами, то в результате автоматического определения адреса доступа может быть выбран менее предпочтительный канал, а более предпочтительный канал будет простаивать. Во избежание такой ситуации, вы можете задать приоритеты каналов связи, исходя из стоимости их обслуживания, быстродействия или других параметров.

Приоритеты каналов связи задаются путем настройки метрик (см. глоссарий, стр. 270) на стороне того сетевого узла, которому необходимо получить доступ к защищенному сегменту сети. Например, в случае взаимодействия узлов филиала и центрального офиса, если в центральном офисе используется одновременно два канала, то настройки производятся на стороне сетевого узла филиала.

Исходя из значений метрик каналов, ПО ViPNet Coordinator HW будет выбирать более приоритетный канал связи. Для этого при включении координатор ViPNet Coordinator HW отправляет служебные сообщения на все известные адреса доступа узлов с определенной задержкой, равной метрике, заданной для этих адресов в миллисекундах. Выбор канала для связи производится с учетом следующего:

- Адрес с наименьшей метрикой считается наиболее приоритетным. Соединение с узлом устанавливается по адресу с наименьшей метрикой всегда, когда этот адрес доступен.
- Если адрес с наименьшей метрикой недоступен или если все метрики равны, то соединение с узлом устанавливается по тому адресу, доступность которого определяется быстрее в результате опроса.
- Если помимо адресов с заданными метриками, есть адреса, для которых метрика установлена в значение `auto` (определяется автоматически), значение `auto` всегда будет на 100 миллисекунд больше максимального значения заданной метрики. То есть канал с автоматически определяемой метрикой всегда будет иметь самый низкий приоритет по сравнению с каналами, для которых метрика задана вручную.
- После установления соединения с узлом определение доступности других его адресов выполняется только при разрыве текущего соединения.

Настройка адресов доступа для узлов, связанных с координатором ViPNet Coordinator HW, осуществляется в соответствующих секциях `[id]` файла `iplir.conf`. Каждый адрес доступа узла

указывается в параметре `accessiplist` этой секции, количество данных параметров в секции не ограничено.

Примечание. IP-адреса доступа узла могут быть настроены централизованно в программе ViPNet Центр управления сетью. В этом случае они будут указаны в соответствующих секциях `[id]` файла `iplir.conf`. Для таких адресов вы также можете настроить приоритет, изменив соответствующие значения метрик (по умолчанию они определяются автоматически).

Чтобы указать IP-адреса доступа к какому-либо узлу, связанному с ViPNet Coordinator HW, и определить приоритеты этих адресов, выполните следующие действия:

1. Завершите работу управляющего демона с помощью команды:

   ```
   hostname> iplir stop
   ```

2. Откройте файл `iplir.conf` для редактирования с помощью команды:

   ```
   hostname# iplir config
   ```

3. В параметрах `accessiplist` секции `[id]` соответствующего узла выполните одно из действий:

   - Укажите IP-адреса доступа к узлу и задайте метрики для каждого из этих адресов в следующем виде:

     ```
     accessiplist = <IP-адрес узла>, <метрика>
     ```

     При задании значения метрики необходимо учитывать следующее:

     - рекомендуется указывать значения метрик больше максимального времени задержки при прохождении служебного сообщения и ответа на него по соответствующему каналу, но не превышающее 9999 миллисекунд;
     - рекомендуемая минимальная разница между метриками — 100 миллисекунд.

   - Если вы уже указали IP-адреса доступа ранее или они были получены из программы ViPNet Центр управления сетью автоматически, в параметре `accessiplist` задайте метрики для этих адресов в соответствии с рекомендациями, описанными выше.

4. Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

   ```
   hostname> iplir start
   ```

В результате в параметры `accessiplist` файла `iplir.conf` автоматически будут добавлены реальный IP-адрес сетевого интерфейса узла, через который будут передаваться IP-пакеты для выбранного IP-адреса доступа, условный номер этого интерфейса и тип регистрации данного IP-адреса доступа узла в виде:

```
accessiplist = <IP-адрес узла>, <метрика>, <IP-адрес интерфейса>, <номер интерфейса>, <тип регистрации>
```

Пример настройки приоритетов адресов доступа см. в документе «ViPNet Coordinator HW. Сценарии работы».

Настройка TCP-туннеля

При удаленном подключении клиентов к сетям ViPNet может возникать проблема с передачей IP-пакетов по протоколу UDP из-за того, что данный протокол блокируется некоторыми интернет-провайдерами. Но в этих случаях обычно бывает разрешен доступ только по протоколу TCP через порты 80 (HTTP) и 443 (HTTPS).

Для решения проблемы доступа можно организовать взаимодействие клиентов с другими узлами сети ViPNet через TCP-туннель (см. глоссарий, стр. 265), настроенный на сервере соединений этих клиентов. В этом случае если удаленный клиент не может связаться со своим сервером соединений по протоколу UDP, он автоматически начинает устанавливать с ним соединение через TCP-туннель. Сервер соединений передает полученные от клиента IP-пакеты дальше на узлы назначения по протоколу UDP. Таким образом, обмен трафиком между узлами ViPNet и удаленными узлами, недоступными по протоколу UDP, будет всегда осуществляться через сервер соединений.

Примечание. В случае если протокол UDP перестал блокироваться, соединение по данному протоколу восстановится при следующих условиях:

- по истечении тайм-аута отправки служебных IP-пакетов с клиента на сервер соединений (по умолчанию тайм-аут равен 5 минут);
- после проверки соединения этого узла с другим;
- после смены IP-адреса узла (например, при изменении точки доступа к Интернету).

Не используйте функцию TCP-туннеля без необходимости, так как ее использование делает ViPNet Coordinator HW уязвимым для атаки типа «отказ в обслуживании».

Для организации TCP-туннеля должны выполняться следующие условия:

- TCP-туннель можно настроить только на координаторе, подключенном к сети в режиме «Без использования межсетевого экрана» или «Через межсетевой экран со статической трансляцией адресов». В случае использования на координаторе стороннего межсетевого экрана необходимо настроить правило, разрешающее соединения через TCP-туннель.
- На ViPNet Coordinator HW в файле `iplir.conf` в списке адресов доступа (параметр `accessiplist` секции `[id]`) должен быть задан хотя бы один публичный адрес, поскольку для установления TCP-соединения клиент использует именно публичный IP-адрес доступа сервера соединений.
- На ViPNet Coordinator HW должен быть включен TCP-туннель и настроен порт для входящих TCP-соединений (см. описание ниже).
- На клиентах должен быть задан номер порта, который используется на ViPNet Coordinator HW для TCP-соединений.
- Как правило, номер порта должен поступать на клиент от сервера соединений в составе служебной информации. Если по каким-либо причинам на клиент не поступила справочная информация с номером порта (например, клиент недоступен), то порт можно задать на нем вручную.

По умолчанию на ViPNet Coordinator HW TCP-туннель выключен. При включении TCP-туннеля автоматически начинает использоваться порт 80 для входящих TCP-соединений.

Чтобы просмотреть текущие настройки TCP-туннеля, выполните команду:

```
hostname# iplir show tcptunnel-info
```

В результате выполнения команды будет указано, включен или выключен TCP-туннель, номер порта, возможное количество TCP-соединений и количество текущих TCP-соединений.

Примечание. ViPNet Coordinator HW текущей версии в любом исполнении может поддерживать не более 100 соединений через TCP-туннель.

Чтобы включить TCP-туннель, в файле `iplir.conf` в секции `[misc]` параметру `tcptunnel_establish` присвойте значение `on`.

Если порт для TCP-соединений, заданный в настройках ViPNet Coordinator HW, используется какой-либо другой службой, во избежание конфликта укажите другой порт для TCP-соединений. Для этого выполните следующие действия:

1. Завершите работу управляющего демона с помощью команды:

   ```
   hostname> iplir stop
   ```

2. Откройте файл `iplir.conf` для редактирования с помощью команды:

   ```
   hostname# iplir config
   ```

3. В собственной секции `[id]` в параметре `tcptunnelport` укажите номер порта, на который будут приходить TCP-пакеты от защищенных узлов.

4. Сохраните изменения в файле `iplir.conf` и запустите управляющий демон с помощью команды:

   ```
   hostname> iplir start
   ```

После изменения номера порта на все сетевые узлы, для которых данный координатор является сервером соединений, будет отправлена служебная информация с новым номером порта для TCP-соединений.

Настройка защиты соединения по технологии L2OverIP

Общее описание технологии L2OverIP

В ViPNet Coordinator HW реализована поддержка технологии L2OverIP (см. глоссарий, стр. 264), которая позволяет организовать защиту удаленных сегментов сети, использующих одно и то же адресное пространство, на канальном уровне модели OSI. В результате узлы из разных сегментов смогут взаимодействовать друг с другом так, как будто они находятся в одном сегменте с прямой видимостью по MAC-адресам. Такая защита может потребоваться, например, для организации работы территориально распределенных ЦОДов (центров обработки данных), локальные сети которых объединены высокоскоростным каналом связи и представляют собой единый домен коллизий (см. глоссарий, стр. 268).

Внимание! Исполнения ViPNet Coordinator HW50 A, B и ViPNet Coordinator HW100 A, B не поддерживают функцию L2OverIP.

Функцию L2OverIP нельзя использовать для объединения сегментов, которые соединены какими-то другими каналами связи. Сегменты должны быть полностью разобщены. Иначе это может парализовать работу всей сети.

Для работы функции L2OverIP в исполнении ViPNet Coordinator HW VA необходимо в настройках среды виртуализации включить неразборчивый режим (Promiscuous Mode) для интерфейса, к которому привязан адаптер виртуальной машины. Подробнее см. руководство администратора платформы виртуализации, которую вы используете.

Технология L2OverIP предполагает взаимодействие между узлами нескольких удаленных сегментов сети через ViPNet Coordinator HW, которые установлены на границе этих сегментов. В основе технологии лежит перехват на канальном уровне модели OSI Ethernet-кадров, отправленных из одного сегмента сети в другой. Каждый ViPNet Coordinator HW осуществляет перехват Ethernet-кадров, отправленных из его сегмента сети в другой, их упаковку в IP-пакеты специального формата и передачу этих IP-пакетов другому ViPNet Coordinator HW по защищенному каналу. ViPNet Coordinator HW, получивший IP-пакеты специального формата, извлекает из них исходные кадры и передает получателям в своем сегменте.

Примечание. В большинстве практически важных случаев нужного результата можно достичь, объединяя не более 252 IP-адресов во всех объединяемых сегментах сети. Многократное превышение этого количества может привести к серьезному снижению производительности сети в объединяемых сегментах.

С увеличением числа IP-адресов в одном широковещательном сегменте сети значительно возрастает количество широковещательных служебных пакетов, что приводит к снижению пропускной способности сети и увеличению нагрузки на ViPNet Coordinator HW.

С помощью функции L2OverIP можно объединить несколько сегментов сети, в том числе сегменты, состоящие из виртуальных локальных сетей (VLAN). Возможны следующие варианты объединения:

- сегменты без VLAN, находящиеся за сетевыми интерфейсами класса `access`;
- виртуальная сеть VLAN из одного сегмента и виртуальная сеть VLAN из другого сегмента;
- все виртуальные сети VLAN из одного сегмента со всеми виртуальными сетями другого сегмента (для этого необходимо указать сетевые интерфейсы класса `trunk`, за которыми находятся виртуальные сети VLAN; кроме того, адреса виртуальных сетей VLAN должны совпадать);
- одна из виртуальных сетей VLAN сегмента и сегмент без VLAN, находящийся за сетевым интерфейсом класса `access`.

Примечание. Текущая версия ViPNet Coordinator HW позволяет объединить не более 31 сегмента сети.

Включение, выключение и настройка функции L2OverIP выполняется с помощью командного интерпретатора (см. «Организация защиты соединения между удаленными сегментами сети на канальном уровне модели OSI» на стр. 93) или веб-интерфейса.

Примечание. С помощью функции L2OverIP нельзя объединить между собой сети, находящиеся за ViPNet Coordinator HW разных версий (3.x и 4.x).

При использовании функции L2OverIP ViPNet Coordinator HW работает как виртуальный сетевой коммутатор, хранящий в памяти таблицу MAC-адресов сетевых узлов, от которых поступает сетевой трафик.

Каждому сегменту сети назначается свой номер порта, номера портов задаются в настройках L2OverIP. Порт, заданный для собственного сегмента сети, называется локальным. Порты, заданные на ViPNet Coordinator HW в других сегментах, называются удаленными. Трафик самого ViPNet Coordinator HW всегда относится к порту с номером 0.

Виртуальный коммутатор может по-разному обрабатывать одноадресные Ethernet-кадры с неизвестным MAC-адресом получателя: блокировать либо обрабатывать как многоадресные с рассылкой на все удаленные порты. Последняя возможность позволяет использовать функцию L2OverIP для схемы, когда два ViPNet Coordinator HW объединены с помощью агрегированного

канала и каждый из физических каналов работает через свою пару ViPNet Coordinator HW. В такой схеме (см. Рисунок 4 на стр. 92) кадры, которыми обмениваются два узла из разных сегментов, могут в одном направлении постоянно пересылаться через одну пару ViPNet Coordinator HW, а в другом направлении — через другую. В этом случае адрес получателя кадров будет определяться как неизвестный, и для обработки пакетов многоадресной рассылки следует использовать режим `smart-broadcast` (см. «Организация защиты соединения между удаленными сегментами сети на канальном уровне модели OSI» на стр. 93).

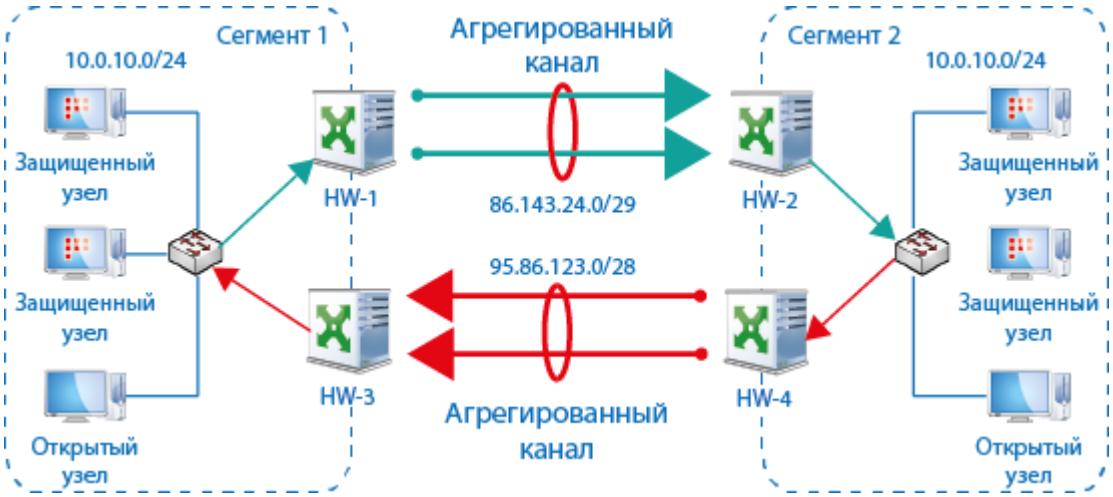

Рисунок 4. Схема работы сети с парами ViPNet Coordinator HW, объединенных с помощью агрегированных каналов

Рассылка Ethernet-кадров с неизвестным MAC-адресом получателя выполняется в соответствии с выбранным режимом обработки:

- кадры, принятые из локального порта, пересылаются на все удаленные порты и на порт с номером 0;
- кадры, принятые из удаленного порта, пересылаются на локальный порт и на порт с номером 0;
- кадры, принятые из порта с номером 0, в зависимости от выбранного режима либо блокируются, либо пересылаются на локальный порт и на все удаленные порты.

Функцию L2OverIP можно использовать в кластере горячего резервирования. В этом случае настройки функции L2OverIP достаточно выполнить на активном сервере, на пассивный сервер они передаются автоматически, если включено резервирование файлов конфигурации. При этом включить функциональность можно только на сервере, находящемся в активном режиме, на пассивном сервере он всегда выключен и будет включен только при переходе сервера в активный режим.

При работе в кластере горячего резервирования таблица MAC-адресов не передается пассивному серверу, и при переключении пассивного сервера в активный режим его таблица пуста. Это может привести к увеличению времени конвергенции сети после смены активного сервера — до тех пор, пока узлы из разных сегментов, взаимодействующие друг с другом, не отправят заново ARP-запросы.

Организация защиты соединения между удаленными сегментами сети на канальном уровне модели OSI

При организации соединения между удаленными сегментами сети на канальном уровне каждый сегмент сети подключается к одному из интерфейсов ViPNet Coordinator HW, установленному на границе сегмента. Если сегмент сети состоит из нескольких VLAN (см. глоссарий, стр. 266), они должны быть объединены с помощью коммутатора в транковый порт, к которому подключается один из интерфейсов ViPNet Coordinator HW (см. «Организация обработки трафика из нескольких VLAN» на стр. 53).

Чтобы организовать защиту соединения сегментов, на каждом ViPNet Coordinator HW включите функцию L2OverIP и задайте ее параметры. Для этого выполните следующие действия:

1 Перейдите в режим администратора с помощью команды `enable`. В ответ на приглашение введите пароль администратора.

2 Укажите сетевой интерфейс, сегмент сети которого будет объединяться с другим сегментом с помощью функции L2OverIP:

```
hostname# iplir set l2overip interface <рабочий интерфейс>
```

Примечание. Если требуется объединить отдельную VLAN одного сегмента сети с другим сегментом сети, на ViPNet Coordinator HW, стоящем на границе сегмента с VLAN, укажите в качестве рабочего интерфейса виртуальный интерфейс VLAN в формате `<физический интерфейс>.<номер виртуального интерфейса>`.

3 Задайте параметры локального сегмента сети, указав уникальный номер порта и IP-адрес внешнего интерфейса с помощью команды:

```
hostname# iplir set l2overip local-port <номер порта> <IP-адрес внешнего интерфейса>
```

В качестве номера порта вы можете задать значение из интервала 1–31. Каждому сегменту необходимо назначить свой номер порта.

4 Задайте параметры удаленного сегмента сети, указав его номер порта и актуальный адрес видимости удаленного ViPNet Coordinator HW с помощью команды:

```
hostname# iplir set l2overip remote-port <номер порта> <IP-адрес видимости удаленного ViPNet Coordinator HW>
```

В зависимости от настройки видимости удаленного ViPNet Coordinator HW укажите его реальный или виртуальный адрес.

Примечание. При указании виртуального адреса удаленного ViPNet Coordinator HW убедитесь, что этот виртуальный адрес является базовым (см. «Принципы назначения виртуальных адресов» на стр. 76). Базовые виртуальные адреса имеют 0 во втором октете (`x.0.x.x`).

5 В случае необходимости вы можете удалить из настроек параметры другого сегмента. Для этого выполните команду:

```
hostname# iplir set l2overip remote-port <номер порта> delete
```

Если IP-адрес видимости другого сегмента изменился, выполните команду:

```
hostname# iplir set l2overip remote-port <номер порта> <новый IP-адрес>
```

6 При необходимости измените время жизни MAC-адреса в таблице MAC-адресов виртуального коммутатора (см. «Общее описание технологии L2OverIP» на стр. 90) при отсутствии трафика, поступающего от этого адреса. Для этого выполните команду:

```
iplir set l2overip mac-ttl <время жизни MAC-адреса>
```

Время жизни MAC-адреса задается в секундах. Значение по умолчанию — 300 секунд. При необходимости вы можете задать значение из интервала 60–86400.

7 Добавьте сетевой фильтр защищенной сети, разрешающий любые соединения по протоколу 97, с помощью команды:

```
hostname# firewall vpn add src @any dst @any proto 97 pass
```

8 Включите функцию L2OverIP с помощью команды:

```
hostname# iplir set l2overip mode switch
```

9 Для просмотра текущего состояния и настроек функции L2OverIP выполните команду:

```
hostname# iplir show l2overip config
```

После настройки и включения функции L2OverIP на всех ViPNet Coordinator HW взаимодействие между узлами удаленных сегментов сети будет защищено на канальном уровне модели OSI.

Внимание! На двух ViPNet Coordinator HW, между которыми организовано соединение L2OverIP, IP-адреса локального и удаленного сегментов должны быть настроены симметричным образом. То есть на каждом ViPNet Coordinator HW в качестве IP-адреса удаленного сегмента должен быть указан IP-адрес видимости сетевого интерфейса, который задан на втором ViPNet Coordinator HW в качестве внешнего интерфейса локального сегмента.

Примеры организации защищенного соединения между удаленными сегментами сети с использованием технологии L2OverIP см. в документе «ViPNet Coordinator HW. Сценарии работы», раздел «Защита соединения между удаленными сегментами сети на канальном уровне модели OSI».

6

Обеспечение отказоустойчивости

Настройка системы защиты от сбоев

Назначение и принципы работы системы защиты от сбоев

Система защиты от сбоев предназначена для контроля работоспособности ПО ViPNet Coordinator HW и создания отказоустойчивого решения на базе узлов ViPNet Coordinator HW. Данная система может работать в одиночном режиме (см. «Работа системы защиты от сбоев в одиночном режиме» на стр. 96) или в режиме кластера горячего резервирования (см. «Работа системы защиты от сбоев в режиме кластера горячего резервирования» на стр. 97).

Настройка системы защиты от сбоев выполняется путем редактирования конфигурационного файла `failover.ini`. Подробнее о параметрах, содержащихся в этом файле см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам».

Работа системы защиты от сбоев в одиночном режиме

По умолчанию после установки программы ViPNet Coordinator HW система защиты от сбоев работает в одиночном режиме. При этом данная система обеспечивает постоянную работоспособность программы, выполняя следующие функции:

- контроль собственной работоспособности;
- контроль работоспособности демонов и драйверов ViPNet Coordinator HW, ведение статистики использования системных ресурсов;
- контроль сбоев при обработке пакетов драйвером ViPNet.

Данные функции реализуются watchdog-драйвером itcswd и демоном failoverd. При загрузке системы сначала запускается watchdog-драйвер, а затем демон failoverd, который запускает остальные демоны и драйверы ViPNet Coordinator HW. Watchdog-драйвер и демон failoverd работают постоянно в фоновом режиме.

Watchdog-драйвер отвечает за контроль собственной работоспособности системы защиты от сбоев, то есть работоспособности демона failoverd. В зависимости от настроек (подробнее см. в описании секции `[misc]` файла `failover.ini` в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам») при запуске демон failoverd может регистрироваться watchdog-драйвером. В этом случае в процессе своей работы он периодически посылает драйверу сообщения для подтверждения своей работоспособности. Если по истечении определенного времени от демона не будет получено ни одного сообщения, драйвер инициирует перезагрузку системы.

Демон failoverd отвечает за контроль работоспособности остальных демонов и драйверов ViPNet Coordinator HW. В зависимости от настроек (подробнее см. в описании секции `[misc]` файла `failover.ini` в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам») при запуске демоны ViPNet Coordinator HW могут регистрироваться демоном failoverd. В этом случае в процессе своей работы каждый из них периодически посылает демону failoverd сообщения для подтверждения своей работоспособности. Если по истечении определенного времени от какого-либо демона не будет получено ни одного сообщения или от драйвера ViPNet будет получено сообщение с кодом ошибки обработки пакетов, демон failoverd считает, что произошел сбой и пытается завершить работу демона. Если завершить работу корректно не удается, демон failoverd производит принудительный перезапуск. Если после пяти перезапусков подряд, работу демона восстановить не удалось, он считается неработоспособным, и демон инициирует перезапуск системы.

Примечание. Система защиты от сбоев контролирует работоспособность только запущенных демонов и драйверов ViPNet Coordinator HW.

Работа системы защиты от сбоев в режиме кластера горячего резервирования

Помимо контроля работоспособности программы ViPNet Coordinator HW (см. «Работа системы защиты от сбоев в одиночном режиме» на стр. 96), в режиме кластера горячего резервирования система защиты от сбоев позволяет передавать функции вышедшего из строя сервера другому (резервному) серверу. Кластер горячего резервирования состоит из двух взаимосвязанных серверов ViPNet Coordinator HW:

- активного сервера — который работает в активном режиме и выполняет функции координатора ViPNet (подробнее см. в документе «ViPNet Coordinator HW. Общее описание»);
- пассивного сервера — который работает в пассивном режиме, то есть в режиме ожидания.

В случае сбоев, критичных для работоспособности ViPNet Coordinator HW на активном сервере, пассивный сервер переключается в активный режим и выполняет функции сбойного сервера, который после перезагрузки переходит в пассивный режим.

При работе в режиме кластера горячего резервирования некоторые функции ViPNet Coordinator HW недоступны (см. «Функции ViPNet Coordinator HW, недоступные в режиме кластера горячего резервирования» на стр. 99).

Внимание! На обоих серверах кластера горячего резервирования должна быть установлена одна и та же версия ПО ViPNet Coordinator HW, а также справочники и ключи одного и того же узла. Оба сервера должны иметь одинаковую аппаратную платформу.

Если вы развертываете кластер на базе ViPNet Coordinator HW VA, оба сервера должны иметь одинаковые параметры эмулируемого аппаратного обеспечения: количество процессоров, ОЗУ и сетевых интерфейсов.

Каждый сетевой интерфейс кластера горячего резервирования имеет один IP-адрес, по которому кластер доступен для других узлов. Этот IP-адрес всегда принадлежит серверу, работающему в данный момент в активном режиме. Сетевые интерфейсы сервера, работающего в пассивном режиме, имеют другие IP-адреса, которые не используются другими узлами для установления соединения с кластером. Причем каждый сетевой интерфейс пассивного сервера имеет индивидуальный IP-адрес и, чтобы после перезагрузки каждый интерфейс сервера получал свой IP-адрес для работы в пассивном режиме, администратор сети ViPNet должен соответствующим образом настроить стек IP на серверах кластера.

Внимание! При работе с типовой схемой кластера горячего резервирования все используемые IP-адреса (IP-адрес активного сервера и два IP-адреса пассивного сервера) должны быть статическими и находиться в одном адресном пространстве. Если возможности по выделению IP-адресов в сети ограничены, может применяться схема, рассчитанная на выделение одного реального IP-адреса для активного сервера (подробнее см. в документе «ViPNet Coordinator HW. Сценарии работы»).

При запуске кластера активным становится тот сервер, который запускается быстрее. После загрузки пассивный сервер периодически посылает в сеть запросы на IP-адреса активного сервера. Если ни один из IP-адресов недоступен в течение заданного времени (то есть активный сервер не найден в сети), пассивный сервер переходит в активный режим. При этом ему назначается IP-адрес активного сервера.

Активный сервер в процессе работы периодически контролирует работоспособность своих интерфейсов, анализируя IP-трафик, проходящий через каждый из них. Если количество IP-пакетов, зарегистрированных по истечении определенного интервала времени, больше количества IP-пакетов, которое было зарегистрировано к моменту начала этого интервала времени, то считается, что интерфейс работает нормально, и счетчик отказов для этого интерфейса сбрасывается. Если в течение этого интервала времени не было ни отправлено, ни принято ни одного пакета, то производится дополнительная проверка — через каждый интерфейс посылаются эхо-запросы стабильным объектам сети (например, маршрутизатору). Если на какой-либо из интерфейсов в течение заданного времени не пришло ответа ни на один эхо-запрос, счетчик отказов для этого интерфейса увеличивается на единицу. При достижении счетчиком определенного значения интерфейс считается неработоспособным, и активный сервер перезагружается. Во время перезагрузки активный сервер становится недоступен, при этом пассивный сервер переходит в активный режим. После перезагрузки сервер, работавший в активном режиме, переходит в пассивный режим.

Примечание. Если стабильные объекты сети находятся в открытой сети, то для отправки эхо-запросов и получения ответов на серверах ViPNet Coordinator HW автоматически создаются соответствующие разрешающие фильтры открытой сети для протокола ICMP.

При работе сервера в активном режиме эхо-запросы для контроля работоспособности его интерфейсов могут посылаться постоянно (подробнее см. в описании файла `failover.ini` в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам»).

Для поддержания справочников и ключей, конфигурационных файлов и журналов на обоих серверах ViPNet Coordinator HW в актуальном состоянии между серверами должен быть создан и настроен выделенный Ethernet-канал (резервный канал), по которому копии файлов с активного сервера будут периодически передаваться на пассивный. Кроме того, по резервному каналу активный сервер передает пассивному копии принятых и готовых к отправке MFTP-конвертов, и серверы обмениваются пакетами синхронизации, содержащими информацию о текущем режиме работы каждого из них. Обмен пакетами синхронизации позволяет избежать ситуации, когда оба сервера кластера работают в активном режиме (например, в случае практически одновременного запуска). Резервный канал используется только для передачи файлов и пакетов с целью резервирования, данные передаются по резервному каналу в открытом виде. Система защиты от сбоев не контролирует работоспособность резервного канала.

Внимание! Адреса резервного канала на серверах кластера не должны совпадать с адресами защищенных узлов, иначе трафик, идущий на эти адреса, будет блокироваться. Если необходимо использовать одинаковые адреса, для разрешения конфликта рекомендуется настроить видимость соответствующих защищенных узлов по виртуальным адресам (см. «Настройка параметров видимости узлов» на стр. 77).

На пассивном сервере кластера блокируется любой защищенный трафик, а также любой открытый трафик, кроме обмена данными по резервному каналу и периодического опроса адресов активного сервера. Сетевые фильтры, разрешающие такой открытый трафик между соответствующими интерфейсами серверов, создаются в ViPNet Coordinator HW автоматически (см. «Сетевые фильтры по умолчанию» на стр. 234).

Функции ViPNet Coordinator HW, недоступные в режиме кластера горячего резервирования

В режиме кластера недоступны следующие сетевые службы ViPNet Coordinator HW:

- DHCP-сервер.
- Служба DHCP-relay.

Перед переключением в режим кластера горячего резервирования необходимо отключить перечисленные функции.

Развертывание кластера горячего резервирования

Перед развертыванием кластера горячего резервирования:

1 Убедитесь в наличии двух серверов, каждый из которых имеет минимум три сетевых интерфейса: для доступа во внешнюю сеть, для доступа во внутреннюю сеть и для

организации резервного канала (см. «Работа системы защиты от сбоев в режиме кластера горячего резервирования» на стр. 97).

2 На обоих серверах установите одинаковую версию программного обеспечения ViPNet Coordinator HW, одинаковые справочники и ключи (подробнее см. в документе «ViPNet Coordinator HW. Подготовка к работе»).

3 Настройте параметры сетевых интерфейсов, соблюдая следующие ограничения:

- Для сетевых интерфейсов, задействованных в работе кластера горячего резервирования, должны быть заданы статические IP-адреса. Назначение IP-адресов по протоколу DHCP не допускается.

Примечание. В качестве резервируемых сетевых интерфейсов вы можете задать как физические, так и виртуальные или VLAN интерфейсы.

- Во избежание конфликтов при копировании конфигурационных файлов с активного сервера на пассивный требуется, чтобы имена интерфейсов, используемых для доступа к соответствующим сетям, совпадали.
- В соответствующих секциях файла `failover.ini` должны быть указаны все сетевые интерфейсы класса `access`, описанные секциями `[adapter]` в файле `iplir.conf` (подробнее см. документ «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам»).

Примечание. В качестве резервируемых сетевых интерфейсов можно задать только интерфейсы класса `access`. Указывать в файле `failover.ini` сетевые интерфейсы классов `trunk` и `slave` запрещено.

4 Между серверами создайте резервный канал для передачи файлов с целью резервирования. Необходимо настроить систему так, чтобы при ее перезапуске связь по резервному каналу устанавливалась автоматически.

5 Выберите один или несколько стабильных объектов сети (например, маршрутизатор), которые будут использоваться для проверки работоспособности интерфейсов активного сервера.

Для развертывания кластера на каждом из серверов выполните следующие действия:

1 Завершите работу демона системы защиты от сбоев командой:

```
hostname> failover stop
```

2 Откройте файл `failover.ini` для редактирования с помощью команды:

```
hostname# failover config edit
```

3 В файле `failover.ini` для сетевых интерфейсов класса `access`, которые используются для взаимодействия с узлами внутренней и внешней сетей, в соответствующих секциях `[channel]` укажите значения следующих параметров:

- `device` — имя интерфейса.

- `activeip` — IP-адрес и маска, которые интерфейс будет иметь в активном режиме.
- `passiveip` — IP-адрес и маска, которые интерфейс будет иметь в пассивном режиме.
- `testip` — IP-адрес маршрутизатора или другого стабильного объекта сети, которому будут посылаться эхо-запросы для проверки работоспособности интерфейса.

  При необходимости можно для каждого из интерфейсов указать несколько параметров `testip`. В этом случае сбоем интерфейса будет считаться ситуация, когда ни от одного из заданных IP-адресов не будет получено ответа.

  Например, чтобы эхо-запросы отправлялись на IP-адреса 192.168.100.34 и 192.168.100.25, добавьте следующие строки:

  ```
  testip = 192.168.100.34
  testip = 192.168.100.25
  ```

Внимание! На серверах кластера значения параметров `device`, `activeip` и `testip` должны быть одинаковыми, а значения параметров `passiveip` — разными. IP-адрес, указанный в параметре `passiveip` каждого сервера, должен совпадать с IP-адресом, который установлен для соответствующего интерфейса в системе.

В качестве параметра `testip` не рекомендуется задавать адрес интерфейса «внутренней петли» (loopback), например `127.0.0.1` или `localhost`, так как в этом случае реальной проверки работоспособности сетевого интерфейса не производится.

- `ident` — текстовая строка, идентифицирующая интерфейс. Для интерфейсов, подключенных к одинаковым сетям, параметры `ident` должны совпадать.
- `checkonlyidle` — указание на необходимость проверки только неактивных интерфейсов. Возможные значения: `yes` (по умолчанию) — активный сервер посылает эхо-запросы на интерфейсы, адреса которых указаны в параметрах `testip`, только если за период опроса IP-адресов на данных интерфейсах не было ни входящих, ни исходящих пакетов; `no` — эхо-запросы на интерфейсы, адреса которых указаны в параметрах `testip`, посылаются всегда.

4 В файле `failover.ini` в секции `[sendconfig]` задайте следующие параметры, контролирующие пересылку файлов с активного сервера на пассивный с целью резервирования:

- `device` — системное имя интерфейса, который используется для организации резервного канала.
- `activeip` — адрес резервного канала другого сервера кластера (который работает в режиме, противоположном режиму данного сервера).

5 При необходимости измените другие параметры секции `[sendconfig]` и параметры работы системы защиты от сбоев, относящиеся к посылке пакетов в сеть, в секции `[network]` файла `failover.ini`.

6 Сохраните изменения в файле `failover.ini`.

7 Переведите систему защиты от сбоев в режим кластера горячего резервирования с помощью команды:

```
hostname# failover config mode cluster
```

Примечание. При переключении в режим кластера горячего резервирования проверяется текущее состояние локального DHCP-сервера. Если DHCP-сервер запущен или в настройках включен его автоматический запуск, то появляется предупреждение о необходимости предварительно завершить работу DHCP-сервера или выключить его автоматический запуск.

8 Запустите демон системы защиты от сбоев командой:

```
hostname> failover start
```

Внимание! Настройку сетевых служб необходимо выполнять после развертывания кластера и запуска демона системы защиты от сбоев. Настройка выполняется на активном сервере кластера.

В результате кластер начнет свою работу и на серверах ViPNet Coordinator HW будут автоматически созданы нередактируемые сетевые фильтры, необходимые для корректной работы системы защиты от сбоев с выбранными параметрами.

Примечание. Чтобы впоследствии изменить пароль пользователя на ViPNet Coordinator HW, работающих в режиме кластера горячего резервирования, выполните команду `hostname# admin passwd` сначала на сервере, функционирующем в пассивном режиме, а затем на сервере, функционирующем в активном режиме. Пароли, задаваемые на обоих серверах, должны совпадать.

Примеры настройки кластера горячего резервирования см. в документе «ViPNet Coordinator HW. Сценарии работы».

Запуск и завершение работы демона системы защиты от сбоев

Чтобы запустить демон системы защиты от сбоев failoverd на одиночном сервере, выполните команду:

```
hostname> failover start
```

Чтобы запустить демон failoverd на сервере кластера горячего резервирования, выполните одну из команд:

- Для запуска в активном режиме:

  ```
  hostname> failover start active
  ```

- Для запуска в пассивном режиме:

```
hostname> failover start passive
```

Внимание! Если при запуске демона failoverd на сервере кластера горячего резервирования вы не укажите параметр `active` или `passive`, то демон будет запущен в том режиме, в котором он находился до завершения работы.

Перед запуском демона системы защиты от сбоев на сервере кластера горячего резервирования в активном или пассивном режиме убедитесь в том, что второй сервер кластера работает в противоположном режиме. Запуск обоих серверов кластера в активном режиме вызовет конфликт IP-адресов и другие неполадки.

Чтобы завершить работу демона failoverd, выполните команду:

```
hostname> failover stop
```

Просмотр информации о работе системы защиты от сбоев

Текущее состояние системы защиты от сбоев

Для просмотра информации о текущем состоянии системы защиты от сбоев узла:

```
hostname> failover show info
```

В результате выводится следующая информация об узле:

- версии ViPNet Coordinator HW и демона failoverd, установленных на узле (`versions`);
- идентификатор и имя узла (`ID`);
- локальное время на узле (`workstation time`);
- режим работы системы защиты от сбоев узла (`failover mode`):
  - `single` — одиночный режим работы;
  - `active` — активный режим работы в составе кластера горячего резервирования;
  - `passive` — пассивный режим работы в составе кластера горячего резервирования;
- время непрерывной работы демона failoverd (`failover uptime`);
- общая загруженность процессора в процентах (`total cpu`);
- общий объем памяти в килобайтах (`total memory`);
- общий объем свободной памяти в килобайтах (`free memory`);
- сведения о текущем состоянии демонов ViPNet Coordinator HW:
  - `initializing` – загружается;
  - `works` — работает;
  - `stopped` — работа завершена;

- `unknown` — неизвестно (например, если произошел сбой в работе демона, была произведена попытка его перезапуска, и данных о его состоянии пока нет);

- сведения о ресурсах процессора, используемых каждым из этих демонов, в процентах (`failover cpu`, `iplir cpu` и так далее).

Примечание. В случае просмотра информации о текущем состоянии системы защиты от сбоев на узле, работающем в составе кластера горячего резервирования, статистика использования системных ресурсов и текущее состояние демонов ViPNet Coordinator HW отображаются для обоих серверов кластера (в столбце `local` — для сервера, на котором выполнена команда, в столбце `remote` — для второго сервера кластера, см. пример ниже)

Например, если демон failoverd работает в одиночном режиме, выводится следующая информация:

```
Versions: ViPNet 4.2.0 (30), daemon 1.5 (1)
Workstation configured for ID 29A0022 (Coordinator_HW)
The workstation works in a single mode of protection against failures
Workstation time (utc: 1204719868) Thu Mar 25 13:07:10 2015

failover mode     * single
failover uptime   * 6d  0:23
total cpu         * 100%
total memory      * 2055840 kB
free memory       * 1883540 kB
failover state    * works
failover cpu      * 0%
iplir state       * works
iplir cpu         * 46%
mftp state        * works
mftp cpu          * 40%
alg state         * works
alg cpu           * 19%
webgui state      * works
webgui cpu        * 0%
```

Если демон failoverd работает в режиме кластера горячего резервирования, выводится следующая информация:

```
Versions: ViPNet 4.2.0 (30), daemon 1.5 (1)
Workstation configured for ID 29A0022 (Coordinator_HW)
Workstation works in a cluster mode of protection against failures
Workstation time (utc: 1204638024) Mon Mar 27 17:35:30 2014
                        * local      * remote
failover mode           * active     * passive
failover uptime         * 3d  5:26   * 0d  0:00
total cpu               * 80%        * 0%
total memory            * 2044104 kB * 2044104 kB
free memory             * 1672360 kB * 1672360 kB
failover state          * works      * works
failover cpu            * 7%         * 0%
```

```
iplir state              * works      * works
iplir cpu                * 0%         * 0%
mftp state               * works      * works
mftp cpu                 * 66%        * 0%
alg state                * works      * stopped
alg cpu                  * 35%        * 0%
webgui state             * works      * stopped
webgui cpu               * 0%         * 0%
```

Если демон failoverd не работает, выводится только информация о его режиме. Например:

```
Failover is in single mode
```

Журнал переключений

События о работе системы защиты от сбоев в режиме кластера горячего резервирования, записываются демоном failoverd в журнал переключений. Среди таких событий:

- <BOOT> — запуск демона failoverd при загрузке операционной системы;
- <P_START> — ручной запуск демона failoverd в пассивном режиме;
- <A_START> — ручной запуск демона failoverd в активном режиме;
- <SWITCH> — переход сервера из пассивного режима работы в активный (или наоборот).

Для просмотра записей в журнале переключений сервера, выполните команду:

```
hostname> failover view <дд.мм.гггг[.чч.мм.сс]> <дд.мм.гггг[.чч.мм.сс]>,
```

указав начало и конец интервала времени, записи за который вас интересуют. Например:

```
failover view 08.11.2016.00.00.00 23.11.2016.16.37.00
```

В результате выполнения команды выводится следующая информация:

- версии ПО ViPNet Coordinator HW и демона failoverd;
- идентификатор и имя сервера;
- указание на работу системы защиты от сбоев в режиме кластера горячего резервирования;
- локальные дата и время на сервере;
- список записей за выбранный интервал времени (дата, время, идентификатор и описание события).

Данная информация выводится в следующем формате:

```
View journal of failover switching
Versions: ViPNet 4.2.0 (30), daemon 1.5 (1)
Workstation configured for ID 29A0022 (Coordinator_HW)
The workstation works in a cluster mode of protection against failures
Workstation time (utc: 1174916969) Mon Mar 27 17:49:29 2014

09 Mar 2014 12:51:42     <P_START> Start failover daemon in passive mode
22 Mar 2014 12:27:27     <A_START> Start failover daemon in active mode
22 Mar 2014 14:10:35     <A_START> Start failover daemon in active mode
```

```
22 Mar 2014 15:30:46     <BOOT> Boot the system
23 Mar 2014 11:09:07     <SWITCH> Switch server from passive mode to active mode
```

Если за указанный интервал времени не произошло ни одного события, выводится следующее сообщение:

```
There are no records in journal of switchings
```

Работа кластера горячего резервирования совместно с коммутационным оборудованием

На практике часто применяются схемы подключения кластера горячего резервирования к коммутаторам, маршрутизаторам и другому коммутационному оборудованию. Настройки данного оборудования могут влиять на работу кластера горячего резервирования. В частности, администратор такого оборудования может настроить блокирование тех или иных сетевых пакетов, среди которых могут оказаться служебные пакеты, необходимые для корректного функционирования ViPNet Coordinator HW в режиме кластера горячего резервирования.

В связи с этим при использовании коммутационного оборудования с кластером горячего резервирования необходимо убедиться в следующем:

- На оборудовании пропускаются эхо-запросы ICMP с IP-адресов активного сервера до всех узлов с IP-адресами, заданными в параметрах `testip` секции `[channel]` файла `failover.ini`, и ответы на них.
- На оборудовании отключена функция Proxy ARP и пропускаются ARP-запросы с IP-адресов пассивного сервера до IP-адресов активного сервера и ответы на них.

Данные рекомендации касаются всех сетевых интерфейсов ViPNet Coordinator HW, для которых существует секция `[channel]` в файле `failover.ini`.

Организация обеспечения электропитания от UPS

Примечание. Приведенные в разделе настройки справедливы только для аппаратных исполнений ViPNet Coordinator HW. Для исполнения ViPNet Coordinator HW VA данная функция не поддерживается.

Чтобы обеспечить электропитание ViPNet Coordinator HW в случае отключения электроэнергии или перебоев в электросети, рекомендуется подключить его к источнику бесперебойного питания (см. глоссарий, стр. 268).

Внимание! В настоящее время поддерживаются только UPS фирмы APC, подключаемые через USB-порт (например, APC Smart-UPS).

Если ViPNet Coordinator HW подключен к UPS, при отключении электроэнергии выполняются следующие действия:

1. ViPNet Coordinator HW начинает питаться от аккумуляторной батареи UPS.
2. Если заряд аккумуляторной батареи достигает критически низкого уровня (этот уровень зависит от следующих параметров, заданных на UPS: `battery.charge`, `battery.charge.low`, `battery.runtime` и `battery.runtime.low`), выполняются следующие действия:
   - **2.1** UPS взаимодействует с компьютером по интерфейсному кабелю и посылает сигнал об истощении батареи.
   - **2.2** ViPNet Coordinator HW получает сигнал и начинает выключаться. При этом на UPS передается информация о том, что для выключения ViPNet Coordinator HW потребуется не менее 180 секунд. Получив эти данные, UPS корректирует расход оставшегося заряда батареи таким образом, чтобы ViPNet Coordinator HW успел штатно завершить работу.

     В результате ViPNet Coordinator HW успевает выключиться до полного разряда батареи и потери данных не происходит.
3. Если подача электроэнергии восстанавливается до достижения батареей критически низкого уровня заряда, ViPNet Coordinator HW снова начинает питаться от электросети.

В общем случае от одного UPS могут питаться несколько компьютеров. Компьютер, к которому подключен интерфейсный кабель UPS, является главным (master), остальные компьютеры — подчиненными (slave). Главный компьютер взаимодействует непосредственно с UPS и отвечает за своевременное оповещение по сети подчиненных компьютеров. Главный компьютер должен быть доступен подчиненным по одному из своих IP-адресов. На рисунке ниже представлена схема подключения нескольких компьютеров к одному UPS.

Рисунок 5. Схема подключения компьютеров к UPS

В частном случае, когда к UPS подключен только один компьютер, подчиненные компьютеры отсутствуют.

Если к UPS подключается кластер горячего резервирования, состоящий из двух ViPNet Coordinator HW, то один ViPNet Coordinator HW будет главным, а другой — подчиненным. При этом никакие другие компьютеры к UPS подключать нельзя. Для связи между двумя ViPNet Coordinator HW кластера можно использовать только резервный канал, так как на нем IP-адреса неизменны при переходе ViPNet Coordinator HW из активного режима кластера в пассивный и наоборот.

Чтобы настроить взаимодействие нескольких ViPNet Coordinator HW с UPS, необходимо на каждом ViPNet Coordinator HW выполнить ряд команд, соответствующих его роли (master или slave).

На ViPNet Coordinator HW, который будет выступать в роли master, выполните следующие действия:

1. Подключите к ViPNet Coordinator HW интерфейсный кабель UPS.
2. Выполните команду `enable` для перехода в режим администратора. В ответ на приглашение введите пароль администратора.
3. Включите взаимодействие ViPNet Coordinator HW с UPS с помощью команды:

   ```
   hostname# ups set monitoring on
   ```
4. Установите на ViPNet Coordinator HW режим master с помощью команды:

   ```
   hostname# ups set mode master
   ```
5. Запустите взаимодействие ViPNet Coordinator HW с UPS с помощью команды:

   ```
   hostname# ups start
   ```
6. Проверьте взаимодействие ViPNet Coordinator HW с UPS с помощью команды:

   ```
   hostname# ups show status
   ```

   Если взаимодействие установлено, отобразится информация о текущем состоянии UPS.

Внимание! Если вы переподключили интерфейсный кабель UPS (то есть отсоединили его от ViPNet Coordinator HW, а затем снова подключили), то для восстановления взаимодействия ViPNet Coordinator HW с UPS выполните последовательно команды `ups stop` и `ups start`.

На ViPNet Coordinator HW, который будет выступать в роли slave, выполните следующие действия:

1. Выполните команду `enable` для перехода в режим администратора. В ответ на приглашение введите пароль администратора.
2. Включите взаимодействие ViPNet Coordinator HW с UPS с помощью команды:

   ```
   hostname# ups set monitoring on
   ```

3 Установите на ViPNet Coordinator HW режим slave с помощью команды:

```
hostname# ups set mode slave <master_IP>
```

ViPNet Coordinator HW, выступающий в роли master, должен быть доступен с данного ViPNet Coordinator HW по этому IP-адресу.

При подключении к UPS кластера горячего резервирования в качестве IP-адреса мастера укажите адрес первого ViPNet Coordinator HW на резервном канале.

4 Запустите взаимодействие ViPNet Coordinator HW с UPS с помощью команды:

```
hostname# ups start
```

5 Проверьте взаимодействие ViPNet Coordinator HW с UPS с помощью команды:

```
hostname# ups show status
```

Если взаимодействие установлено, отобразится информация о текущем состоянии UPS, подключенного к ViPNet Coordinator HW в роли master.

Также вы можете настроить автоматическое включение ViPNet Coordinator HW после появления электропитания. Это может понадобиться в случае долговременного отключения электроэнергии. Чтобы выполнить такую настройку, задайте в BIOS параметры, приведенные в таблице ниже.

Таблица 7. Параметры настройки BIOS для автоматического включения ViPNet Coordinator HW после появления электропитания

Аппаратная платформа ViPNet Coordinator HW	Пункт меню/подменю	Параметр	Значение
HW100 X1, X2, X3, X8	Integrated Peripherals > SuperIO Device	PWRON After PWR-Fail	On
HW1000 Q2, Q3	Power > APM Configuration	Restore on AC/Power Loss	Power On
HW2000 Q2, Q3	Advanced > Power On Configuration	Restore on AC Power Loss	Power On
HW1000 Q4, Q5, Q6, HW2000 Q4, HW5000 Q1	Advanced > APM	Restore AC Power Loss	Power On

Примечание. Для аппаратных платформ HW50 N1, N2, N3, N4 и HW100 N1, N2, N3 автоматическое включение после появления электропитания настроить нельзя.

7

Настройка сетевых фильтров

Основные принципы фильтрации трафика

Фильтрации подвергается весь трафик, который проходит через сетевой узел:

- открытый (нешифрованный) трафик;
- защищенный (зашифрованный) трафик;
- туннелируемый трафик.

Рисунок 6. Виды IP-трафика

Наибольшую опасность представляет трафик из открытой сети, поскольку в случае атаки достаточно сложно обнаружить ее источник и принять оперативные меры по ее пресечению.

И открытый, и защищенный трафик может быть локальным или широковещательным. Под локальным трафиком понимается входящий или исходящий трафик конкретного узла (то есть когда сетевой узел является отправителем или получателем IP-пакетов). Под широковещательным трафиком имеется в виду передача узлом IP-пакетов, у которых IP-адрес или MAC-адрес назначения является широковещательным адресом (то есть когда пакеты передаются всем узлам определенного сегмента сети).

Кроме этого, через координатор может проходить транзитный трафик. Координатор логически не является ни отправителем, ни получателем транзитных IP-пакетов, такие пакеты следуют через него на другие узлы.

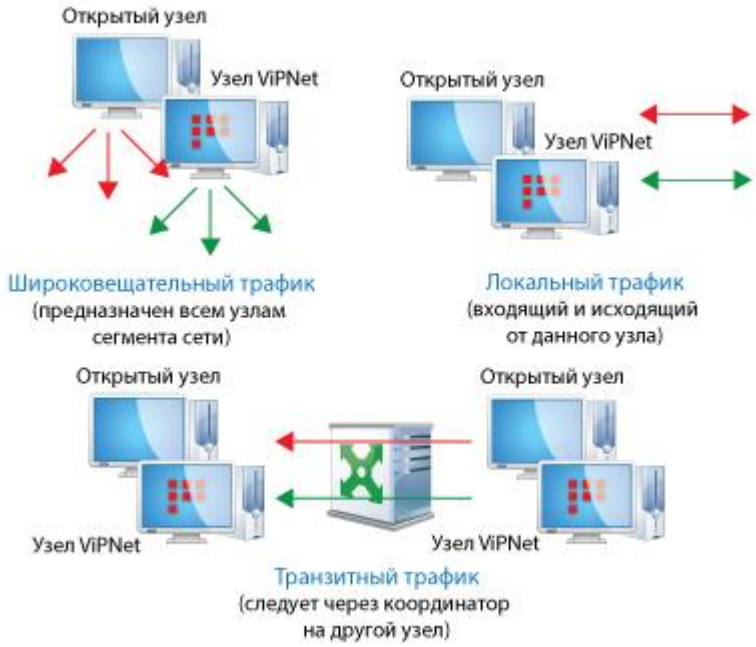

Рисунок 7. Виды защищенного и открытого трафика

Для того чтобы правильно настроить сетевые фильтры, необходимо понимать основные принципы фильтрации трафика:

- Все открытые IP-пакеты, в том числе те, которые передаются между координаторами и туннелируемыми ресурсами, проверяются в соответствии с правилами антиспуфинга, если они настроены.

  Если IP-пакет имеет адрес отправителя, разрешенный правилом антиспуфинга, он пропускается, в противном случае — блокируется.

- Все входящие и исходящие открытые и зашифрованные IP-пакеты проверяются в соответствии с условиями сетевых фильтров в порядке убывания приоритета этих фильтров (см. «Общие сведения о сетевых фильтрах» на стр. 116).

- Если IP-пакет соответствует условию одного из фильтров, то он пропускается или блокируется этим фильтром.

- Если IP-пакет был пропущен или заблокирован одним из фильтров, то фильтры с более низким приоритетом никак на него не влияют.

- Если IP-пакет был пропущен одним из фильтров, то ответные IP-пакеты в рамках текущего соединения будут пропускаться автоматически.

- Если IP-пакет не был обработан ни одним фильтром, то он блокируется фильтром по умолчанию.

- Вновь созданные фильтры влияют как на новые, так и на уже существующие соединения. То есть, если фильтр, блокирующий трафик какого-либо соединения, добавлен после установления этого соединения, то соединение будет разорвано.

Для того чтобы правильно настроить фильтры открытой сети (см. «Общие сведения о сетевых фильтрах» на стр. 116), необходимо понимать схему фильтрации открытого трафика в ViPNet Coordinator HW (см. Рисунок 8 на стр. 115):

1. Все входящие IP-пакеты проверяются на предмет фрагментации, если включена соответствующая настройка межсетевого экрана (см. «Настройка дополнительных параметров межсетевого экрана» на стр. 148).
2. Если IP-пакет фрагментирован, то он блокируется. Если IP-пакет не фрагментирован, то для него выполняется трансляция адреса назначения (Destination NAT) (см. «Трансляция адреса назначения» на стр. 137), если создано соответствующее правило трансляции адресов.
3. После трансляции адреса назначения IP-пакет проходит проверку встроенным средством антиспуфинга, если оно включено (см. «Настройка антиспуфинга» на стр. 145).
4. В случае успешной проверки встроенным средством антиспуфинга для IP-пакета возможны следующие варианты дальнейшего пути:
   - Если IP-пакет относится к протоколу HTTP (транспортный протокол TCP, порт 80) и прокси-сервер включен:
     - IP-пакет проходит проверку на уровне фильтрации содержимого трафика (см. «Настройка фильтрации содержимого трафика» на стр. 170).
     - В случае успешной проверки IP-пакет проходит антивирусную проверку, если она включена (см. «Настройка антивируса» на стр. 168).
     - В случае успешной антивирусной проверки IP-пакет пропускается.
   - Если IP-пакет не относится к протоколу HTTP или прокси-сервер выключен:
     - IP-пакет проходит проверку сетевыми фильтрами (см. «Общие сведения о сетевых фильтрах» на стр. 116).
     - В случае успешной проверки сетевыми фильтрами для IP-пакета выполняется трансляция адреса источника (Source NAT), если создано соответствующее правило трансляции адресов (см. «Трансляция адреса источника» на стр. 138).
     - Затем IP-пакет пропускается.

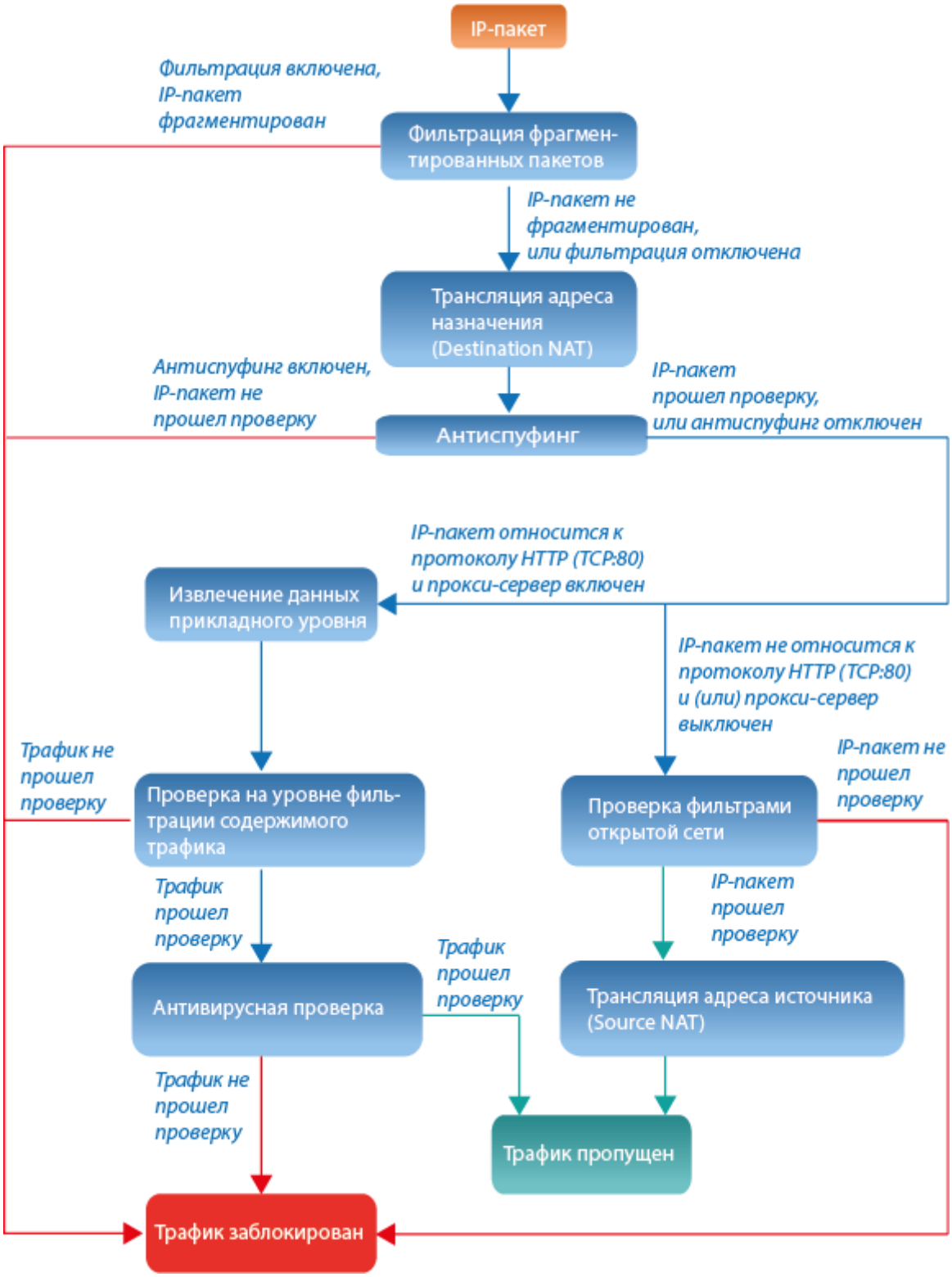

Рисунок 8. Схема фильтрации открытого трафика ViPNet Coordinator HW

Общие сведения о сетевых фильтрах

Различаются сетевые фильтры для защищенного трафика, для открытого трафика (локального и транзитного) и для туннелируемого трафика. Они выполняют следующие функции:

- Фильтры открытой сети могут разрешать либо блокировать обмен IP-трафиком с открытыми узлами.

Примечание. Открытыми узлами (см. глоссарий, стр. 270) называются узлы, на которых не установлено программное обеспечение ViPNet с функцией шифрования трафика. К ним относятся также компьютеры с программным обеспечением ViPNet CryptoService и ViPNet Registration Point.

- Фильтры защищенной сети могут разрешать или блокировать обмен IP-трафиком с защищенными узлами ViPNet, с которыми данный узел имеет связь.
- Фильтры для туннелируемого трафика могут разрешать или блокировать IP-пакеты, передаваемые между туннелируемыми узлами и узлами сети ViPNet, с которыми данный координатор имеет связь.

Внимание! Работа с фильтрами туннелируемых узлов возможна только при наличии лицензии на туннелирование хотя бы одного соединения.

Различаются следующие сетевые фильтры:

- Служебные фильтры, включающие:
  - Фильтры, разрешающие входящий и исходящий IP-трафик для служб ViPNet.
  - Фильтры, разрешающие открытый IP-трафик, который используется для проверки работоспособности сетевых интерфейсов в режиме кластера горячего резервирования (см. глоссарий, стр. 268).
- Фильтры, поступившие в составе политик безопасности. Политика безопасности представляет собой набор параметров, регулирующих безопасность сетевого узла. Она формируется администратором сети ViPNet в программе ViPNet Policy Manager (см. глоссарий, стр. 265), может включать сетевые фильтры и правила трансляции адресов и рассылается на узлы с помощью транспортного модуля MFTP. При получении политики безопасности из программы ViPNet Policy Manager она немедленно применяется на узле.
- Предустановленные фильтры и фильтры, заданные пользователем. Предустановленные фильтры разрешают только некоторые типы IP-пакетов (см. «Сетевые фильтры по умолчанию» на стр. 234). Для работы с какими-либо дополнительными сервисами пользователю необходимо создать соответствующие фильтры.

- Блокирующий фильтр по умолчанию.

Примечание. Если ViPNet Coordinator HW версии 4.x обновлялся с версии 3.x, вместо предустановленных фильтров на узле будут присутствовать фильтры, которые использовались до обновления в сконвертированном формате (подробнее см. в документе «ViPNet Coordinator HW. Подготовка к работе»).

Служебные фильтры создаются в ViPNet Coordinator HW автоматически, имеют самый высокий приоритет, то есть применяются в первую очередь, и недоступны для редактирования. После служебных фильтров применяются фильтры, поступившие в составе политик безопасности из программы ViPNet Policy Manager. Эти фильтры также недоступны для редактирования. Затем применяются предустановленные фильтры и фильтры, заданные пользователем. Их можно изменять и удалять. Наименьший приоритет имеет блокирующий фильтр по умолчанию, который нельзя ни изменить, ни удалить.

Последовательность применения сетевых фильтров в порядке убывания приоритета представлена ниже.

Рисунок 9. Последовательность применения сетевых фильтров

Сетевые фильтры могут включать в себя следующие параметры:

- условие — адрес отправителя и получателя IP-пакетов, на которые действует фильтр, и протоколы, используемые для передачи этих IP-пакетов (например, TCP, UDP, ICMP);
- расписание применения фильтра — ежедневно, еженедельно или по календарю;
- действие, применяемое к IP-пакетам — пропускать или блокировать IP-пакеты, соответствующие условию фильтра.

О том, как просмотреть фильтры, заданные на узле, создать или изменить фильтры на узле см. в соответствующем разделе ниже.

Группы объектов

Группы объектов позволяют упростить процессы создания и изменения сетевых фильтров и правил трансляции сетевых адресов в ViPNet Coordinator HW. Каждая группа объединяет несколько объектов одного типа (например, IP-адреса или сетевые интерфейсы). Группы можно указывать при задании параметров фильтра или правила трансляции вместо перечисления отдельных объектов.

В зависимости от типа объединяемых объектов различаются следующие группы:

- Группа сетевых узлов ViPNet — содержит любую комбинацию узлов ViPNet, используется в фильтрах защищенной сети и фильтрах туннелируемых узлов.
- Группа IP-адресов — содержит любую комбинацию IP-адресов, диапазонов IP-адресов и DNS-имен, используется в фильтрах открытой сети, фильтрах туннелируемых узлов и правилах трансляции адресов.
- Группа сетевых интерфейсов — содержит любую комбинацию сетевых интерфейсов, используется в фильтрах открытой сети и фильтрах туннелируемых узлов.
- Группа протоколов — содержит любую комбинацию сетевых протоколов и портов, используется в фильтрах открытой и защищенной сетей, фильтрах туннелируемых узлов и правилах трансляции адресов.
- Группа расписания — содержит любую комбинацию параметров, определяющих время действия фильтра, используется в фильтрах открытой и защищенной сетей, фильтрах туннелируемых узлов и правилах трансляции адресов.

Каждая группа объектов относится к одному из следующих видов:

- Системные группы объектов (на стр. 119) — настроенные по умолчанию группы с фиксированными именами, которые могут использоваться для задания адресов отправителей и получателей IP-пакетов в фильтрах (см. «Создание сетевого фильтра» на стр. 124) и правилах трансляции адресов, а также при создании пользовательских групп объектов (см. «Создание группы объектов» на стр. 121). Такие группы объектов не отображаются в списках групп (см. «Просмотр групп объектов» на стр. 122), их нельзя ни изменить, ни удалить.
- Группы объектов из программы ViPNet Policy Manager (см. глоссарий, стр. 265) — группы, получаемые в составе политик безопасности, и используемые в соответствующих фильтрах (см. «Общие сведения о сетевых фильтрах» на стр. 116). Такие группы нельзя ни удалять, ни изменять, ни использовать для задания параметров пользовательских фильтров и групп объектов.
- Пользовательские группы объектов — группы, создаваемые пользователем на узле, а также несколько групп, настроенных по умолчанию (см. «Пользовательские группы объектов по умолчанию» на стр. 120). Такие группы можно использовать для задания параметров фильтров (см. «Создание сетевого фильтра» на стр. 124) и правил трансляции, а также при создании других пользовательских групп объектов (см. «Создание группы объектов» на стр. 121). При необходимости их можно удалить.

Системные группы объектов

В таблице ниже описаны доступные системные группы объектов и указано соответствие имен системных групп объектов ViPNet Coordinator HW в командном интерпретаторе и веб-интерфейсе.

Таблица 8. Системные группы объектов

Имя группы объектов в командном интерпретаторе ViPNet (в веб-интерфейсе)	**Описание**
`allclients` (Все клиенты)	Все клиенты (см. глоссарий, стр. 268), с которыми у узла есть связь. Можно использовать в фильтрах защищенной сети и фильтрах туннелируемых узлов для задания адреса отправителя для входящих соединений узла или адреса получателя для исходящих соединений узла.
`allcoordinators` (Все координаторы)	Все координаторы (см. глоссарий, стр. 269), с которыми у узла есть связь. Можно использовать в фильтрах защищенной сети и фильтрах туннелируемых узлов для задания адреса отправителя для входящих соединений узла или адреса получателя для исходящих соединений узла.
`broadcast` (Широковещательные адреса)	Все широковещательные адреса. Можно использовать в фильтрах защищенной и открытой сетей для идентификации широковещательных адресов получателей. Использование других адресов совместно в этой группой недопустимо.
`local` (Мой узел)	Собственный узел. Можно использовать в фильтрах защищенной и открытой сетей для задания адреса отправителя для исходящих соединений узла или адреса получателя для входящих соединений узла. Использование других адресов совместно в этой группой недопустимо.
`remote` (Другие узлы)	Другие узлы (любые, кроме собственного). Можно использовать в фильтрах защищенной и открытой сетей для задания адреса отправителя для входящих соединений узла или адреса получателя для исходящих соединений узла. Использование других адресов совместно в этой группой недопустимо.
`tunneledip` (Туннелируемые IP-адреса)	Все IP-адреса, туннелируемые координатором. Можно использовать в фильтрах туннелируемых узлов для задания адреса отправителя для входящих соединений узла или адреса получателя для исходящих соединений узла.

Имя группы объектов в командном интерпретаторе ViPNet (в веб-интерфейсе)	Описание
`multicast` (Групповые адреса)	Диапазон адресов для групповой рассылки (224.0.0.0 – 239.255.255.255). Можно использовать в фильтрах открытой сети для задания адреса получателя. Использование других адресов совместно в этой группой недопустимо.

Кроме того, в сетевых фильтрах и правилах трансляции адресов может использоваться объект `any`, обозначающий любое значение, то есть:

- для локальных и транзитных фильтров открытой сети — все IP-адреса;
- для фильтров защищенной сети — все узлы сети ViPNet;
- для фильтров туннелируемых узлов — все IP-адреса или все узлы сети ViPNet;
- для протоколов — все протоколы;
- для расписаний — все время (постоянно).

Использование других условий совместно с объектом `any` недопустимо.

Пользовательские группы объектов по умолчанию

По умолчанию в ViPNet Coordinator HW настроены следующие пользовательские группы объектов:

- Группы IP-адресов:
  - PrivateNetworkIP (частные IP-адреса) — включает IP-адреса локальных сетей: 10.0.0.0/8; 172.16.0.0/12; 192.168.0.0/16.
  - InternetIP (публичные IP-адреса) — включает все IP-адреса, за исключением частных IP-адресов.
- Группы протоколов, которые наиболее часто используются при создании сетевых фильтров. Они перечислены в приложении (см. «Пользовательские группы протоколов по умолчанию» на стр. 238).
- Группы расписаний:
  - Workdays (рабочие дни) — включает рабочие дни недели (с понедельника по пятницу).
  - Weekends (выходные дни) — включает выходные дни недели (субботу и воскресенье).

Примечание. О том, как просмотреть пользовательские группы объектов, созданные на узле, см. в разделе Просмотр групп объектов (на стр. 122).

Создание группы объектов

Чтобы создать группу объектов, выполните команду:

`hostname# firewall <тип объектов> add name @<имя> <состав> [exclude <исключения>]`, где:

- параметр `<тип объектов>` может принимать одно из следующих значений:
  - `ip-object` — IP-адреса;
  - `vpn-object` — ViPNet-узлы;
  - `interface-object` — сетевые интерфейсы;
  - `service-object` — протоколы;
  - `schedule-object` — расписания;
- имя группы должно начинаться с символа «`@`», не содержать пробелов и быть уникальным в рамках групп объектов одного типа;
- в параметрах `<состав>` и `<исключения>` указываются объекты входящие и не входящие в группу соответственно, причем:
  - IP-адреса и идентификаторы ViPNet-узлов указываются через запятую или в виде диапазона (подробнее см. в разделах Адрес отправителя (на стр. 126) и Адрес получателя (на стр. 128));
  - сетевые интерфейсы разделяются пробелом, и перед именем каждого сетевого интерфейса необходимо указать слово `interface`;
  - протоколы или расписания указываются в соответствии с синтаксисом, описанным в разделах Протокол (на стр. 129) и Расписание (на стр. 130).

Примечание. Если при вводе команды была допущена синтаксическая ошибка, то в результате выполнения такой команды появится сообщение с указанием слова, содержащего ошибку. Исправьте ошибку и выполните команду еще раз.

В результате будет создана группа объектов, и вы сможете использовать ее для задания параметров сетевых фильтров (см. «Создание сетевого фильтра» на стр. 124) и правил трансляции адресов (см. «Создание правила трансляции адресов» на стр. 140).

Примеры создания групп объектов

Чтобы создать группу IP-адресов, включающую сегмент сети за исключением нескольких IP-адресов, выполните команду:

```
hostname# firewall ip-object add name @IP_group_1 110.35.14.0/24 exclude
110.35.14.3,110.35.14.13
```

Чтобы создать группу расписания, включающую выходные дни с 9 до 23 часов, выполните команду:

```
hostname# firewall schedule-object add name @weekend weekly sa su at 09:00-23:00
```

Чтобы создать группу интерфейсов, включающую сетевые интерфейсы eth0 и eth1, выполните команду:

```
hostname# firewall interface-object add name @intgroup interface eth0 interface eth1
```

Просмотр групп объектов

Чтобы просмотреть пользовательские группы объектов и группы объектов, полученные из программы ViPNet Policy Manager, выполните одну из команд:

- для просмотра всех групп объектов:

  ```
  hostname> firewall object show
  ```

- для просмотра групп объектов определенного типа (см. «Создание группы объектов» на стр. 121):

  ```
  hostname> firewall <тип объектов> show
  ```

  Например, для просмотра всех групп IP-адресов:

  ```
  hostname> firewall ip-object show
  ```

В результате выполнения команды будет отображена таблица, содержащая те или иные группы объектов. Пример таблицы групп IP-адресов представлен на рисунке ниже.

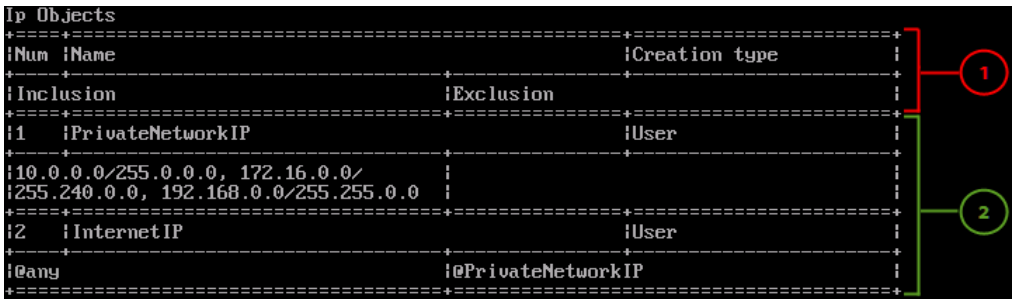

Рисунок 10. Просмотр групп IP-адресов

Цифрами обозначены:

1 Заголовок таблицы, где:

- `Num` — порядковый номер группы объектов.
- `Name` — имя группы объектов.
- `Creation type` — вид группы объектов: для групп из программы ViPNet Policy Manager — `Policy`, для пользовательских групп — `User`.

- `Inclusion` — состав группы объектов.
- `Exclusion` — исключения группы объектов.

2 Параметры групп объектов.

Удаление групп объектов

Вы можете удалить пользовательскую группу объектов, если она не используется в других группах объектов, сетевых фильтрах или правилах трансляции адресов.

Для удаления группы объектов выполните команду:

```
hostname# firewall object delete @<имя>
```

Примечание. При попытке удаления группы объектов, используемой в другой группе объектов, сетевом фильтре или правиле трансляции адресов, появится сообщение об ошибке со списком всех групп, фильтров и правил, которые используют данную группу. Удалите эти группы, фильтры и правила, затем выполните команду для удаления группы еще раз.

В результате группа объектов будет удалена, и вы больше не сможете использовать ее для задания параметров сетевых фильтров, правил трансляции адресов и при создании других групп объектов.

Создание сетевого фильтра

Для каждого вида трафика создаются отдельные сетевые фильтры, которые хранятся в соответствующей таблице (см. «Просмотр сетевых фильтров» на стр. 131).

Чтобы создать сетевой фильтр, выполните команду:

`hostname# firewall <тип> add [<номер>] [rule <имя>] src <адрес отправителя> dst <адрес получателя> [<протокол>] [<расписание>] <действие>`, указав следующие параметры:

- тип фильтра:
  - `local` — локальный фильтр открытой сети;
  - `forward` — транзитный фильтр открытой сети;
  - `vpn` — фильтр защищенной сети;
  - `tunnel` — фильтр туннелируемых узлов;

Примечание. Если в программе ViPNet Центр управления сетью (ЦУС) (см. глоссарий, стр. 265) запрещено использование туннелирования на узле ViPNet Coordinator HW, то на этом узле по умолчанию не будет фильтров туннелируемых узлов.

После получения разрешения на использование туннелирования создайте следующий фильтр:

`hostname# firewall tunnel add src @any dst @any pass`

- Порядковый номер фильтра в таблице, определяющий его приоритет. Чем меньше порядковый номер фильтра, тем выше его приоритет (наиболее приоритетный фильтр имеет номер 1). При создании фильтра с порядковым номером, меньшим максимального номера в таблице фильтров, номера фильтров, следующих за этим фильтром, автоматически увеличиваются на единицу (то есть их приоритет понижается).
- Имя фильтра. Если имя состоит из нескольких слов, разделенных пробелом, заключите его в кавычки (например, `rule "number one"`).

Примечание. При создании сетевого фильтра можно не указывать его номер. В этом случае фильтру будет присвоен номер, следующий за номером последнего фильтра в соответствующей таблице, и он будет иметь самый низкий приоритет.

Имя фильтра можно указывать без слова `rule`. Кроме этого, имя фильтра можно вообще не указывать, фильтр будет создан без имени.

- Параметры, определяющие процесс обработки IP-пакетов этим фильтром. Подробное описание этих параметров см. в соответствующих разделах:
  - Адрес отправителя (на стр. 126).

- Адрес получателя (на стр. 128).
- Протокол (на стр. 129).
- Расписание (на стр. 130).
- Действие (на стр. 130).

Примечание. Если при вводе команды была допущена синтаксическая ошибка, то в результате выполнения такой команды появится сообщение с указанием слова, содержащего ошибку. Исправьте ошибку и выполните команду еще раз.

В результате будет создан фильтр заданного типа. Вы можете просмотреть (см. «Просмотр сетевых фильтров» на стр. 131), изменить (см. «Изменение сетевого фильтра» на стр. 133) или удалить его (см. «Удаление сетевого фильтра» на стр. 134).

Примеры создания сетевых фильтров

Чтобы разрешить отправку FTP-запросов со своего узла открытому узлу с адресом 192.168.2.4, выполните команду:

```
hostname# firewall local add src @local dst 192.168.2.4 service @FTP pass
```

Чтобы создать локальный фильтр, блокирующий IP-пакеты, отправляемые узлом с адресом 192.168.30.1 через порт 2525 на порт 443 открытого узла с адресом 172.16.35.1 по протоколу TCP/IP, выполните команду:

```
hostname# firewall local add src 192.168.30.1 dst 172.16.35.1 tcp sport 2525 dport 443
drop
```

Чтобы создать фильтр защищенной сети, блокирующий пакеты от защищенного узла с идентификатором 0x1a12000a в защищенную сеть с идентификатором 0x1b14, выполните команду:

```
hostname# firewall vpn add src 0x1a12000a dst 0x1b14 drop
```

Чтобы создать фильтр, разрешающий отправку открытых транзитных IP-пакетов от узла с адресом 192.168.0.1 на узел с адресом 192.168.30.3 через координатор, выполните команду:

```
hostname# firewall forward add src 192.168.0.1 dst 192.168.30.3 pass
```

Чтобы создать фильтр, разрешающий отправку IP-пакетов от туннелируемого узла с адресом 192.168.0.1 на защищенный узел с идентификатором 0x1234abab, выполните команду:

```
hostname# firewall tunnel add src 192.168.0.1 dst 0x1234abab pass
```

Чтобы создать фильтр, разрешающий отправку IP-пакетов от туннелируемого узла с адресом 192.168.0.1 туннелируемому узлу с адресом 192.168.2.3, выполните команду:

```
hostname# firewall tunnel add src 192.168.0.1 dst 192.168.2.3 pass
```

Блокирование веб-сайтов по доменным именам с помощью сетевых фильтров

Для корректной работы запрещающих сетевых фильтров, в которых в качестве адресов отправителей или получателей заданы доменные имена, выполните следующие действия:

3 Задайте для ViPNet Coordinator HW IP-адрес внешнего DNS-сервера с помощью команды (для примера используется IP-адрес 8.8.8.8):

```
hostname# inet dns forwarders add 8.8.8.8
```

4 Создайте разрешающие локальные фильтры для протокола DNS с помощью команд:

```
hostname# firewall local add 1 rule "Allow DNS 1" src 8.8.8.8 dst @local service @DNS pass
```

```
hostname# firewall local add 1 rule "Allow DNS 2" src @local dst 8.8.8.8 service @DNS pass
```

Эти фильтры разрешат обмен трафиком между ViPNet Coordinator HW и внешним DNS-сервером.

5 Создайте запрещающие сетевые фильтры для протокола DNS с помощью команд:

```
hostname# firewall forward add 2 rule "Deny DNS forward" src @any dst @any service @DNS drop
```

```
hostname# firewall vpn add 2 rule "Deny DNS vpn" src @any dst @any service @DNS drop
```

```
hostname# firewall tunnel add 2 rule "Deny DNS tunnel" src @any dst @any service @DNS drop
```

Эти фильтры запретят обмен трафиком между клиентам защищенной сети и любым DNS-сервером, кроме заданного для ViPNet Coordinator HW.

6 Для клиентов защищенной сети в качестве DNS-сервера задайте IP-адрес ViPNet Coordinator HW.

После этого вы можете создавать запрещающие сетевые фильтры для доменных имен веб-сайтов, например:

```
hostname# firewall forward add src @any dst facebook.com drop
```

Адрес отправителя

Адрес отправителя IP-пакетов является обязательным параметром сетевого фильтра и описывается лексемой:

```
src <адрес отправителя>
```

Возможные значения адреса отправителя:

- для локальных фильтров открытой сети:
  - IP-адрес или доменное имя узла;

Примечание. В ViPNet Coordinator HW версий 4.x при задании адреса отправителя в виде доменного имени на DNS-сервер отправляется соответствующий запрос для разрешения данного имени. DNS-запись, полученная в ответ на такой запрос, сохраняется в кэше и используется до истечения времени ее жизни. Актуальность данных в кэше обеспечивается регулярной отправкой запросов на DNS-сервер.

Если разрешения указанного в фильтре доменного имени по каким-либо причинам не произошло, то такой фильтр не загружается в драйвер ViPNet до тех пор, пока в результате отправки повторных запросов от DNS-сервера не будет получен корректный ответ.

- диапазон IP-адресов узлов — два ограничивающие диапазон IP-адреса, разделенные дефисом. При этом второй адрес (конец диапазона) должен быть больше первого (начала диапазона), например:

  `src 192.168.1.1-192.168.1.10`

- маска адресов подсети — адрес подсети в формате классовой или бесклассовой адресации CIDR (Classless Internet Domain Routing), например:

  `src 192.168.1.0/24` или `src 192.168.1.0/255.255.255.0`

- доменное имя узла, например:

  `src mydomain.ru`

Примечание. При необходимости вы можете задать несколько IP-адресов и доменных имен отправителей IP-пакетов в одной лексеме, перечислив их через запятую, например: `src 192.168.30.2,192.169.1.1,mydomain.ru`

- системная группа объектов `any`, `local` или `remote` либо одна или несколько пользовательских групп IP-адресов (см. «Группы объектов на стр. 118»). Имя группы необходимо дополнить символом «@», несколько групп указываются через запятую. Например:

  `src @IP_group_1,@IP_group_2`

Внимание! Использование других адресов отправителей IP-пакетов совместно с системными группами объектов `any`, `local` и `remote` недопустимо.

- сетевой интерфейс собственного узла, через который будут проходить IP-пакеты, в следующем виде:

  `src interface {<системное имя> | @<имя группы интерфейсов> | byip {<IP-адрес> | <диапазон IP-адресов> | <маска адресов>}}`

- для транзитных фильтров открытой сети: то же, что для локальных фильтров открытой сети, но в таких фильтрах нельзя использовать системные группы объектов;
- для фильтров защищенной сети:
  - идентификатор узла ViPNet, например:

```
src 0x1a12000a
```

- системные группы объектов `any`, `allcoordinators`, `allclients`, `local` или `remote` либо пользовательские группы узлов ViPNet (см. «Группы объектов» на стр. 118). Имя группы необходимо дополнить символом «`@`», несколько групп указываются через запятую. Например:

  ```
  src @allclients,@allcoordinators
  ```

- идентификатор сети ViPNet, например:

  ```
  src 0x1a12
  ```

Примечание. При задании адреса отправителя для фильтров защищенной сети:

- В одной лексеме можно указать несколько идентификаторов узлов или сетей ViPNet, пользовательских групп узлов ViPNet и системную группу `allcoordinators` или `allclients`, перечислив их через запятую. Например: `src @allcoordinators,0x1a12000a`
- Недопустимо указывать IP-адреса, доменные имена, маски адресов и сетевой интерфейс узла.
- Недопустимо использовать другие адреса отправителей совместно с группами объектов `any`, `local` и `remote`.

- для фильтров туннелируемых узлов: то же, что для локальных фильтров открытой сети и фильтров защищенной сети, но в таких фильтрах можно использовать только системные группы объектов `any`, `allcoordinators`, `allclients`, `tunneledip`.

Внимание! Если в фильтре туннелируемых узлов в качестве адреса отправителя указан адрес защищенного узла, то в качестве адреса получателя в этом фильтре должен быть указан адрес открытого узла, и наоборот.

Адрес получателя

Адрес получателя IP-пакетов является обязательным параметром сетевого фильтра. Синтаксис адреса получателя повторяет синтаксис адреса отправителя (см. «Адрес отправителя» на стр. 126), за исключением следующих особенностей:

- Адрес получателя описывается после адреса отправителя лексемой:

  ```
  dst <адрес получателя>
  ```

- Для задания адреса получателя нельзя использовать сетевой интерфейс узла.
- Для задания адреса получателя, помимо пользовательских групп объектов, можно использовать следующие системные группы объектов (на стр. 119):

  - для локальных фильтров открытой сети: `any`, `local`, `remote`, `broadcast`, `multicast;`

Внимание! Использование других адресов отправителей IP-пакетов совместно с системными группами объектов `any`, `local`, `remote`, `broadcast` и `multicast` недопустимо.

- для фильтров защищенной сети: `any`, `allcoordinators`, `allclients`, `local`, `remote`, `broadcast`.
- для фильтров туннелируемых узлов: `any`, `allcoordinators`, `allclients`, `tunneledip`.

Протокол

Протокол не является обязательным параметром сетевого фильтра.

Протоколы можно указывать, используя:

- имена протоколов, написанные строчными буквами и разделенные пробелами, например:

  `tcp udp icmp`

  В этом случае также можно задать дополнительные параметры для протоколов, например:

  - для протоколов TCP и UDP: `sport` (порт или диапазон портов источника пакета) и (или) `dport` (порт или диапазон портов назначения пакета). При использовании обоих этих параметров сначала необходимо указать параметр `sport`, затем — параметр `dport`. Например: `tcp sport 1024-65535 dport 1024`;
  - для протокола ICMP: `type` (тип пакета) и/или `code` (код пакета). При использовании обоих этих параметров сначала необходимо указать параметр `type`, затем — параметр `code`. Если параметр `type` не задан, то под условие будут попадать все ICMP-пакеты указанного типа, например: `icmp code 12`;
- номера протоколов. В этом случае для каждого протокола необходимо указать ключевое слово `proto` и его номер. Например, для протоколов TCP, UDP и ICMP: `proto 6 proto 17 proto 1`

Внимание! При использовании номеров протоколов задать дополнительные параметры протоколов невозможно.

- пользовательские группы протоколов, в том числе заданные по умолчанию (см. «Пользовательские группы объектов по умолчанию» на стр. 120), в виде:

  `service @<имя группы>`

  Например: `service @DNS`

Расписание

Расписание задает время действия фильтра и не является обязательным параметром сетевого фильтра.

Примечание. Если расписание задано, то оно действует в соответствии с зоной UTC, установленной на ViPNet Coordinator HW. Если расписание не задано, то фильтр действует постоянно.

Расписание описывается одной из следующих лексем:

- `daily <чч:мм>-<чч:мм>` — фильтр действует ежедневно в течение заданного интервала времени. Время указывается в 24-часовом формате: `чч` — часы, `мм` — минуты.
- `weekly [mo] [tu] [we] [th] [fr] [sa] [su] [at <чч:мм>-<чч:мм>]` — фильтр действует еженедельно в заданные дни недели (`mo` — понедельник, `tu` — вторник, `we` — среда, `th` — четверг, `fr` — пятница, `sa` — суббота, `su` — воскресенье) и интервал времени.
- `calendar <дд.мм.гггг>-<дд.мм.гггг> [at <чч:мм>-<чч:мм>]` — фильтр действует в заданные даты и интервал времени. Дата указывается в следующем формате: `дд` — день, `мм` — месяц, `гггг` — год.
- `schedule <имя группы объектов>` — фильтр действует по расписанию, описанному группой объектов соответствующего типа.

Для задания расписания можно использовать соответствующие пользовательские группы, в том числе заданные по умолчанию (см. «Пользовательские группы объектов по умолчанию» на стр. 120).

Действие

Действие является обязательным параметром сетевого фильтра и определяет, что делает фильтр с IP-пакетом, соответствующим его условию.

Действие описывается одной из следующих лексем:

- `pass` — пропускать IP-пакет;
- `drop` — блокировать IP-пакет.

Просмотр сетевых фильтров

Вы можете просмотреть сетевые фильтры, заданные на узле (см. «Общие сведения о сетевых фильтрах» на стр. 116). Для этого выполните одну из команд:

- для просмотра всех фильтров:

  ```
  hostname> firewall rules show
  ```

  Примечание. В результате выполнения команды `firewall rules show` отображаются не только сетевые фильтры, заданные на узле, но и правила трансляции адресов (см. «Просмотр, изменение, удаление правил трансляции адресов» на стр. 142).

- для просмотра сетевых фильтров определенного типа (см. «Создание сетевого фильтра» на стр. 124):

  ```
  hostname> firewall <тип> show
  ```

  Например, для просмотра локальных фильтров открытой сети:

  ```
  hostname> firewall local show
  ```

- для просмотра сетевых фильтров с конкретными параметрами:

  ```
  hostname> firewall <тип> show <параметр>,
  ```

  указав в качестве параметра порядковый номер, имя, адрес отправителя (на стр. 126), адрес получателя (на стр. 128), протокол (на стр. 129) или действие (на стр. 130) фильтра. При необходимости вы можете указать несколько параметров.

  Например, для просмотра локального фильтра открытой сети с несколькими адресами отправителей или одним адресом получателя выполните команду:

  ```
  hostname> firewall local show src 192.168.30.2,192.169.1.1,mydomain.ru dst
  192.168.0.1
  ```

  Внимание! Поиск сетевых фильтров для отображения осуществляется по строгому совпадению с указанными параметрами. Например:

  - если указан параметр `src 192.168.1.10`, отобразится фильтр, содержащий адрес `src 1.1.1.1,192.168.1.10,2.2.2.2`, но не отобразится фильтр, содержащий адрес `src 192.168.1.10-192.168.1.11`.
  - если указан параметр `rule Name`, отобразится фильтр с именем `Name`, но не отобразится фильтр с именем `Name1`.

В результате выполнения команды будет отображена таблица, содержащая те или иные сетевые фильтры. Пример таблицы локальных фильтров открытой сети представлен на рисунке ниже.

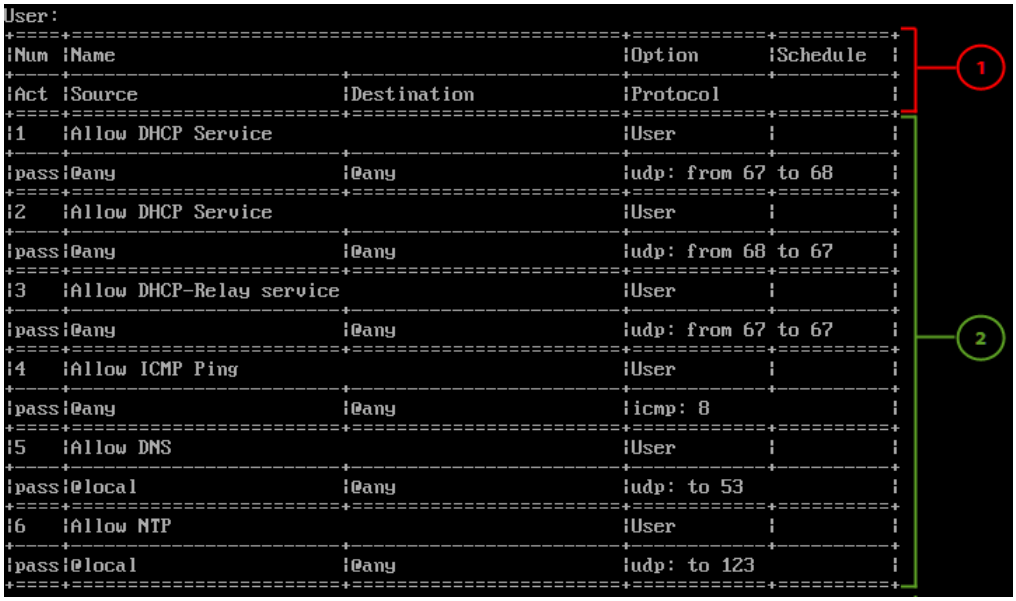

Рисунок 11. Просмотр локальных фильтров открытой сети

Цифрами на рисунке обозначены:

1. Заголовок таблицы, где:
   - `Num` — порядковый номер фильтра, определяющий его приоритет.
   - `Name` — имя фильтра.
   - `Option` — категория сетевого фильтра: для фильтров из программы ViPNet Policy Manager — `Policy`, для предустановленных фильтров и фильтров пользователя — `User`, для фильтров системы защиты от сбоев — `Failover`.
   - `Schedule` — расписание применения фильтра.
   - `Act` — действие фильтра.
   - `Source` — адрес отправителя.
   - `Destination` — адрес получателя.
   - `Protocol` — протокол.
2. Параметры фильтров, включающие:
   - строку с порядковым номером, именем и категорией фильтра;
   - одну или несколько строк, содержащих адрес отправителя, адрес получателя, протокол, расписание и действие фильтра.

Изменение сетевого фильтра

Вы можете отредактировать ранее созданный сетевой фильтр, добавив в него адрес отправителя, адрес получателя, протокол или расписание либо изменив его приоритет (порядковый номер в таблице фильтров соответствующего типа).

Чтобы добавить в фильтр адрес отправителя, адрес получателя, протокол или расписание:

1 Выполните команду:

```
hostname# firewall <тип> change append <номер> <параметр>,
```

указав в качестве параметра адрес отправителя (на стр. 126), адрес получателя (на стр. 128), протокол (на стр. 129) или расписание (на стр. 130). При необходимости вы можете указать несколько параметров.

Примечание. Если при вводе команды была допущена синтаксическая ошибка, то в результате выполнения такой команды появится сообщение с указанием слова, содержащего ошибку. Исправьте ошибку и выполните команду еще раз.

2 Подтвердите операцию вводом символа `y` и нажатием клавиши **Enter**. В результате новый адрес отправителя, адрес получателя, протокол или расписание будет добавлено к уже имеющимся в фильтре.

Чтобы изменить приоритет фильтра, в интерпретаторе ViPNet выполните команду:

```
hostname# firewall <тип> move rule <текущий номер> to <новый номер>
```

В результате приоритет фильтра и, соответственно, его порядковый номер в таблице будут изменены. При этом номера фильтров, следующих за этим фильтром, автоматически увеличатся на единицу, то есть их приоритет понизится (см. «Просмотр сетевых фильтров» на стр. 131).

Пример добавления условия в сетевой фильтр

Для добавления в фильтр

```
1 RuleName1 src 192.168.1.1,2.2.2.2/24 dst @local drop
```

нового адреса отправителя, выполните команду:

```
hostname# firewall local change append 1 src 192.168.3.3
```

В результате фильтр будет изменен следующим образом:

```
1 RuleName1 src 192.168.1.1,2.2.2.2/24,192.168.3.3 dst @local drop
```

Удаление сетевого фильтра

Чтобы удалить ненужный сетевой фильтр, выполните следующие действия:

- Если вы знаете порядковый номер фильтра, который нужно удалить, выполните команду:

  ```
  hostname# firewall <тип> delete <номер>
  ```

  Например, для локального фильтра открытой сети с порядковым номером 1:

  ```
  hostname# firewall local delete 1
  ```

- Если вы не знаете номер фильтра, который нужно удалить, то найдите фильтр по известным вам параметрам, а затем удалите его. Для этого:

  - Выполните команду:

    ```
    hostname# firewall <тип> delete <параметр>,
    ```

    указав в качестве параметра имя, адрес отправителя (на стр. 126), адрес получателя (на стр. 128), протокол (на стр. 129) или действие (на стр. 130) фильтра. При необходимости вы можете указать несколько параметров.

    Например, для локальных фильтров открытой сети с адресом отправителя 192.168.20.2:

    ```
    hostname# firewall local delete src 192.168.20.2
    ```

    В результате будет отображен список сетевых фильтров, которые содержат указанные параметры.

    

    Внимание! Поиск сетевых фильтров для отображения осуществляется по строгому совпадению с указанными параметрами.

  - Если найдено несколько фильтров, в списке найдите фильтр, который нужно удалить, и введите его номер, указанный в столбце `Num`. Затем подтвердите удаление вводом символа `y` и нажатием клавиши **Enter**.
  - Если найдет один фильтр, подтвердите его удаление вводом символа `y` и нажатием клавиши **Enter**.

В результате сетевой фильтр будет удален. При этом номера фильтров, имеющих более низкий приоритет по сравнению с удаленным фильтром, автоматически уменьшатся на единицу (см. «Просмотр сетевых фильтров» на стр. 131).

8

Настройка правил трансляции адресов

Трансляция адресов в технологии ViPNet

Трансляция сетевых адресов (NAT) — это механизм преобразования IP-адресов одной сети в IP-адреса другой сети. Технология трансляции адресов описана в RFC 2663 (http://tools.ietf.org/html/rfc2663).

ViPNet Coordinator HW может выполнять трансляцию адресов следующих типов:

- Трансляция адреса источника (на стр. 138), называемая также маскарадингом (masquerading) или динамической трансляцией.

  В этом случае при прохождении через координатор пакетов от отправителей с частными адресами в них заменяется адрес отправителя на внешний (реальный) адрес координатора. При получении ответных пакетов в них подменяется адрес получателя обратно на частный адрес, и в таком виде пакет доставляется в частную сеть.

  Используется для подключения локальной сети к Интернету, когда количество узлов локальной сети превышает выданное поставщиком услуг Интернета количество публичных IP-адресов. В результате локальные сети, использующие частные адреса, получают доступ к ресурсам Интернета.

- Трансляция адреса назначения (на стр. 137), называемая также форвардингом портов (port forwarding) или статической трансляцией.

  В этом случае пакеты, приходящие из Интернета на определенный порт внешнего адреса координатора, перенаправляются на указанный адрес внутренней сети путем подмены в них адреса получателя, а у ответных пакетов от компьютера внутренней сети подменяется адрес отправителя.

  Используется для организации доступа к ресурсам локальной сети из Интернета. В результате локальные сети, использующие частные адреса, могут быть доступны пользователям Интернета по публичным IP-адресам.

- Одновременная трансляция адресов источника и назначения. Может использоваться в сложных схемах маршрутизации трафика.

Трансляция сетевых адресов выполняется координатором, только если настроены соответствующие правила. Координатор должен иметь как минимум два сетевых интерфейса:

- внешний интерфейс — имеет публичный IP-адрес и обеспечивает доступ в Интернет;
- внутренний интерфейс — имеет частный IP-адрес.

Внимание! Правила трансляции, описанные в данном разделе, относятся только к открытому трафику. Для защищенного трафика действуют автоматические механизмы трансляции адресов, параметры которых не могут быть изменены.

Трансляция адреса назначения

Трансляция адреса узла назначения предназначена для организации доступа из Интернета к серверам локальной сети, не имеющим публичного IP-адреса. Правило трансляции адреса назначения ставит в соответствие частным IP-адресам локальных узлов публичный IP-адрес координатора. В соответствии с правилом, в заголовках IP-пакетов публичный IP-адрес (или IP-адрес и порт) назначения заменяется частным адресом локальной сети. Таким образом, по публичному IP-адресу внешние пользователи могут получить доступ к ресурсам локальной сети.

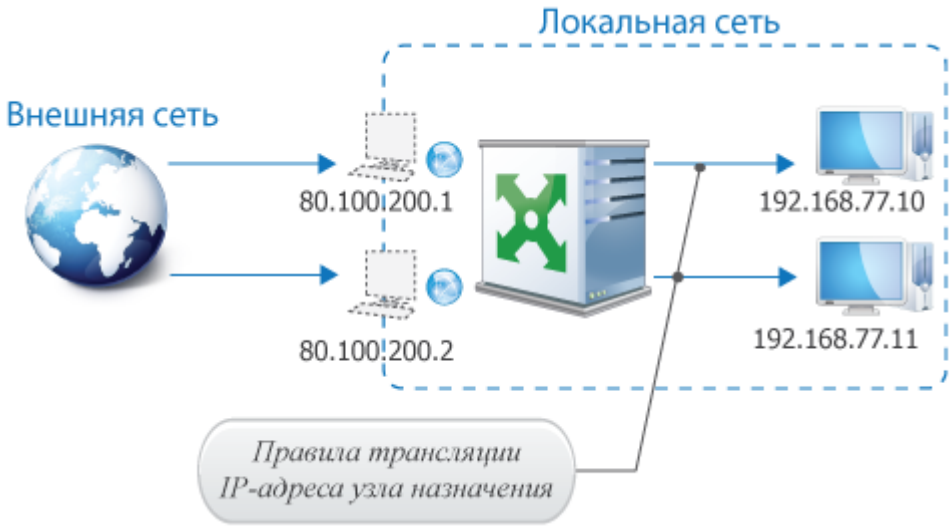

Рисунок 12. Доступ к внутренним ресурсам при помощи правил трансляции IP-адресов узлов назначения

Если для внешнего IP-адреса координатора задано правило трансляции адреса назначения, то при обращении к этому адресу из Интернета будут выполняться следующие преобразования:

- Во входящих IP-пакетах от внешнего узла координатор подменяет адрес получателя (публичный IP-адрес координатора) локальным адресом в соответствии с описанным правилом. Затем пакет передается через внутренний сетевой интерфейс на узел локальной сети, которому адресован пакет.
- При прохождении ответных пакетов (в рамках уже созданной сессии) координатор производит обратную замену IP-адресов. Адрес отправителя (IP-адрес локального узла) подменяется публичным IP-адресом внешнего сетевого интерфейса координатора. Затем ответный пакет отправляется по назначению (узлу в Интернете).

  Таким образом, при передаче в Интернете пакет выглядит так, будто отправитель и получатель этого пакета имеют публичные IP-адреса.

Внимание! При трансляции адреса узла назначения инициировать соединение может только внешний узел. Чтобы локальный узел мог также иметь доступ в Интернет (двусторонний NAT), необходимо в дополнение к правилу трансляции адреса узла назначения задать также правило трансляции адреса источника (см. «Трансляция адреса источника» на стр. 138).

Трансляция адреса источника

Трансляция адреса источника предназначена для организации доступа локальных компьютеров в Интернет. Правило трансляции адреса источника ставит в соответствие нескольким частным IP-адресам локальных узлов публичный IP-адрес координатора. В соответствии с правилом, в заголовках IP-пакетов частные IP-адреса источника заменяются на публичный IP-адрес. Таким образом, узлы локальной сети могут устанавливать соединения с узлами в Интернете от имени публичного IP-адреса координатора.

Рисунок 13. Организация доступа в Интернет при помощи правила трансляции IP-адреса источника

Если на координаторе настроено правило трансляции адреса источника, то транзитные IP-пакеты, проходящие через координатор из локальной сети в Интернет (или другие глобальные сети), будут преобразованы следующим образом:

- В момент передачи IP-пакета из локальной сети в Интернет координатор преобразует адрес и (или) порт отправителя пакета для протоколов TCP и UDP. Для пакетов протокола ICMP преобразуется адрес отправителя, остальные параметры запоминаются. В процессе преобразования частный адрес отправителя пакета заменяется на публичный адрес внешнего сетевого интерфейса координатора, обеспечивающего доступ в глобальную сеть. При дальнейшей передаче в Интернете пакет имеет публичный IP-адрес отправителя. Номера портов отправителя (для протоколов TCP и UDP) и запоминаемые параметры (для протокола ICMP) пакетов имеют уникальные значения для всех исходящих IP-соединений внешнего сетевого интерфейса координатора. После преобразования пакет отправляется адресату в Интернете.
- При прохождении ответных пакетов координатор производит обратное преобразование указанных параметров. То есть в момент передачи ответного IP-пакета координатор заменяет в нем адрес получателя на частный адрес узла локальной сети, которому адресован ответный пакет. Преобразование происходит на основании уникальных номеров портов, присвоенных исходящим пакетам (для протоколов TCP и UDP), и запоминаемых параметров исходящих пакетов (для протокола ICMP). Номера портов (для протоколов TCP и UDP) также

преобразуются в свои истинные значения. Затем ответные пакеты передаются через внутренний сетевой интерфейс узлу локальной сети, которому адресован пакет.

Примечание. Для всех протоколов, кроме TCP, UDP и ICMP, преобразуются только IP-адреса. Для протоколов с частичным преобразованием трансляция IP-адреса источника не будет работать, если несколько узлов локальной сети одновременно инициируют соединение с одним и тем же IP-адресом публичной сети.

Создание правила трансляции адресов

Правила трансляции адресов хранятся в соответствующей таблице (см. «Просмотр, изменение, удаление правил трансляции адресов» на стр. 142).

Примечание. Правила трансляции IP-адресов работают только при наличии транзитных фильтров открытой сети (см. «Создание сетевого фильтра» на стр. 124), пропускающих трафик от узлов, к которым будет применяться трансляция адресов.

Чтобы создать правило трансляции адресов, выполните команду:

`hostname# firewall nat add [<номер>] [rule <имя>] src <адрес отправителя> dst <адрес получателя> [<протокол>] [<расписание>] change {<адрес отправителя> | <адрес получателя>}`, указав следующие параметры:

- Порядковый номер правила в таблице, определяющий его приоритет. Чем меньше порядковый номер правила, тем выше его приоритет (наиболее приоритетное правило имеет номер 1). При создании правила с порядковым номером, меньшим максимального номера в таблице правил, номера правил, следующих за этим правилом, автоматически увеличиваются на единицу (то есть их приоритет понижается).
- Имя правила. Если имя состоит из нескольких слов, разделенных пробелом, заключите его в кавычки (например, `rule "number one"`).

Примечание. При создании правила трансляции адресов можно не указывать его номер. В этом случае правилу будет присвоен номер, следующий за номером последнего правила в таблице, и оно будет иметь самый низкий приоритет.

Имя правила можно указывать без слова `rule`. Кроме этого, имя правила можно вообще не указывать, правило будет создано без имени.

- Параметры, определяющие процесс обработки трафика правилом:
  - адрес отправителя (на стр. 126) и адрес получателя (на стр. 128) являются обязательными параметрами правила и задаются в виде IP-адреса или доменного имени узла, диапазона IP-адресов узлов, списка IP-адресов или доменных имен узлов, маски адресов подсети, доменного имени сети, одной или нескольких пользовательских групп IP-узлов;
  - протокол (на стр. 129) и расписание (на стр. 130) являются необязательными параметрами правила и задаются так же, как для сетевых фильтров;
  - действие правила в виде одной из лексем:
    - для трансляции адреса источника: `change src {<адрес> | auto}`,

указав внешний адрес координатора, на который будет заменяться адрес отправителя пакетов либо параметр `auto`, чтобы адрес отправителя автоматически заменялся на публичный адрес внешнего сетевого интерфейса координатора.

- для трансляции адреса и/или порта назначения: `change dst [<адрес>]:[<порт>]`,

  указав адрес и/или порт узла локальной сети, которому будут перенаправляться пакеты. В случае, если указан только порт назначения, адрес назначения не будет изменен.

Примечание. Если при вводе команды была допущена синтаксическая ошибка, то в результате выполнения такой команды появится сообщение с указанием слова, содержащего ошибку. Исправьте ошибку и выполните команду еще раз.

В результате будет создано правило трансляции адресов. Вы можете просмотреть, изменить или удалить его (см. «Просмотр, изменение, удаление правил трансляции адресов» на стр. 142).

Примеры создания правил трансляции адресов

Чтобы при отправке пакета узлом с адресом 10.0.0.1 внешнему узлу с адресом 192.168.20.1 частный адрес отправителя пакета заменялся на публичный адрес внешнего сетевого интерфейса координатора, создайте правило трансляции адреса источника с помощью команды:

```
hostname# firewall nat add src 10.0.0.1 dst 192.168.20.1 change src auto
```

Чтобы при отправке пакета внешним узлом с адресом mydomain.ru узлу с адресом 192.168.20.1 координатор подменял адрес получателя (публичный IP-адрес координатора) на локальный адрес, создайте правило трансляции адреса назначения с помощью команды:

```
hostname# firewall nat add src mydomain.ru dst 192.168.20.1 change dst 10.0.0.7
```

Чтобы при отправке пакета внешним узлом с адресом mydomain.ru узлу с адресом 192.168.20.1 по протоколу TCP/IP через порт 8080 координатор подменял адрес получателя (публичный IP-адрес координатора) на локальный адрес, создайте правило трансляции адреса назначения с помощью команды:

```
hostname# firewall nat add src mydomain.ru dst 192.168.20.1 tcp dport 8080 change dst
10.0.0.7:8080
```

Чтобы одновременно транслировать адреса источника и назначения, например, от адреса 10.0.2.15 до адреса 192.168.1.2, выполните команду:

```
firewall nat add src 10.0.2.15 dst 192.168.1.2 change src auto dst 10.0.2.15
```

Просмотр, изменение, удаление правил трансляции адресов

Вы можете просмотреть, изменить или удалить правила трансляции адресов, заданные на узле.

Чтобы просмотреть правила, выполните одну из команд:

- для просмотра всех правил:

  `hostname> firewall nat show`

- для просмотра правил с конкретными параметрами:

  `hostname> firewall nat show <параметр>`,

  указав в качестве параметра порядковый номер, имя, адрес отправителя, адрес получателя, протокол или действие правила. При необходимости вы можете указать несколько параметров.

Внимание! Поиск правил трансляции адресов для отображения осуществляется по строгому совпадению с указанными параметрами.

В результате выполнения команды будет отображена таблица, содержащая те или иные правила трансляции адресов. Пример такой таблицы показан на рисунке ниже.

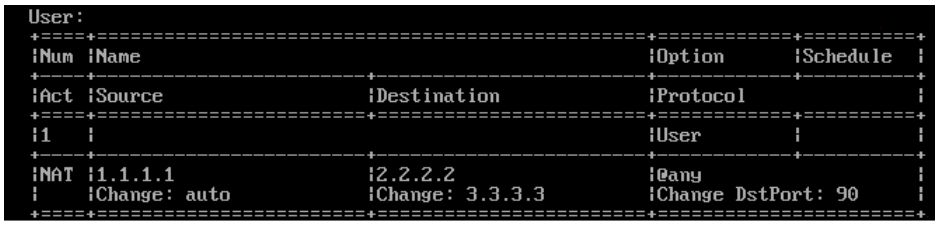

Рисунок 14: Просмотр правил трансляции адресов

Таблица содержит следующие столбцы:

- `Num` — порядковый номер правила, определяющий его приоритет.
- `Name` — имя правила.
- `Option` — категория правила: для правил из программы ViPNet Policy Manager — `Policy`, для правил пользователя — `User`.
- `Schedule` — расписание применения правила.
- `Act` — действие правила (`NAT`).
- `Source` — адрес источника до или после трансляции.
- `Destination` — адрес назначения до или после трансляции.
- `Protocol` — протокол.

Изменение и удаление правил трансляции адресов выполняется аналогично изменению и удалению сетевых фильтров с помощью команд:

```
hostname# firewall nat change append <номер> <параметр>
```

```
hostname# firewall nat delete <параметр>
```

Подробнее см. в разделах Изменение сетевого фильтра (на стр. 133) и Удаление сетевого фильтра (на стр. 134).

9

Тонкая настройка межсетевого экрана

Настройка антиспуфинга

В ViPNet Coordinator HW реализована функция антиспуфинга, то есть блокирования входящих IP-пакетов от отправителей, IP-адреса которых недопустимы на данном сетевом интерфейсе узла. Антиспуфинг работает только для открытого трафика, поскольку для защищенного трафика IP-адрес отправителя не имеет значения. Открытые пакеты сначала проверяются правилами антиспуфинга, а затем уже обрабатываются сетевыми фильтрами (см. «Общие сведения о сетевых фильтрах» на стр. 116).

Основная задача антиспуфинга — это защита от сетевых атак, называемых спуфингом. При спуфинге злоумышленник посылает на какой-либо компьютер IP-пакет, в котором в качестве адреса отправителя указан не адрес злоумышленника, а адрес какого-либо другого узла, которому разрешено соединение с данным координатором. Например, таким образом можно отправить открытый пакет из Интернета через координатор, задав в качестве адреса отправителя адрес частной внутренней сети, которая также подключена к данному координатору. Правила антиспуфинга позволяют исключить такую возможность.

По умолчанию антиспуфинг в ViPNet Coordinator HW выключен. Для повышения уровня безопасности сети рекомендуется включить его, то есть задать для каждого сетевого интерфейса узла диапазоны IP-адресов, пакеты от которых недопустимы на данном интерфейсе. Пакеты с адресами, попадающими в такой диапазон, будут блокироваться.

Чтобы просмотреть текущее состояние антиспуфинга, в интерпретаторе ViPNet выполните команду:

```
hostname> iplir option get antispoofing
```

Чтобы включить (`on`) или выключить (`off`) антиспуфинг в ViPNet Coordinator HW, выполните команду:

```
hostname# iplir option set antispoofing {on | off}
```

При включении антиспуфинга на сетевом узле на основе таблицы маршрутизации автоматически задаются правила антиспуфинга.

Внимание! В случае использования сложных схем маршрутизации на узле (с метриками маршрутов или асимметричными маршрутами) включать функцию антиспуфинга не рекомендуется, так как она может работать некорректно.

Правила антиспуфинга применяются для открытого транзитного трафика следующим образом:

- Для всех сетевых интерфейсов, кроме интерфейса, соответствующего маршруту по умолчанию, блокируются IP-пакеты, адреса отправителей которых не совпадают с адресами, маршрутизируемыми через данный интерфейс.
- Для сетевого интерфейса, соответствующего маршруту по умолчанию, блокируются IP-пакеты, адреса отправителей которых совпадают с зарегистрированными маршрутами других интерфейсов.

Пример настройки антиспуфинга

Пусть на сетевом узле используется следующая таблица маршрутизации:

Таблица 9. Пример таблицы маршрутизации на узле

Сетевой адрес	Маска сети	Адрес шлюза	Интерфейс	Метрика
0.0.0.0	0.0.0.0	10.0.8.1	10.0.8.54	20
10.0.8.0	255.255.255.0	On-link	10.0.8.54	276
10.0.8.54	255.255.255.255	On-link	10.0.8.54	276
10.0.8.255	255.255.255.255	On-link	10.0.8.54	276
127.0.0.0	255.0.0.0	On-link	127.0.0.1	306
127.0.0.1	255.255.255.255	On-link	127.0.0.1	306
127.255.255.255	255.255.255.255	On-link	127.0.0.1	306
192.168.48.0	255.255.255.0	On-link	192.168.48.1	276
192.168.48.1	255.255.255.255	On-link	192.168.48.1	276
192.168.48.255	255.255.255.255	On-link	192.168.48.1	276
192.168.59.0	255.255.255.0	On-link	192.168.59.1	276
192.168.59.1	255.255.255.255	On-link	192.168.59.1	276
192.168.59.255	255.255.255.255	On-link	192.168.59.1	276
224.0.0.0	224.0.0.0	On-link	127.0.0.1	306
224.0.0.0	224.0.0.0	On-link	10.0.8.54	276
224.0.0.0	224.0.0.0	On-link	192.168.48.1	276
224.0.0.0	224.0.0.0	On-link	192.168.59.1	276
255.255.255.255	255.255.255.255	On-link	127.0.0.1	306
255.255.255.255	255.255.255.255	On-link	10.0.8.54	276
255.255.255.255	255.255.255.255	On-link	192.168.48.1	276
255.255.255.255	255.255.255.255	On-link	192.168.59.1	276

Рассматриваемый узел имеет четыре сетевых интерфейса. Сетевой интерфейс с адресом 127.0.0.1 является интерфейсом «внутренней петли» (loopback), поэтому не будем его учитывать.

При включении антиспуфинга на основе приведенной таблицы маршрутизации на узле сформируются следующие правила антиспуфинга:

- На сетевом интерфейсе с адресом 192.168.48.1 разрешены входящие пакеты только от узлов с IP-адресами 192.168.48.0/24.
- На сетевом интерфейсе с адресом 192.168.59.1 разрешены входящие пакеты только от узлов с IP-адресами 192.168.59.0/24.

- На сетевом интерфейсе с адресом 10.0.8.54 разрешены все входящие пакеты, кроме пакетов от узла с IP-адресами 192.168.48.0/24, 192.168.59.0/24.

Настройка дополнительных параметров межсетевого экрана

Чтобы посмотреть текущие настройки дополнительных параметров межсетевого экрана, выполните команду:

```
hostname> iplir option get <параметр>,
```

указав один из следующих параметров:

- `antispoofing` — включение или выключение антиспуфинга. Возможные значения:
  - `off` (по умолчанию) — антиспуфинг выключен.
  - `on` — антиспуфинг включен.
- `block-fragmented-packets` — включение или выключение блокирования входящих фрагментированных IP-пакетов, принимаемых по всем сетевым интерфейсам. Возможные значения:
  - `off` (по умолчанию) — фрагментированные пакеты пропускаются.
  - `on` — фрагментированные пакеты блокируются, в журнал регистрации IP-пакетов вносится соответствующая запись (см. «Просмотр журнала регистрации IP-пакетов» на стр. 212).
- `connection-ttl-ip` — время жизни соединений по протоколу IP при отсутствии активности в нем. Указывается в секундах, по умолчанию — 300 (5 минут).
- `connection-ttl-tcp` — время жизни соединений по протоколу TCP при отсутствии активности в нем. Указывается в секундах, по умолчанию — 1800 (30 минут).
- `connection-ttl-udp` — время жизни соединений по протоколу UDP при отсутствии активности в нем. Указывается в секундах, по умолчанию — 300 (5 минут).
- `max-connections` — максимальное количество параллельно установленных соединений. Значение данного параметра зависит от аппаратной платформы ViPNet Coordinator HW.

  Таблица 10. Значения параметра max-connections для аппаратных платформ ViPNet Coordinator HW

Аппаратная платформа ViPNet Coordinator HW	Объем оперативной памяти	Значение по умолчанию	Максимальное значение
HW50 N1, N2, N3	1,8 Гбайт	150000	150000
HW100 X1, X2	1 Гбайт	150000	150000
HW100 X3, X8, N1, N2, N3	2 Гбайт	150000	150000
HW1000 Q2, Q3, Q4, Q5, Q6	2 Гбайт	800000	1000000
HW2000 Q2, Q4	4 Гбайт	2500000	3000000

Аппаратная платформа ViPNet Coordinator HW	Объем оперативной памяти	Значение по умолчанию	Максимальное значение
HW2000 Q3	8 Гбайт	6000000	6500000
HW5000 Q1	8 Гбайт	6000000	6500000
HW VA	1 Гбайт	150000	150000

- `dynamic-ports` — диапазон портов, используемых для динамической трансляции IP-адресов. По умолчанию — 1025–65535.
- `dynamic-timeouts` — включение или выключение режима динамических тайм-аутов соединений. Возможные значения: `off` (по умолчанию) или `on`.

  Данный режим используется для противодействия флуд-атакам. Когда количество соединений достигает определенного процента от максимума, тайм-ауты всех соединений уменьшаются на определенную величину. Эта величина тем больше, чем ближе число соединений к максимуму. При этом тайм-ауты не уменьшаются ниже определенного минимума. Когда количество соединений уменьшается до определенного процента от максимального, значения тайм-аутов восстанавливаются до исходной величины.
- `block-other-protocols` — включение или выключение блокирования IP-пакетов, передаваемых по всем протоколам, кроме IP, ARP и RARP. Возможные значения:
  - `off` (по умолчанию) — пропускаются пакеты, передаваемые по всем проколам, при этом обрабатываются только IP-пакеты.
  - `on` — блокируются пакеты, передаваемые по всем протоколам, кроме IP, ARP и RARP. ARP- и RARP-пакеты пропускаются всегда, поскольку это необходимо для корректной работы протокола IP.
- `cleanup-interval` — время, по истечении которого производится удаление соединений с истекшим временем жизни. Указывается в десятых долях секунды, по умолчанию — 20.

Для изменения настройки какого-либо параметра межсетевого экрана в ViPNet Coordinator HW выполните команду:

```
hostname# iplir option set <параметр> <значение>
```

10

Настройка обработки прикладных протоколов

О прикладных протоколах

Функционирование внутренних сетевых сервисов (например, IP-телефонии, DNS- и FTP-служб) обеспечивается прикладными протоколами. При использовании прикладных протоколов IP-адреса часто передаются в теле IP-пакета. По этой причине, если в защищенной сети применяются виртуальные IP-адреса (см. глоссарий, стр. 267) или трансляция адресов, такие сервисы могут быть недоступны для защищенных узлов. Кроме того, при работе по некоторым протоколам, помимо управляющего соединения для отправки команд, между сервером и клиентом устанавливается дополнительное соединение для передачи данных через случайно выбранный порт. В результате IP-пакеты могут следовать на разные порты защищенного узла и будут всегда блокироваться, так как создать фильтр, пропускающий такие пакеты, будет невозможно.

Функция обработки прикладных протоколов позволяет решить описанные проблемы, так как обеспечивает:

- Подмену виртуального IP-адреса защищенного узла в теле IP-пакета на реальный (для всех протоколов).
- Подмену IP-адреса защищенного узла на транслируемый адрес при статической или динамической трансляции IP-адресов (только для протокола FTP).
- Включение сетевого фильтра, пропускающего IP-пакеты, для дополнительного соединения на случайно выбранный порт, который используется прикладным протоколом. При этом для установления управляющего соединения с открытыми узлами на узле необходимо задать соответствующие фильтры открытой сети.

Вы можете использовать эту функцию для всех видов трафика. Список поддерживаемых прикладных протоколов см. в разделе Поддерживаемые прикладные протоколы (на стр. 153).

Пример обработки протокола FTP

При передаче файлов между клиентом и сервером по протоколу FTP устанавливается два TCP-соединения:

- для отправки команд на сервер и получения ответов от него (управляющее соединение);
- для передачи данных (дополнительное соединение).

Соединение клиента с сервером может происходить в активном или пассивном режиме. В активном режиме клиент инициирует управляющее соединение со своего порта из диапазона 1024–65535 на 21-й порт сервера. Сервер подключается к порту клиента, с которого тот инициировал соединение, и инициирует соединение для передачи данных через свой 20-й порт. В пассивном режиме после установления управляющего соединения сервер сообщает клиенту случайный номер его порта из диапазона 1024–65535, по которому будет установлено соединение для передачи данных. То есть в активном режиме клиент принимает соединение для передачи данных от сервера, а в пассивном режиме он сам инициирует это соединение.

Для установления управляющего и дополнительного соединений в активном или пассивном режиме работы протокола FTP выполните следующие настройки:

- Создайте фильтр открытой сети, разрешающий исходящее TCP-соединение на 21-й порт FTP-сервера.
- Убедитесь, что для разрешения дополнительного соединения в активном режиме работы включена обработка протокола FTP (см. «Настройка параметров обработки прикладных протоколов» на стр. 154), которая активирует необходимый фильтр трафика.

Пример обработки протокола SIP

Протокол SIP предназначен для организации, модификации и завершения сеансов связи (например, мультимедийных конференций, телефонных соединений) и распределения мультимедийной информации.

Вызывающий SIP-клиент отправляет запрос (например, приглашение для начала сеанса связи, подтверждение приема ответа на запрос, завершение сеанса связи) вызываемому SIP-клиенту с указанием его SIP-адреса. В зависимости от способа установления соединения запрос направляется вызываемому клиенту либо напрямую, либо с участием прокси-сервера SIP, либо с участием сервера переадресации. Вызываемый клиент в зависимости от типа полученного запроса передает вызывающему клиенту ответ на запрос (например, информацию об ошибке при обработке запроса, об успешной обработке запроса, об отклонении входящего вызова).

Для установления сеанса связи между SIP-клиентами протокол SIP регламентирует установление соединений TCP и UDP через порт 5060.

Чтобы установить сеанс связи между SIP-клиентами, убедитесь, что включена обработка протокола SIP (см. «Настройка параметров обработки прикладных протоколов» на стр. 154), и создайте фильтр открытой сети, разрешающий входящее и исходящее соединение по протоколам TCP и UDP на порт 5060 (если хотя бы один из SIP-клиентов является открытым узлом).

Примечание. Некоторые SIP-клиенты, например SIP-клиент компании ZoiPer, могут устанавливать соединения TCP и UDP не через порт 5060, а через динамический порт, выбранный произвольно из определенного диапазона. Поэтому если хотя бы один из SIP-клиентов является открытым узлом, возможность использования динамических портов необходимо отключить в настройках SIP-клиента.

Между SIP-клиентами нельзя использовать статическую или динамическую трансляцию адресов.

Поддерживаемые прикладные протоколы

В ViPNet Coordinator HW можно настроить обработку следующих прикладных протоколов:

- DNS — для доступа к сетевым узлам по доменным именам.
- FTP — для передачи файлов между FTP-клиентом и FTP-сервером.

  Команды FTP-протокола EPRT и EPSV, используемые некоторыми FTP-клиентами, не поддерживаются ViPNet Coordinator HW.
- H.323 — для передачи мультимедийных данных в IP-сетях (IP-телефония, видео- и аудиоконференции).

  Передача мультимедийных данных с использованием протокола H.323 возможна только между телекоммуникационными серверами, туннелируемыми координаторами ViPNet Coordinator HW. При этом на координаторах ViPNet Coordinator HW должен быть разрешен весь туннелируемый трафик, поскольку между телекоммуникационными серверами могут устанавливаться дополнительные служебные соединения.
- SCCP — для передачи сообщений между Skinny-клиентами (проводными и беспроводными IP-телефонами Cisco) и сервером голосовой почты Cisco Unity и Cisco CallManager.

  Передача мультимедийных данных с использованием протокола SCCP возможна, если сервер голосовой почты туннелируется координатором ViPNet Coordinator HW.
- SIP — для установления соединений, включающих обмен мультимедийными данными между клиентами.

  Не поддерживаются SIP-клиенты и телекоммуникационные серверы, использующие при подключении протокол TLS, SIP-клиенты, использующие «компактные поля» (compact headers), а также SIP-клиенты, использующие сжатие данных в сообщениях SIP.

  Для некоторых SIP-клиентов могут не обрабатываться такие дополнительные данные, как телефонные справочники, информация о доступности абонента и другие.

Список поддерживаемых прикладных протоколов изменить нельзя.

Настройка параметров обработки прикладных протоколов

По умолчанию в ViPNet Coordinator HW включена обработка всех поддерживаемых прикладных протоколов для открытого и защищенного трафика (см. «Поддерживаемые прикладные протоколы» на стр. 153). Для обработки заданы наиболее часто используемые сетевые протоколы и порты.

Примечание. Для непрерывной работы прикладных сервисов при штатной работе ViPNet Coordinator HW не рекомендуется завершать работу демона algd.

При завершении работы демона algd, если запущен управляющий демон iplircfg, установленные до этого момента соединения могут продолжать функционировать до 5 минут. Установление новых соединений при выключенном демоне algd невозможно.

Чтобы просмотреть текущие настройки обработки прикладных протоколов, выполните команду:

```
hostname> alg show
```

В результате отобразится таблица, показанная ниже.

```
+==============+========+================================================+======+
|SERVICE       |PROTOCOL|PORTS                                           |ON/OFF|
+==============+========+================================================+======+
|FTP           |TCP     |21                                              |ON    |
+==============+========+================================================+======+
|DNS           |UDP     |53                                              |ON    |
+==============+========+================================================+======+
|H323          |TCP     |1720                                            |ON    |
+==============+========+================================================+======+
|H323          |UDP     |1719                                            |ON    |
+==============+========+================================================+======+
|SCCP          |TCP     |2000                                            |ON    |
+==============+========+================================================+======+
|SIP           |TCP     |5060,5080                                       |ON    |
+==============+========+================================================+======+
|SIP           |UDP     |5060,5080                                       |ON    |
+==============+========+================================================+======+
```

Рисунок 15. Просмотр параметров прикладных протоколов

Таблица содержит следующие столбцы:

- `Service` — обрабатываемый прикладной протокол.
- `Protocol` — сетевой протокол для обработки.
- `Ports` — порты для обработки.
- `On/Off` — состояние обработки: `on` — включена, `off` — выключена.

Чтобы настроить и включить обработку прикладного протокола, выполните команду:

`hostname# alg module <прикладной протокол> process <сетевой протокол> <порты> on`, указав:

- один из поддерживаемых прикладных протоколов: `dns`, `ftp`, `h323`, `sccp` или `sip`;
- один из сетевых протоколов для обработки выбранного прикладного протокола: `tcp` или `udp`;

> **Примечание.** Для некоторых прикладных протоколов не поддерживается обработка протоколом TCP или UDP. Например, для протокола FTP не поддерживается обработка протоколом UDP.

- порт, диапазон портов либо список портов или диапазонов портов, разделенных запятой.

> **Внимание!** Нулевое значение порта означает выключение обработки прикладного протокола для указанного сетевого протокола.

В результате на узле будет включена обработка прикладного протокола, и вы сможете работать с соответствующими приложениями.

> **Примечание.** Выбранные параметры обработки прикладных протоколов должны соответствовать параметрам, указанным в настройках соответствующих приложений (FTP-клиент, DNS-клиент, SIP-клиент и так далее).

Чтобы изменить ранее выбранные сетевой протокол или порты для обработки какого-либо прикладного протокола, необходимо настроить и включить обработку этого прикладного протокола заново. Например, если кроме стандартных портов для обработки данных, передаваемых по протоколу SIP, используются еще несколько портов для протокола UDP, то их можно добавить с помощью команды:

```
hostname# alg module sip process udp 10000,21-65 on
```

Чтобы выключить обработку прикладного протокола:

- для одного сетевого протокола — задайте для этого сетевого протокола порт, равный нулю, и включите обработку. Например:

  ```
  hostname# alg module sip process tcp 0 on
  ```

- для всех сетевых протоколов — выполните команду:

  ```
  hostname# alg module <прикладной протокол> process off
  ```

> **Внимание!** При выключении обработки прикладного протокола работа соответствующих сетевых сервисов на защищенном узле будет невозможна.

11

Настройка сетевых служб

Настройка параметров DHCP-сервера

В состав ПО ViPNet Coordinator HW входит DHCP-сервер, который может использоваться для динамического назначения IP-адресов сетевым узлам (DHCP-клиентам). Одновременно с выделением IP-адресов DHCP-сервер может назначать дополнительные параметры настройки клиентов, например, IP-адреса шлюза по умолчанию и WINS-серверов.

В качестве адресов DNS-сервера и NTP-сервера DHCP-сервер всегда предоставляет клиентам адрес интерфейса, с которым он работает. Клиенты, получившие от ViPNet Coordinator HW вместе с IP-адресом адреса DNS- и NTP-серверов, будут осуществлять запросы на разрешение имен и синхронизацию времени через соответствующие серверы, запущенные на ViPNet Coordinator HW. Если на ViPNet Coordinator HW эти серверы не запущены, то клиенты не смогут работать с DNS-именами и синхронизировать свое время.

Внимание! Использование ViPNet Coordinator HW в качестве DHCP-сервера возможно только при работе в одиночном режиме. Работа DHCP-сервера в режиме кластера горячего резервирования не поддерживается.

Чтобы запустить DHCP-сервер на одном из сетевых интерфейсов Ethernet, выполните следующие действия:

1 Укажите сетевой интерфейс Ethernet, на котором будет работать DHCP-сервер, с помощью команды:

```
hostname# inet dhcp server interface <имя интерфейса>
```

Примечание. Сетевой интерфейс для DHCP-сервера не должен быть дополнительным, а также должен быть включен и иметь статический IP-адрес.

2 Задайте диапазон IP-адресов для распределения сервером с помощью команды:

```
hostname# inet dhcp server range <начальный IP-адрес> <конечный IP-адрес>
```

Диапазон IP-адресов должен принадлежать сети интерфейса и не должен включать адрес самого интерфейса.

Примечание. DHCP-сервер может выделять клиентам любые диапазоны IP-адресов. Однако в локальной сети, маршрутизируемой в сеть Интернет, рекомендуется выделять IP-адреса только из диапазонов, установленных стандартом для частных сетей: `10.0.0.0/8`, `172.16.0.0/12`, `192.168.0.0/16`.

3 Чтобы задать IP-адрес шлюза по умолчанию, выполните команду:

```
hostname# inet dhcp server router <IP-адрес>
```

IP-адрес шлюза должен принадлежать сети интерфейса и не должен совпадать с адресом самого интерфейса.

4 Чтобы включить или выключить автоматический запуск DHCP-сервера при загрузке ViPNet Coordinator HW, выполните команду:

```
hostname# inet dhcp server mode {on | off}
```

По умолчанию автоматический запуск DHCP-сервера выключен.

Примечание. Перед тем как вы включаете или запускаете DHCP-сервер, убедитесь, что служба DHCP-relay выключена (см. «Настройка DHCP-relay» на стр. 159).

5 Чтобы запустить DHCP-сервер или завершить его работу, выполните команду:

```
hostname# inet dhcp server {start | stop}
```

6 Чтобы просмотреть текущие параметры DHCP-сервера, выполните команду:

```
hostname> inet show dhcp server
```

Настройка DHCP-relay

В состав ПО ViPNet Coordinator HW входит служба DHCP-relay, которая позволяет использовать ViPNet Coordinator HW в качестве агента DHCP-relay. Такая необходимость может возникнуть в случае, если сетевые узлы должны получать IP-адреса от удаленного DHCP-сервера, к которому они не могут обращаться напрямую. Агент DHCP-relay обеспечивает взаимодействие таких сетевых узлов с DHCP-сервером: он принимает от узлов DHCP-запросы и передает их DHCP-серверу. Ответы, полученные от DHCP-сервера, агент перенаправляет сетевым узлам.

С помощью агента DHCP-relay также можно организовать выделение IP-адресов узлам разветвленной сети, включающей несколько подсетей. В этом случае не нужно устанавливать в каждой подсети свой DHCP-сервер, а достаточно использовать только один DHCP-сервер.

ViPNet Coordinator HW может обслуживать в качестве агента DHCP-relay несколько локальных сетей. Это могут быть как сети, подключенные к разным физическим интерфейсам ViPNet Coordinator HW, так и виртуальные локальные сети, подключенные к одному физическому интерфейсу.

Узлы, которые будут обращаться к DHCP-серверу через ViPNet Coordinator HW, могут быть защищенными и открытыми. В свою очередь, DHCP-сервер может быть защищенным, открытым или туннелируемым узлом.

При настройке службы DHCP-relay следует учитывать следующие рекомендации и ограничения:

- На ViPNet Coordinator HW требуется дополнительно задать правило фильтрации транзитного трафика, разрешающее UDP-трафик по портам 67 и 68. Аналогичные правила для локального открытого и широковещательного трафика заданы по умолчанию (см. «Сетевые фильтры по умолчанию» на стр. 234).
- Если агент DHCP-relay перенаправляет запросы на DHCP-сервер, который туннелируется другим координатором ViPNet, то между ViPNet Coordinator HW, выступающим в качестве агента DHCP-relay, и координатором, туннелирующим DHCP-сервер, необходимо настроить видимость по реальным IP-адресам (см. «Настройка параметров видимости узлов» на стр. 77). Если между этими координаторами настроена видимость по виртуальным адресам, ответы от DHCP-сервера будут отправляться на реальный адрес ViPNet Coordinator HW, выступающего в качестве агента DHCP-relay, и не будут доставлены.
- Использование ViPNet Coordinator HW в качестве агента DHCP-relay возможно только при работе ViPNet Coordinator HW в одиночном режиме. Работа службы DHCP-relay в режиме кластера горячего резервирования не поддерживается.

Чтобы запустить службу DHCP-relay на сетевом интерфейсе Ethernet, выполните следующие действия:

1. Задайте список сетевых интерфейсов Ethernet, через которые DHCP-relay должен получать DHCP-запросы от клиентов:
   - Чтобы добавить сетевой интерфейс в список, выполните команду:

     ```
     hostname# inet dhcp relay add listen-interface <имя интерфейса>
     ```

На каждом добавляемом сетевом интерфейсе должен быть задан статический IP-адрес, принадлежащий адресному пространству соответствующей подсети. Сетевой интерфейс, который используется для соединения с удаленным сервером может также получать IP-адрес по DHCP.

- Чтобы удалить сетевой интерфейс из списка, выполните команду:

  ```
  hostname# inet dhcp relay delete listen-interface <имя интерфейса>
  ```

2 Задайте внешний DHCP-сервер, на который необходимо перенаправлять DHCP-запросы клиентов, а также сетевой интерфейс ViPNet Coordinator HW, который должен использоваться для доступа к удаленному серверу, с помощью следующей команды:

```
hostname# inet dhcp relay external-interface <имя интерфейса> server {<IP-адрес сервера> | <DNS-имя сервера>}
```

3 Чтобы включить или выключить автоматический запуск службы DHCP-relay при загрузке ViPNet Coordinator HW, выполните команду:

```
hostname# inet dhcp relay mode {on | off}
```

По умолчанию автоматический запуск DHCP-relay выключен.

Примечание. Перед тем как вы включаете или запускаете службу DHCP-relay, убедитесь, что работа DHCP-сервера завершена (см. «Настройка параметров DHCP-сервера» на стр. 157).

4 Чтобы запустить службу DHCP-relay или завершить ее работу, выполните команду:

```
hostname# inet dhcp relay {start | stop}
```

Примечание. Чтобы впоследствии завершить работу службы DHCP-relay и одновременно сбросить все ее настройки, а также выключить автоматический запуск службы при загрузке ViPNet Coordinator HW, выполните команду:

```
hostname# inet dhcp relay reset
```

Настройка параметров DNS-сервера

В состав ПО ViPNet Coordinator HW входит DNS-сервер (далее — локальный DNS-сервер), который может использоваться для разрешения (преобразования) символьных имен в IP-адреса в ответ на собственные запросы и на запросы других сетевых узлов (DNS-клиентов).

Локальный DNS-сервер перенаправляет поступающие к нему DNS-запросы на вышестоящие DNS-серверы и передает полученные ответы DNS-клиентам. По умолчанию локальный DNS-сервер настроен таким образом, что он может выполнять разрешение имен с использованием корневых DNS-серверов. Для этого требуется наличие подключения к Интернету. При отсутствии доступа к Интернету для разрешения имен следует использовать другие доступные (не корневые) DNS-серверы. В этом случае необходимо добавить адреса доступных серверов в настройки локального DNS-сервера.

Все DNS-серверы, отличные от корневых, добавляются в качестве DNS-серверов пересылки (forwarder). Если адрес доступного DNS-сервера известен заранее, то его можно задать в процессе первоначальной установки справочников и ключей (подробнее см. в документе «ViPNet Coordinator HW. Подготовка к работе»).

Внимание! В ситуации, когда DNS-серверы пересылки, заданные в настройках локального DNS-сервера, недоступны, но при этом доступны корневые DNS-серверы, DNS-клиенты будут получать ответы на свои запросы с задержкой.

Список DNS-серверов пересылки можно сформировать вручную. Также можно получить адреса DNS-серверов от внешнего DHCP-сервера, если на одном из интерфейсов ViPNet Coordinator HW установлен режим DHCP (см. «Настройка параметров DHCP-сервера» на стр. 157). В полученный от DHCP-сервера список можно добавлять адреса вручную. После перезагрузки ViPNet Coordinator HW или установки на интерфейсе статического IP-адреса список, полученный от внешнего DHCP-сервера, удаляется из настроек локального DNS-сервера. В этом случае для разрешения имен будут использоваться корневые DNS-серверы.

Примечание. Если режим DHCP установлен на нескольких интерфейсах ViPNet Coordinator HW, то результат обработки собственных DNS-запросов и запросов DNS-клиентов не определен, так как неизвестно, какой интерфейс будет включен первым и какой список DNS-серверов пересылки получит ViPNet Coordinator HW.

Чтобы настроить параметры DNS-сервера, выполните следующие действия:

1 По умолчанию DNS-запросы к серверу разрешены для любых узлов. Если необходимо явно указать IP-адреса узлов и подсетей, узлам которых разрешены запросы к DNS-серверу, задайте их с помощью команды:

```
hostname# inet dns clients add {<IP-адрес> | <IP-адрес/длина маски>}
```

Для проверки текущего списка IP-адресов, которым разрешены DNS-запросы к серверу, используйте команду:

```
hostname# inet dns clients list
```

Для удаления IP-адресов из списка используйте команду:

```
hostname# inet dns clients delete {<IP-адрес> | <IP-адрес/длина маски>}
```

2 Добавьте IP-адреса DNS-серверов, на которые ViPNet Coordinator HW будет перенаправлять DNS-запросы, с помощью команды:

```
hostname# inet dns forwarders add <IP-адрес>
```

По умолчанию DNS-запросы перенаправляются на корневые DNS-серверы из домена root-servers.net.

Для проверки текущего списка DNS-серверов пересылки используйте команду:

```
hostname# inet dns forwarders list
```

Для удаления DNS-серверов из списка используйте команду:

```
hostname# inet dns forwarders delete <IP-адрес>
```

3 Чтобы включить или выключить автоматический запуск DNS-сервера при загрузке ViPNet Coordinator HW, выполните команду:

```
hostname# inet dns mode {on | off}
```

По умолчанию автоматический запуск DNS-сервера включен.

4 Чтобы запустить DNS-сервер или завершить его работу, выполните команду:

```
hostname# inet dns {start | stop}
```

5 Чтобы просмотреть текущие параметры DNS-сервера, выполните команду:

```
hostname> inet show dns
```

Настройка параметров NTP-сервера

В состав ПО ViPNet Coordinator HW входит NTP-сервер (далее — локальный NTP-сервер), который может использоваться для синхронизации времени на самом ViPNet Coordinator HW и на других сетевых узлах (NTP-клиентах).

По умолчанию локальный NTP-сервер настроен таким образом, что при наличии подключения к Интернету он может осуществлять синхронизацию времени с использованием публичных NTP-серверов из кластера `pool.ntp.org`. Этот кластер серверов можно дополнить другими NTP-серверами (публичными или корпоративными).

Внимание! Для синхронизации времени рекомендуется использоваться только доверенные NTP-серверы из защищенной сети.

Такая необходимость может возникнуть в случае отсутствия доступа к Интернету или при наличии более близкого и менее нагруженного NTP-сервера (например, корпоративного). Если адрес дополнительного NTP-сервера известен заранее, то его можно задать в процессе первоначальной установки справочников и ключей (подробнее см. в документе «ViPNet Coordinator HW. Подготовка к работе»).

Примечание. Если в качестве дополнительного NTP-сервера используется защищенный узел, видимый по реальному адресу, то для успешной синхронизации времени с таким сервером при загрузке системы рекомендуется в файле `iplir.conf` в секции `[id]` для этого защищенного узла установить параметр `visibility` в значение `real`. Описание секции и параметра см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам», в разделе «Файл iplir.conf».

Список дополнительных NTP-серверов можно сформировать вручную или получить их адреса от внешнего DHCP-сервера. При этом в полученный от DHCP-сервера список можно добавлять адреса вручную. После перезагрузки ViPNet Coordinator HW или установки на интерфейсе статического IP-адреса список, полученный от внешнего DHCP-сервера, удаляется из настроек локального NTP-сервера.

Примечание. Если режим DHCP установлен на нескольких интерфейсах ViPNet Coordinator HW, то результат синхронизации времени на самом ViPNet Coordinator HW и на NTP-клиентах не определен, так как неизвестно, какой интерфейс будет включен первым и какой список дополнительных NTP-серверов получит ViPNet Coordinator HW.

Чтобы настроить параметры NTP-сервера, выполните следующие действия:

1 Чтобы добавить IP-адрес NTP-сервера для синхронизации системного времени ViPNet Coordinator HW, выполните команду:

```
hostname# inet ntp add {IP-адрес | DNS-имя}
```

2 Чтобы удалить адрес NTP-сервера из списка, выполните команду:

```
hostname# inet ntp delete {IP-адрес | DNS-имя}
```

3 Чтобы включить или выключить автоматический запуск NTP-сервера при загрузке ViPNet Coordinator HW, выполните команду:

```
hostname# inet ntp mode {on | off}
```

По умолчанию автоматический запуск NTP-сервера включен.

4 Чтобы запустить NTP-сервер или завершить его работу, выполните следующую команду:

```
hostname# inet ntp {start | stop}
```

Внимание! NTP-серверы, указанные по DNS-имени, будут доступны, только если запущен DNS-сервер (см. «Настройка параметров DNS-сервера» на стр. 161).

Если ViPNet Coordinator HW не удалось подключиться ни к одному из NTP-серверов, локальный NTP-сервер не запускается. Информация о попытках подключения к NTP-серверам записывается в журнал событий.

1 Чтобы просмотреть текущие параметры и состояние NTP-сервера, выполните команду:

```
hostname> inet show ntp
```

Настройка параметров прокси-сервера

Прокси-сервер, входящий в состав программного обеспечения ViPNet Coordinator HW, обладает следующими возможностями:

- Кэширование данных для ускорения доступа пользователей к часто запрашиваемым ресурсам.
- Работа в «прозрачном» режиме, для которого не требуется дополнительная настройка приложений на рабочих местах пользователей.
- Фильтрация содержимого трафика.
- Проверка трафика с помощью антивируса.

Схема работы прокси-сервера при обработке запроса пользователя представлена на следующем рисунке.

Рисунок 16: Схема работы прокси-сервера ViPNet Coordinator HW

При обращении пользователя к какому-либо интернет-ресурсу запрос обрабатывается следующим образом:

1 Запрос от клиента прокси-сервера поступает на ViPNet Coordinator HW:
   - Запрос обрабатывается сетевыми фильтрами:
     - если для запроса сработал запрещающий сетевой фильтр, то передача IP-пакетов блокируется, в журнал регистрации IP-пакетов добавляется запись об этом событии (см. «Типы событий в журнале регистрации IP-пакетов» на стр. 241);

- если запрос не попал ни под один из запрещающих сетевых фильтров, он передается на прокси-сервер.

  - Если на прокси-сервере включена функция фильтрации содержимого трафика, то запрос обрабатывается ее правилами:

    - если ни одно правило не сработало для данного типа содержимого HTTP-трафика, то применяется правило по умолчанию (блокирующее или разрешающее);

    - если для данного типа содержимого трафика сработало запрещающее правило, то передача данных этого типа содержимого блокируется.

  - Если на прокси-сервере включена антивирусная проверка трафика, данные проверяются на наличие вирусов. Если пользователь попытается отправить файл, в котором был обнаружен вирус, он увидит соответствующее сообщение от антивируса и попытка отправки файла будет заблокирована. Запись об этом событии будет создана в системном журнале (см. «События журнала устранения неполадок, связанные с аутентификацией и настройкой оборудования» на стр. 246).

2. Если запрос не был заблокирован при фильтрации содержимого трафика или при проверке антивирусом, он передается на интернет-ресурс.

3. Ответ от интернет-ресурса поступает на ViPNet Coordinator HW:

   - Ответ обрабатывается сетевыми фильтрами аналогично п. 1.

   - Если на прокси-сервере включена функция фильтрации содержимого трафика, то ответ обрабатывается ее правилами аналогично п. 1.

   - Если на прокси-сервере включена антивирусная проверка трафика, данные проверяются на наличие вирусов. Если пользователь попытается загрузить файл (в том числе открыть HTML-страницу), в котором был обнаружен вирус, он увидит соответствующее сообщение от антивируса и попытка загрузки файла будет заблокирована. Запись об этом событии будет создана в системном журнале (см. «События журнала устранения неполадок, связанные с аутентификацией и настройкой оборудования» на стр. 246).

4. В случае успешного прохождения всех проверок трафик от прокси-сервера передается пользователю.

Если вы хотите использовать встроенный прокси-сервер для защиты интернет-соединения пользователей вашей локальной сети, настройте основные параметры прокси-сервера (см. «Настройка основных параметров прокси-сервера» на стр. 167). Вы также можете включить антивирусную проверку (см. «Настройка антивируса» на стр. 168) и фильтрацию содержимого трафика (см. «Настройка фильтрации содержимого трафика» на стр. 170).

Настройка основных параметров предполагает выбор внешнего сетевого интерфейса сервера, задание IP-адресов, на которых прокси-сервер будет принимать запросы от пользователей, и IP-адресов локальных сетей, которым разрешено использовать прокси-сервер. Вы также можете включить «прозрачный» режим работы прокси-сервера. Если прокси-сервер работает в обычном режиме («прозрачный» режим выключен), в пользовательских приложениях, например в веб-браузере, требуется указать IP-адрес и порт прокси-сервера.

Если прокси-сервер работает в «прозрачном» режиме, дополнительная настройка приложений не требуется. При этом пользователи не имеют возможности отказаться от использования прокси-

сервера. На компьютерах пользователей задайте IP-адрес прокси-сервера (узла с ПО ViPNet Coordinator HW) в качестве шлюза по умолчанию.

Примечание. По умолчанию в «прозрачном» режиме работы прокси-сервера к DNS-серверу ViPNet Coordinator HW могут обращаться только защищенные узлы. Если необходимо предоставить доступ к DNS-серверу для каких-либо открытых узлов, на ViPNet Coordinator HW создайте сетевой фильтр с помощью следующей команды:

```
hostname# firewall local add rule "AllowDNS" src <IP-адреса открытых узлов> dst @local udp dport 53 pass
```

Настройка основных параметров прокси-сервера

Для настройки основных параметров прокси-сервера выполните следующие действия:

1 Завершите работу прокси-сервера, если он запущен, с помощью команды:

```
hostname> service http-proxy stop
```

Внимание! На запущенном прокси-сервере нельзя выполнить настройку параметров.

2 Укажите IP-адреса и порты, на которых прокси-сервер будет принимать запросы от пользователей:

- Чтобы добавить IP-адрес и порт, выполните команду:

  ```
  hostname# service http-proxy listen-address add <интерфейс> <порт>
  ```

- Чтобы просмотреть список заданных IP-адресов и портов, выполните команду:

  ```
  hostname> service http-proxy listen-address list
  ```

- Чтобы удалить IP-адрес и порт, выполните команду:

  ```
  hostname# service http-proxy listen-address delete <интерфейс> <порт>
  ```

Внимание! Для приема запросов рекомендуется использовать сетевые интерфейсы со статическими IP-адресами. При изменении IP-адресов сетевых интерфейсов необходимо завершить работу службы прокси-сервера, в качестве адресов для подключения указать текущие IP-адреса, затем снова запустить прокси-сервер.

3 Укажите внешний IP-адрес ViPNet Coordinator HW, который используется для подключения к Интернету. Для этого выполните команду:

```
hostname# service http-proxy external-address set <интерфейс>
```

Чтобы просмотреть заданный внешний IP-адрес, выполните команду:

```
hostname> service http-proxy external-address show
```

4 Если требуется, задайте размер кэша прокси-сервера с помощью команды:

```
hostname# service http-proxy cache <размер>
```

Размер кэша указывается в мегабайтах, значение по умолчанию 256 Мбайт. Кэш используется для хранения копии данных, к которым часто обращаются пользователи.

5 Чтобы включить или выключить прозрачный режим работы прокси-сервера, выполните команду:

```
hostname# service http-proxy transparent-mode {on | off}
```

6 По завершении настроек перед запуском прокси-сервера добавьте необходимые для его работы сетевые фильтры и правила трансляции с помощью команды:

```
hostname# service http-proxy fw-rules apply
```

При выполнении команды все необходимые сетевые фильтры и правила трансляции будут созданы автоматически.

Внимание! Если для подключения к Интернету ViPNet Coordinator HW использует маршрут с несколькими шлюзами (multi-hop route), для корректной работы прокси-сервера на ViPNet Coordinator HW создайте сетевой фильтр с помощью следующей команды:

```
hostname# firewall local add src @local dst @InternetIP tcp dport 80
tcp dport 21 tcp dport 443 pass
```

---

7 Чтобы включить или выключить автоматический запуск прокси-сервера при загрузке ViPNet Coordinator HW, выполните команду:

```
hostname# service http-proxy mode {on | off}
```

8 Чтобы запустить службу прокси-сервера, выполните команду:

```
hostname# service http-proxy start
```

Внимание! Для корректной работы прокси-сервера перед его запуском необходимо настроить и включить DNS-сервер (см. «Настройка параметров DNS-сервера» на стр. 161).

Чтобы завершить работу службы прокси-сервера, выполните команду:

```
hostname# service http-proxy stop
```

Настройка антивируса

Если ViPNet Coordinator HW используется в качестве прокси-сервера, вы можете настроить антивирусную проверку данных, передаваемых через прокси-сервер по протоколу HTTP в обоих направлениях: из Интернета к пользователю и от пользователя в Интернет (например, загрузка файла на файлообменный ресурс).

Внимание! Если при загрузке зараженного файла на файлообменные ресурсы используется метод множественной загрузки (multipart upload), то антивирусная проверка не сможет обнаружить вирус в этом файле.

Для антивирусной проверки содержимого трафика в прокси-сервер встроено антивирусное решение, разрабатываемое компанией «Лаборатория Касперского».

Для настройки параметров антивируса выполните следующие действия:

1 Остановите прокси-сервер, если он запущен, с помощью команды:

```
hostname# service http-proxy stop
```

Внимание! Настройку антивируса необходимо выполнять только при остановленном прокси-сервере.

2 Если вы еще не установили лицензионный ключ для Kaspersky Anti-Virus или срок действия ключа истек, установите ключ с помощью команды:

```
hostname# service http-proxy antivirus kav key install
```

Затем вставьте USB-накопитель с файлом лицензионного ключа антивируса и нажмите **Enter**.

3 Чтобы немедленно загрузить обновления сигнатур для выбранной антивирусной программы, выполните команду:

```
hostname# service http-proxy antivirus kav fetch
```

Внимание! Обновление антивирусных баз возможно только при наличии действующего лицензионного ключа.

4 Чтобы настроить автоматическое обновление антивирусных баз, выполните команду:

```
hostname# service http-proxy antivirus kav schedule-fetch <частота обновления>
```

Чтобы задать частоту обновления базы, укажите одно из следующих значений:

- `none` (по умолчанию) — чтобы отключить автоматическое обновление;
- число от 1 до 5 — чтобы обновлять базу от одного до пяти раз в течение суток. При этом обновление будет выполняться через равные промежутки времени.

Если вы настроите обновление один раз в сутки, база будет ежедневно обновляться в 00:00. Если вы настроите обновление три раза в сутки, база будет ежедневно обновляться в 0:00, 8:00, 16:00.

Например: `hostname# service http-proxy antivirus kav schedule-fetch 3`

5 Для просмотра расписания автоматического обновления выполните команду:

```
hostname> service http-proxy antivirus show-status
```

6 Чтобы включить антивирусную проверку содержимого, выполните команду:

```
hostname# service http-proxy antivirus kav mode on
```

Чтобы отключить активную антивирусную программу, выполните команду:

```
hostname# service http-proxy antivirus kav mode off
```

7 Чтобы просмотреть информацию об установленной лицензии Kaspersky Anti-Virus, выполните команду:

```
hostname# service http-proxy antivirus kav key show
```

Чтобы установить лицензию, выполните команду:

```
hostname# service http-proxy antivirus kav install
```

Чтобы удалить лицензию, выполните команду:

```
hostname# service http-proxy antivirus kav delete
```

8 После завершения настройки антивируса запустите прокси-сервер с помощью команды:

```
hostname# service http-proxy start
```

Настройка фильтрации содержимого трафика

Прокси-сервер ViPNet Coordinator HW имеет функцию проверки HTTP-трафика по его содержимому. С помощью правил фильтрации содержимого трафика вы можете настроить блокировку трафика по MIME-типу файла или приложения, а также по типу HTTP-метода запроса к удаленному ресурсу.

Фильтрация содержимого трафика включается автоматически при запуске прокси-сервера (см. «Настройка основных параметров прокси-сервера» на стр. 167). Для настройки параметров этой функции выполните следующие действия:

1 Если прокси-сервер запущен, завершите его работу с помощью команды:

```
hostname# service http-proxy stop
```

Внимание! Если прокси-сервер запущен, настроить его параметры невозможно.

2 Чтобы включить или выключить фильтрацию содержимого трафика, выполните команду:

```
hostname# service http-proxy content-filter mode {on | off}
```

3 Чтобы задать действие правила по умолчанию, выполните команду:

```
hostname# service http-proxy content-filter default-action {pass | drop},
```

указав следующие параметры:

- `pass` — правило по умолчанию пропускает HTTP-трафик, не подпадающий под действие других правил;
- `drop` — правило по умолчанию блокирует HTTP-трафик, не подпадающий под действие других правил.

Эта команда создает правила по умолчанию для запросов и ответов (см. ниже) с заданным действием (пропускать или блокировать).

4 Чтобы добавить правило фильтрации содержимого трафика, выполните команду:

```
hostname# service http-proxy content-filter add [num <номер>] [rule <имя>] src <адрес отправителя> dst <адрес получателя> [command <HTTP-метод>] [mime-type <тип содержимого>] <действие>,
```

указав следующие параметры:

- `<номер>` — порядковый номер правила в таблице, определяющий его приоритет. Чем меньше порядковый номер правила, тем выше его приоритет (наиболее приоритетное правило имеет номер 1). Если указанный номер правила меньше последнего номера в таблице, нумерация правил, следующих после нового правила, будет автоматически изменена (их номера будут увеличены на 1). Например: последний номер правила в таблице — 5, вы добавили правило с номером 3. Правило добавится с указанным номером, при этом правило с номером 3, которое было создано ранее добавленного вами, получит номер 4. Правила с номерами 4 и 5, соответственно, получат номер 5 и 6.;
- `<имя>` — имя правила. Если имя состоит из нескольких слов, разделенных пробелом, заключите его в кавычки (например, `rule "number one"`).

Примечание. Порядковый номер и имя не являются обязательными параметрами правила фильтрации содержимого трафика. При создании правила без указания номера ему будет присвоен номер, следующий за номером последнего правила в соответствующей таблице, и оно будет иметь самый низкий приоритет. При создании правила без указания имени будет создано правило без имени.

- Параметры, определяющие процесс обработки трафика этим правилом. Подробное описание этих параметров см. в соответствующих разделах:
  - Адрес отправителя (на стр. 172).
  - Адрес получателя (на стр. 172).
  - HTTP-метод (на стр. 172).
  - Тип содержимого (на стр. 173).
  - Действие (на стр. 130).

Примечание. Если при вводе команды была допущена синтаксическая ошибка, то в результате выполнения такой команды появится сообщение с указанием слова, содержащего ошибку. Исправьте ошибку и выполните команду еще раз.

Адрес отправителя

Адрес отправителя является обязательным параметром правила фильтрации содержимого трафика и описывается лексемой:

```
src <адрес отправителя>
```

Возможные значения адреса отправителя:

- IP-адрес, например:

  ```
  src 192.168.1.36
  ```

- адрес подсети в формате классовой адресации, например:

  ```
  src 192.168.1.0/24
  ```

- системная группа объектов `any`, например:

  ```
  src @any
  ```

Примечание. В качестве отправителя всегда необходимо указывать клиента прокси-сервера. Использование доменных имен не допускается.

Адрес получателя

Адрес получателя является обязательным параметром правила фильтрации содержимого трафика и описывается лексемой:

```
dst <адрес получателя>
```

Возможные значения адреса получателя:

- IP-адрес или доменное имя узла;
- адрес подсети в формате классовой адресации, например:

  ```
  dst 192.168.1.0/24
  ```

- системная группа объектов `any`, например:

  ```
  dst @any
  ```

Примечание. В качестве получателя всегда необходимо указывать удаленный HTTP-сервер.

HTTP-метод

HTTP-метод не является обязательным параметром правила фильтрации содержимого трафика.

HTTP-метод описывается лексемой:

```
command <HTTP-метод>
```

Вы можете указать любой из методов протокола HTTP/1.0 (но только один):

- GET
- HEAD
- POST
- PUT
- DELETE
- PATCH
- CONNECT
- OPTIONS
- TRACE

Внимание! Вы можете указать HTTP-метод только в том случае, если в правиле фильтрации содержимого трафика не был задан тип содержимого.

Правило с заданным HTTP-методом будет действовать в тех случаях, когда прокси-сервер ViPNet Coordinator HW получит от клиента запрос к удаленному HTTP-серверу с этим методом. Например, необходимо запретить заполнение и отправку форм на Facebook. Для этого необходимо создать правило, которое будет блокировать метод POST для facebook.com:

```
hostname# service http-proxy content-filter add src @any dst facebook.com command POST
drop
```

Тип содержимого

Тип содержимого не является обязательным параметром правила фильтрации содержимого трафика.

Тип содержимого описывается лексемой:

```
mime-type <тип содержимого>
```

Вы можете указать любой тип содержимого из поддерживаемых (см. «Поддерживаемые типы содержимого» на стр. 257), но только один.

Внимание! Вы можете указать тип содержимого только в том случае, если в правиле фильтрации содержимого трафика не был задан HTTP-метод.

Правило с заданным типом содержимого будет блокировать трафик, передающий этот тип содержимого. Например, следующее правило будет блокировать загрузку изображений формата PNG:

```
hostname# service http-proxy content-filter add src @any dst @any mime-type image/png
drop
```

Примечание. MIME-тип файла может зависеть от настроек удаленного HTTP-сервера, с которого этот файл был загружен. То есть, файл будет иметь тот MIME-тип, который был назначен ему HTTP-сервером.

Настройка параметров точки доступа к сети Wi-Fi

Внимание! Использование ViPNet Coordinator HW в качестве точки доступа Wi-Fi возможно только в исполнениях со встроенными адаптерами Wi-Fi: ViPNet Coordinator HW50 A, B на аппаратной платформе HW50 N2 и ViPNet Coordinator HW100 A, B на аппаратной платформе HW100 N2.

Если включена точка доступа Wi-Fi, сетевому интерфейсу `wlan0` автоматически присваивается IP-адрес 192.168.20.1 и на этом интерфейсе запускается DHCP-сервер. DHCP-сервер имеет собственные параметры, которые не могут быть просмотрены или изменены пользователем:

- Диапазон распределяемых IP-адресов: 192.168.20.2–192.168.20.20.
- Адреса DNS- и NTP-сервера: 192.168.20.1 (адрес сетевого интерфейса `wlan0`).

Примечание. Если требуется обеспечить возможность соединений между устройствами, которые подключены к сети Wi-Fi, и устройствами, подключенными к сети Ethernet, на узле ViPNet Coordinator HW создайте транзитные фильтры открытой сети, разрешающие пропускание IP-пакетов между этими сетями (см. «Создание сетевого фильтра» на стр. 124).

Чтобы настроить ViPNet Coordinator HW для выполнения функций точки доступа Wi-Fi, выполните следующие действия:

1. Переключите сетевой интерфейс `wlan0` в режим точки доступа с помощью команды:

   ```
   hostname# inet wifi role access-point
   ```

2. Задайте имя вашей сети Wi-Fi и способ аутентификации пользователей в сети. Для этого используйте одну из следующих команд:

   - Если аутентификация в сети не требуется, выполните команду:

     ```
     hostname# inet wifi access-point authentication open
     ```

     Затем по запросу введите имя сети.

   - Если для аутентификации требуется использовать пароль, выберите режим аутентификации WPA-PSK или WPA2-PSK. Для этого выполните команду:

     ```
     hostname# inet wifi access-point authentication {wpa-psk | wpa2-psk}
     ```

     Затем по запросу введите имя сети и пароль. Заданный пароль нужно будет сообщить пользователям, подключающимся к вашей сети Wi-Fi.

3. Если требуется задать стандарт беспроводной связи для вашей сети Wi-Fi, выполните команду:

   ```
   hostname# inet wifi access-point hwmode {b | g}
   ```

   Поддерживаются следующие стандарты:

- b — IEEE 802.11b — 2,4 ГГц, скорость соединения до 11 Мбит/с.
- g — IEEE 802.11g — 2,4 ГГц, скорость соединения до 54 Мбит/с (используется по умолчанию).

4 Если требуется задать номер используемого канала Wi-Fi, выполните команду:

```
hostname# inet wifi access-point channel <номер канала>
```

По умолчанию используется канал номер 1, допустимы номера от 1 до 11.

5 Чтобы включить сетевой интерфейс `wlan0`, выполните команду:

```
hostname# inet wifi mode on
```

В результате ViPNet Coordinator HW будет работать в режиме точки доступа Wi-Fi.

Примечание. ViPNet Coordinator HW может также работать в качестве клиента сети Wi-Fi (см. «Подключение к сети Wi-Fi» на стр. 58). Использование ViPNet Coordinator HW одновременно в качестве клиента и точки доступа Wi-Fi не поддерживается.

12

Настройка маршрутизации

Общие сведения о маршрутизации

ViPNet Coordinator HW поддерживает функции маршрутизации IP-трафика в сетях со сложной структурой.

Под маршрутизацией понимается процесс выбора маршрута следования IP-пакета (см. глоссарий, стр. 269), передаваемого в сети от одного узла другому. Маршрут следования выбирается из подмножества маршрутов, заданных на маршрутизаторе и хранящихся в таблице маршрутизации.

В таблицу маршрутизации маршруты могут попадать в явном виде (статические маршруты) либо с помощью алгоритмов маршрутизации на основе информации о топологии и состоянии сети, предоставляемой протоколами маршрутизации (динамические маршруты). В первом случае не требуется никаких дополнительных условий. Но стоит заметить, что в сетях со сложной структурой процесс настройки статических маршрутов может стать весьма трудоемким из-за большого числа маршрутов, которые требуется создать. Во втором случае на всех маршрутизаторах в сети должно быть настроено использование определенного протокола динамической маршрутизации, по которому маршрутизаторы будут обмениваться друг с другом информацией о доступных им сетях, автоматически строить доступные маршруты в каждую сеть и выбирать из них наилучшие.

Статическую маршрутизацию удобно использовать в небольших сетях либо в крупных сетях в частных случаях. В сетях с разветвленной и неоднородной топологией рекомендуется использовать динамическую маршрутизацию. Кроме автоматического формирования таблиц маршрутизации, динамическая маршрутизация позволяет:

- Автоматически выбирать по определенным критериям наилучший маршрут из нескольких доступных.
- Организовать защиту от сбоев. В случае сбоя какого-либо маршрутизатора автоматически выбирается другой наилучший маршрут и загружается в таблицу маршрутизации.
- Организовать динамическую балансировку нагрузки передаваемого IP-трафика.

ViPNet Coordinator HW может выполнять маршрутизацию IP-трафика с использованием следующих видов маршрутов:

- статических маршрутов, в том числе на основе маршрутов с несколькими шлюзами;
- динамических маршрутов, формируемых по протоколу DHCP (см. глоссарий, стр. 264) и PPP (см. глоссарий, стр. 265);
- динамических маршрутов, формируемых протоколом OSPF (см. глоссарий, стр. 265).

Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW

В ViPNet Coordinator HW для хранения маршрутов используются две таблицы маршрутизации:

- Routing Information Base (далее — RIB) — полная таблица маршрутизации, которая содержит все статические маршруты, созданные администратором, и все динамические маршруты, полученные по протоколам DHCP, PPP и OSPF.
- Forwarding Information Base (далее — FIB) — таблица маршрутизации, которая содержит только наилучшие статические и динамические маршруты в каждую сеть, выбранные из RIB; загружается в ядро системы и используется при маршрутизации IP-трафика.

Все новые маршруты, независимо от способа их создания, попадают в RIB. Каждый маршрут имеет следующие характеристики:

- Источник получения маршрута — определяет, каким образом был сформирован маршрут: с использованием статического правила или по протоколу динамической маршрутизации.
- Административная дистанция — определяет приоритет маршрута от каждого источника. Используется тогда, когда в таблице маршрутизации задано несколько маршрутов в одну и ту же сеть. Чем меньше административная дистанция, тем более приоритетным считается маршрут.

  Административная дистанция задается для каждого статического маршрута. В случае динамических маршрутов административная дистанция задается не для конкретного маршрута, а для всего протокола сразу. Поэтому все динамические маршруты, добавленные в RIB этим протоколом, имеют одинаковую дистанцию.

  Административная дистанция для протокола PPP отдельно не задается и равна дистанции для протокола DHCP. Административная дистанция для протоколов DHCP/PPP и OSPF не может быть одинаковой, чтобы маршруты этих протоколов не перемешивались. Также административная дистанция статических правил маршрутизации не может совпадать с дистанцией, заданной для какого-либо протокола динамической маршрутизации.

Примечание. В ViPNet Coordinator HW для каждого типа маршрутов задается по умолчанию следующая административная дистанция:

- для статических маршрутов — 10;
- для маршрутов DHCP/PPP-сервера — 70;
- для маршрутов протокола OSPF — 110.

Для первых двух типов маршрутов вы можете менять значение административной дистанции, для маршрутов последнего типа менять это значение нельзя.

Для маршрутов DHCP/PPP-сервера задается общая административная дистанция.

- Метрика — определяет приоритет внутри списка динамических маршрутов, сформированных и добавленных в RIB определенным протоколом динамической маршрутизации. Чем меньше метрика, тем выше приоритет маршрута.

  Для DHCP-протокола метрики указываются вручную, причем на каждом сетевом интерфейсе, на котором включен режим DHCP и настроен параметр получения маршрутов от DHCP-сервера. Таким образом, если на разные сетевые интерфейсы будут получены одинаковые DHCP-маршруты в одну и ту же сеть, то будет выбран маршрут с наименьшей метрикой. Для PPP-протокола также может быть задана метрика вручную в том случае, если ViPNet Coordinator HW подключен к нескольким сетям, одной из которых является сеть 3G или 4G, а в другой развернут DHCP-сервер. Если метрики для DHCP/PPP-протоколов не были заданы, то используется метрика по умолчанию.

- Для динамических маршрутов, формируемых протоколом OSPF, метрики формируются самим протоколом, поэтому их задавать не требуется.

- Вес — определяет долю IP-трафика, который будет проходить по маршруту через указанный шлюз. Используется только в статических маршрутах и задается для шлюза. Позволяет настроить балансировку передаваемого IP-трафика между несколькими шлюзами, когда часть IP-пакетов направляется на один шлюз, а часть — на другой (см. «Настройка балансировки IP-трафика» на стр. 188). Поэтому вес задается тогда, когда создается несколько статических маршрутов с одинаковым IP-адресом назначения и разными шлюзами.

Таблица FIB формируется на основе содержания таблицы RIB и включает в себя только наилучшие маршруты в каждую сеть. Если в результате заполнения таблицы RIB в ней оказывается несколько маршрутов в одну сеть, то наилучший маршрут определяется по следующим правилам:

- Если в RIB присутствуют статические маршруты с разной дистанцией, то в FIB загружается маршрут с наименьшей дистанцией. Например, имеется два маршрута в сеть 10.0.5.0 с маской 255.255.255.0. Для первого маршрута задана административная дистанция — 10, для второго — 30. В результате в FIB попадет первый маршрут.

- Если в RIB присутствуют статические маршруты с одинаковой дистанцией, то в FIB загружаются все маршруты.

- Если в RIB присутствуют динамические маршруты, то в FIB загружается маршрут с наибольшим приоритетом (то есть наименьшей метрикой, заданной протоколом динамической маршрутизации). Если приоритеты равны, то в FIB попадают все маршруты.

Маршруты от одного источника (Static, DHCP, OSPF) в одну и ту же сеть с одинаковыми метриками (или административными дистанциями в случае статических маршрутов) при маршрутизации суммируются — объединяются в один маршрут с несколькими шлюзами (multi-hop route). При просмотре (см. «Просмотр общей таблицы маршрутизации» на стр. 182) такие маршруты отображаются в виде одного маршрута с несколькими шлюзами:

```
10.100.2.0/24 [30/23] (weight 1) via 10.1.30.202, eth1
                      (weight 1) via 10.1.31.201, eth2
```

Внимание! Если хотя бы один из шлюзов объединенного маршрута multi-hop route выйдет из строя, работоспособность маршрута не может быть гарантирована.

Если маршрут был удален из RIB, то он также удаляется из FIB. При этом по вышеописанным правилам выбирается новый наилучший маршрут в ту же сеть из имеющихся в RIB и немедленно загружается в FIB.

Просмотр общей таблицы маршрутизации

В процессе или после настройки маршрутизации вы можете просмотреть список всех существующих маршрутов (содержимое RIB (см. «Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW» на стр. 179)). Для этого выполните команду:

```
hostname> inet show routing
```

В результате выполнения команды будет выдана информация в следующем виде:

Рисунок 17. Просмотр таблицы маршрутизации

Цифрами на рисунке обозначены:

1. Статический маршрут по умолчанию. Выбран как наилучший. Объединяет три маршрута, поэтому включает в себя три шлюза. Каждому шлюзу назначен вес, поэтому в процессе маршрутизации будет осуществляться балансировка нагрузки.
2. Маршрут в подсеть 10.0.3.0 с маской 255.255.255.0, к которой подключен один из интерфейсов ViPNet Coordinator HW. Выбран как наилучший.
3. Маршруты в подсети 10.100.1.0 и 10.100.2.0 с маской 255.255.255.0, полученные по протоколу OSPF (см. глоссарий, стр. 265). Выбраны как наилучшие.
4. Статический маршрут в подсеть 192.168.0.0 с маской 255.255.0.0. Выбран как наилучший. Объединяет два маршрута, поэтому включает в себя два шлюза. Каждому шлюзу назначен вес, поэтому в процессе маршрутизации будет осуществляться балансировка нагрузки.

Каждый раз перед списком маршрутов выводится список возможных атрибутов, которые могут присваиваться для маршрутов, и их расшифровка. В таблице ниже приведены пояснения по данным атрибутам.

Таблица 11. Атрибуты маршрутов

Атрибут	Расшифровка в командном интерпретаторе	Пояснение
C	connected	Маршрут в подсеть, к которой подключен один из интерфейсов ViPNet Coordinator HW.
S	static	Статический маршрут.
O	OSPF	Маршрут протокола динамической маршрутизации OSPF.
B	BGP	Маршрут протокола динамической маршрутизации BGP. В текущей версии ViPNet Coordinator HW не поддерживается работа по данному протоколу.
D	DHCP/PPP	Маршрут, предоставленный DHCP/PPP-сервером.
>	selected route	Наилучший маршрут. Отображается в том случае, если в таблице есть несколько маршрутов в одну и ту же сеть с разной административной дистанцией (см. глоссарий, стр. 266).
*	FIB route	Маршрут, загруженный в таблицу FIB и используемый для маршрутизации.

Примечание. В таблице маршрутизации могут присутствовать другие атрибуты, кроме описанных в таблице выше. Эти атрибуты зарезервированы для других протоколов маршрутизации и в текущей версии ViPNet Coordinator HW не используются.

Общие сведения для работы по протоколу OSPF

Протокол OSPF является внутренним шлюзовым протоколом (Interior Gateway Protocol, IGP) и используется для распространения данных маршрутизации внутри одной автономной системы (см. глоссарий, стр. 266).

Работа по данному протоколу устроена следующим образом. OSPF-маршрутизаторы сначала посредством широковещательной рассылки (multicast) устанавливают связь друг с другом. Затем посредством однонаправленной рассылки (unicast) начинают обмениваться друг с другом сообщениями типа Link State Advertise (LSA), в которых передают информацию о состоянии каналов связи. На маршрутизаторе при получении LSA-сообщения информация из него заносится в базу данных (link state database). Если LSA-сообщение содержит новую информацию о канале связи по сравнению с той, которая содержится в базе данных, то оно передается соседним маршрутизаторам (см. глоссарий, стр. 269). После заполнения базы данных на каждом маршрутизаторе по алгоритму Дейкстры вычисляется множество кратчайших маршрутов ко всем сетям назначения и их стоимость. Таким образом, каждый OSPF-маршрутизатор имеет собственное представление о состоянии каналов связи и топологии сети, но при этом все маршрутизаторы используют одну базу данных состояний каналов связи для вычисления кратчайших маршрутов.

Межсетевая среда, в которой находятся OSPF-маршрутизаторы, представляется как совокупность областей (см. глоссарий, стр. 270), соединенных друг с другом через некоторую базовую область. В зависимости от этого выделяют несколько типов областей и несколько типов маршрутизаторов.

Выделяют следующие типы областей маршрутизации:

- Область 0 или базовая область (backbone area) — область, с которой должны быть соединены все остальные области автономной системы.
- Стандартная область — область, способная граничить как с другими областями, так и с другими автономными системами.
- Стандартная тупиковая область (stub area) — область, граничащая только с другими областями.
- Полностью тупиковая область (totally stubby area) — область, граничащая только с одной областью.
- Не полностью тупиковая область (not-so-stubby area) — стандартная тупиковая область, имеющая возможность взаимодействовать с другими автономными системами с помощью пограничных маршрутизаторов (ABR).

Каждый OSPF-маршрутизатор входит в определенную область маршрутизации и обменивается информацией только с маршрутизаторами, входящими в эту же область. При этом обмен производится не напрямую, а через посредника — маршрутизатор, выбранный главным в этой области. Информация между разными областями передается через маршрутизатор, который располагается на границе этих областей. Если автономная система взаимодействует с другой

автономной системой, то информация передается через маршрутизатор, который располагается на границе систем. Такой подход позволяет уменьшить объем рассылаемых LSA-сообщений, тем самым уменьшить нагрузку на маршрутизаторы и повысить скорость построения таблиц маршрутизации.

Таким образом, выделяют следующие типы маршрутизаторов:

- Внутренний маршрутизатор (internal router, IR) — маршрутизатор, все сетевые интерфейсы которого находятся в одной области.
- Назначенный маршрутизатор (designated router, DR) — маршрутизатор, выбранный главным среди всех внутренних маршрутизаторов в одной области.
- Резервный назначенный маршрутизатор (backup designated router, BDR) — маршрутизатор, берущий на себя функции главного в случае, если он вышел из строя.
- Межобластной граничный маршрутизатор (area border router, ABR) — маршрутизатор, соединяющий несколько областей OSPF одной автономной системы.
- Граничный маршрутизатор автономной системы (autonomous system boundary router, ASBR) — маршрутизатор, граничащий с другой автономной системой, работающей по другому протоколу динамической маршрутизации (IGRP, EIGRP, IS-IS, RIP, BGP, Static).

Настройка статической маршрутизации

Если ViPNet Coordinator HW функционирует в сети, конфигурация которой никогда не меняется или в которой используется минимальное количество путей для доставки IP-пакетов, то вы можете настроить маршрутизацию на ViPNet Coordinator HW вручную с помощью статических маршрутов. Кроме этого, настройка маршрутизации вручную также может потребоваться, когда нет возможности использовать в сети маршрутизаторы, работающие по протоколам динамической маршрутизации, или в следующих случаях:

- Чтобы направлять часть IP-трафика всегда по конкретным маршрутам (например, есть требование направлять IP-трафик по самому надежному каналу), или направлять IP-трафик пользователей в Интернет через конкретный шлюз.
- Если нужен статический маршрут по умолчанию на случай, когда работа по протоколам динамической маршрутизации будет невозможна.
- Чтобы распределять передачу IP-трафика по нескольким маршрутам.

Создание статических маршрутов

Вы можете создать маршрут по умолчанию или ряд статических маршрутов следующим образом:

- Чтобы добавить маршрут по умолчанию (см. глоссарий, стр. 269), выполните команду:

  ```
  hostname# inet route add default next-hop <IP-адрес шлюза> [distance <1-255> [weight <1-255>]]]
  ```

- Чтобы добавить статический маршрут, выполните команду:

  ```
  hostname# inet route add <IP-адрес назначения> next-hop <IP-адрес шлюза> [netmask <маска сети> [distance <1-255> [weight <1-255>]]]
  ```

Совет. Если вам требуется добавить один статический маршрут с несколькими шлюзами, создайте несколько статических маршрутов в одну и ту же сеть с одинаковой административной дистанцией, указав в каждом по одному шлюзу. В результате эти маршруты будут просуммированы и объединены в один с несколькими шлюзами. Как правило, такие маршруты требуются для балансировки нагрузки, поэтому в маршрутах также должен быть задан вес шлюзам (см. ниже).

В обеих командах вы можете задать необязательные параметры `distance` — административную дистанцию маршрута (см. глоссарий, стр. 266) и `weight` — вес шлюза (см. глоссарий, стр. 266).

Административная дистанция задается, чтобы назначить приоритет маршруту. Если вы не укажете значение административной дистанции, то оно будет задано автоматически: `distance=10`. Если вы укажете административную дистанцию, и ее значение будет совпадать со значением

административной дистанции, заданной протоколам динамической маршрутизации (см. «Настройка административной дистанции для маршрутов DHCP-протокола» на стр. 191), появится сообщение, что маршрут не может быть добавлен. В этом случае создайте маршрут с другим значением административной дистанции.

Вес задается в маршруте для шлюза. Его требуется задавать в том случае, если создается несколько маршрутов в одну сеть с разными шлюзами, между которыми требуется производить балансировку нагрузки. Подробнее см. раздел Настройка балансировки IP-трафика (на стр. 188).

В результате маршрут будет добавлен в RIB (см. «Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW» на стр. 179). Если маршрут будет определен как наилучший, то он также будет загружен в FIB, после чего начнет использоваться при маршрутизации IP-трафика.

При необходимости вы можете удалить созданные маршруты. Подробнее см. раздел Удаление статического маршрута (на стр. 187).

Удаление статического маршрута

При необходимости вы можете удалить маршрут по умолчанию или статический маршрут следующим образом:

- Чтобы удалить маршрут по умолчанию (см. глоссарий, стр. 269), выполните команду:

  ```
  hostname# inet route delete default [next-hop <IP-адрес шлюза>]
  ```

- Чтобы удалить статический маршрут, выполните команду:

  ```
  hostname# inet route delete <IP-адрес назначения> [netmask <маска сети> next-hop <IP-адрес шлюза>]
  ```

Если при вводе команды вы не указали маску сети или IP-адрес шлюза, и в процессе проверки будет установлено, что в таблице RIB (см. «Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW» на стр. 179) есть несколько маршрутов в указанную сеть, то будет выдан список всех маршрутов. Если вы согласны на удаление всех этих маршрутов, введите символ **y** и нажмите клавишу **Enter**. Аналогичная ситуация возникнет при удалении маршрута с несколькими шлюзами.

В результате маршрут или все маршруты с указанными параметрами будут удалены из RIB. Если маршруты присутствуют в таблице FIB, то из нее они также будут удалены.

Примечание. Если требуется удалить все статические маршруты, включая все маршруты по умолчанию, воспользуйтесь командой:

```
hostname# inet route clear
```

Настройка балансировки IP-трафика

Если вы хотите ускорить процесс отправки IP-пакетов в сеть назначения, то вы можете создать несколько статических маршрутов в данную сеть через разные шлюзы и настроить балансировку IP-трафика между этими маршрутами. Это позволит в процессе разных TCP-соединений отправлять часть IP-пакетов по одному маршруту, часть — по другому, что повысит скорость передачи IP-пакетов в сеть назначения. Балансировка IP-трафика настраивается с помощью одного из параметров маршрутов — веса (см. глоссарий, стр. 266) (`weight`). При этом все маршруты, которые будут участвовать в балансировке IP-трафика, должны иметь одинаковую административную дистанцию, то есть иметь одинаковый приоритет.

Примечание. При маршрутизации маршруты в одну сеть назначения и с одинаковой дистанцией объединяются в один маршрут с несколькими шлюзами (multi-hop route). При просмотре общей таблицы маршрутизации они отображаются как один маршрут с несколькими шлюзами (см. «Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW» на стр. 179).

Для настройки балансировки IP-трафика выполните следующие действия:

1. Создайте нужное количество маршрутов (см. «Создание статических маршрутов» на стр. 186) с одинаковым адресом назначения и с разными шлюзами.
2. Задайте каждому маршруту одинаковую административную дистанцию.
3. Задайте в каждом маршруте вес шлюзу. IP-трафик будет распределяться между шлюзами в соответствии с соотношением их весов. Например:
   - если вы создаете 2 маршрута и в каждом маршруте вес шлюза равен 1, то IP-трафик будет разделен между этими маршрутами поровну, то есть 50% IP-трафика будет передаваться по одному маршруту, 50% — по второму;
   - если вы создаете 3 маршрута, и в первом маршруте вес шлюза равен 1, во втором — вес шлюза равен 2, в третьем — вес шлюза равен 3, то по второму маршруту будет передаваться в два раза больше IP-трафика, чем по первому, по третьему — в три раза больше, чем по первому.

Пример команд для создания статических маршрутов с весом шлюзов равным 1:

```
hostname# inet route add 10.0.5.0 next-hop 10.0.1.1 netmask 255.255.255.0 distance 20
weight 1

hostname# inet route add 10.0.5.0 next-hop 10.0.4.3 netmask 255.255.255.0 distance 20
weight 1
```

Примечание. Если в процессе создания маршрута вы не укажете вес шлюза, то автоматически шлюзу будет назначен вес равный 1. Вы не можете задать вес равный 0.

Балансировка IP-трафика будет осуществляться, только если созданные маршруты признаны наилучшими и присутствуют в таблице FIB (см. «Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW» на стр. 179).

Просмотр статических маршрутов

Если вы хотите просмотреть список всех статических маршрутов, которые были созданы и присутствуют в RIB (см. «Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW» на стр. 179), выполните команду:

```
hostname> inet show routing static
```

В результате выполнения команды появится список маршрутов в следующем виде:

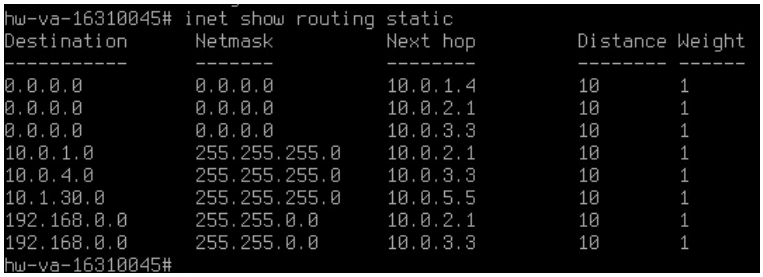

Рисунок 18. Просмотр статических маршрутов

В списке для каждого маршрута указаны: IP-адрес назначения, маска сети, шлюз, административная дистанция маршрута (см. глоссарий, стр. 266) и вес шлюза (см. глоссарий, стр. 266).

Настройка динамической маршрутизации

Если в вашей сети поддерживается работа по протоколу DHCP (см. глоссарий, стр. 264), PPP (см. глоссарий, стр. 265) или OSPF (см. глоссарий, стр. 265), то вы можете настроить на ViPNet Coordinator HW динамическую маршрутизацию. Данные настройки позволят автоматически создавать таблицы маршрутизации на основе маршрутов, формируемых по данным протоколам, и распространять их между другими маршрутизаторами в рамках одной сети.

Внимание! Исполнения ViPNet Coordinator HW50 A, B и ViPNet Coordinator HW100 A, B не поддерживают динамическую маршрутизацию.

Настройку динамической маршрутизации должен выполнять опытный администратор, понимающий принципы работы протоколов динамической маршрутизации, в том числе протокола OSPF. Приведенная в руководстве информация носит справочный характер.

Настройка параметров динамических маршрутов от DHCP/PPP-протокола

Если на каком-либо сетевом интерфейсе ViPNet Coordinator HW включен режим DHCP и настроено автоматическое получение маршрутов от DHCP-сервера (см. «Настройка сетевых интерфейсов Ethernet» на стр. 50), то в процессе маршрутизации будут использоваться эти маршруты. В этом случае выполните следующие дополнительные настройки:

1. Настройте административную дистанцию для протокола DHCP (см. «Настройка административной дистанции для маршрутов DHCP-протокола» на стр. 191). Это позволит определить приоритет маршрутов DHCP-сервера среди всех маршрутов, имеющихся в таблице RIB (статических маршрутов, маршрутов протокола OSPF, если такие есть).
2. Если режим DHCP включен на нескольких сетевых интерфейсах, настройте метрики DHCP-протокола на каждом сетевом интерфейсе, на котором включен этот режим (см. «Настройка метрики для маршрутов DHCP-протокола» на стр. 191). Это позволит определить приоритет среди маршрутов DHCP-сервера в одну и ту же сеть, полученных с разных сетевых интерфейсов.

Если ViPNet Coordinator HW подключен к сети 3G или 4G и получает информацию для маршрута по умолчанию от PPP-сервера провайдера (см. «Подключение к мобильной сети 3G, 4G» на стр. 55), и при этом он также подключен к другой сети и получает в ней маршрутную информацию от DHCP-сервера, то в этом случае задайте метрику для PPP-протокола (см. «Настройка метрики для маршрутов PPP-протокола» на стр. 192). Это позволит определить, какой маршрут по умолчанию (сформированный на основе шлюза по умолчанию от PPP-сервера или предоставленный DHCP-

сервером) является приоритетным. Административная дистанция для этого протокола не задается и равна дистанции для протокола DHCP.

Настройка административной дистанции для маршрутов DHCP-протокола

Если на ViPNet Coordinator HW настроено взаимодействие с DHCP-сервером, от которого поступают динамические маршруты для маршрутизации IP-трафика, а также настроена статическая маршрутизация (см. «Настройка статической маршрутизации» на стр. 186) или динамическая маршрутизация по протоколу OSPF (см. «Настройка параметров динамической маршрутизации по протоколу OSPF» на стр. 195), то требуется задать административную дистанцию для маршрутов DHCP-протокола. Административная дистанция определит приоритет данных маршрутов среди всех остальных, и в случае, если будет обнаружено несколько маршрутов в одну и ту же сеть из разных источников, будет выбран тот маршрут, у которого выше приоритет. Чем меньше значение административной дистанции (см. глоссарий, стр. 266), тем выше приоритет маршрута.

В настройках ViPNet Coordinator HW вы можете выполнить следующие действия:

- Задать административную дистанцию для маршрутов, поступающих от DHCP-сервера. Эта дистанция будет определять приоритет всех маршрутов DHCP-сервера, независимо от того, на какой сетевой интерфейс они поступили.

  Чтобы задать административную дистанцию для маршрутов от DHCP-сервера, выполните команду:

  ```
  hostname# inet dhcp client route-distance <1-255>
  ```

- Задать административную дистанцию для маршрутов по умолчанию вместе с общей административной дистанцией для всех маршрутов DHCP-сервера. Эта дистанция будет определять приоритет только маршрутов по умолчанию. Если эта дистанция не задана, то у маршрутов по умолчанию дистанция и приоритет соответственно будет таким же, как и у других маршрутов DHCP-сервера.

  Чтобы задать административную дистанцию для маршрутов по умолчанию от DHCP-сервера, выполните команду:

  `hostname# inet dhcp client route-distance <1-255> [default-route <1-255>]`, где `default-route <1-255>` — значение административной дистанции для маршрутов по умолчанию.

При необходимости перед или после настройки административной дистанции вы можете посмотреть, на каких сетевых интерфейсах включен режим DHCP и какая административная дистанция задана. Подробнее см. раздел Просмотр настроек DHCP в режиме клиента (на стр. 193).

Настройка метрики для маршрутов DHCP-протокола

Если на ViPNet Coordinator HW режим DHCP включен на нескольких сетевых интерфейсах, то может сложиться ситуация, когда с этих интерфейсов поступят одинаковые маршруты от DHCP-сервера в одну и ту же сеть. В этом случае требуется определить, какой маршрут является наилучшим, чтобы

добавить его в FIB (см. «Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW» на стр. 179) и использовать при маршрутизации. Параметром, позволяющим выбрать наилучший маршрут, выступает метрика (см. глоссарий, стр. 270). Ее можно задать на каждом сетевом интерфейсе, на котором включен режим DHCP, чтобы определить приоритет маршрутов DHCP-сервера на разных интерфейсах. Чем меньше значение метрики, тем выше приоритет маршрута. Если на интерфейсах заданы одинаковые метрики, то поступившие маршруты в одну и ту же сеть будут объединены в один маршрут с несколькими шлюзами.

В ViPNet Coordinator HW вы можете задать или удалить метрику для маршрутов DHCP-сервера на конкретном сетевом интерфейсе (далее — специфичная метрика). Если специфичная метрика не задана, то вместо нее используется метрика по умолчанию.

- Чтобы добавить специфичную метрику на сетевом интерфейсе, выполните команду:

  ```
  hostname# inet ifconfig <имя интерфейса> dhcp route-metric <1-255>
  ```

  Примечание. Задать специфичную метрику для протокола DHCP вы можете на сетевых интерфейсах следующих типов:

  - Ethernet (см. «Настройка сетевых интерфейсов Ethernet» на стр. 50).
  - VLAN (см. «Организация обработки трафика из нескольких VLAN» на стр. 53).
  - Wi-Fi (wlan) (см. «Подключение к сети Wi-Fi» на стр. 58).
  - Агрегированный интерфейс (см. «Использование агрегированных сетевых интерфейсов» на стр. 61).

  Если режим DHCP включен только на одном сетевом интерфейсе, то настройка специфичной метрики не требуется, поскольку для группы маршрутов в рамках одного сетевого интерфейса она всегда будет одинаковой.

- Чтобы удалить метрику, заданную на сетевом интерфейсе, выполните команду:

  ```
  hostname# inet ifconfig <имя интерфейса> dhcp route-metric none
  ```

  После удаления метрики на этом интерфейсе будет использоваться метрика по умолчанию.

При необходимости перед или после настройки метрик вы можете посмотреть, на каких сетевых интерфейсах включен режим DHCP и какие метрики заданы. Подробнее см. раздел Просмотр настроек DHCP в режиме клиента (на стр. 193).

Настройка метрики для маршрутов PPP-протокола

Если ViPNet Coordinator HW подключен к мобильной сети 3G, 4G (см. «Подключение к мобильной сети 3G, 4G» на стр. 55), то при первом соединении с сервером провайдера он получает от него по протоколу PPP (см. глоссарий, стр. 265) IP-адрес шлюза по умолчанию, на основе которого создается маршрут по умолчанию (см. глоссарий, стр. 269).

Примечание. Шлюз по умолчанию поступает на ViPNet Coordinator HW от PPP-сервера только, если была выполнена команда `hostname# inet usb-modem set route on` (см. «Подключение к мобильной сети 3G, 4G» на стр. 55).

Если кроме подключения к сети 3G, 4G, ViPNet Coordinator HW также подключен к другим сетям, в которых информация об адресах и маршрутах распространяется DHCP-сервером, то он также будет получать маршруты по умолчанию по протоколу DHCP. Если требуется, чтобы маршрут по умолчанию PPP-протокола имел выше приоритет, чем маршруты по умолчанию DHCP-протокола, и использовался в процессе маршрутизации, требуется задать метрику для протокола PPP (далее – специфичную метрику). Причем она должна быть меньше метрики DHCP-протокола на каждом сетевом интерфейсе, на которых они заданы. Если специфичная метрика не будет задана, то вместо нее будет использоваться метрика по умолчанию (см. «Изменение метрики по умолчанию для маршрутов от DHCP/PPP-протокола» на стр. 193). В этом случае маршруты по умолчанию PPP- и DHCP-протокола будут иметь одинаковый приоритет.

- Чтобы добавить специфичную метрику для протокола PPP, выполните команду:

  ```
  hostname# inet usb-modem set route-metric <1-255>
  ```

  

  Примечание. Данную операцию вы можете выполнить только при следующих условиях:

  - включено использование 3G-, 4G-модема;
  - разрешено применение маршрута по умолчанию от PPP-сервера провайдера.

  При невыполнении данных условий появится сообщение, что метрика не может быть задана. О том, как включить использование USB-модема и настроить применение маршрута по умолчанию, см. в разделе Подключение к мобильной сети 3G, 4G (на стр. 55).

- Чтобы удалить специфичную метрику для протокола PPP, выполните команду:

  ```
  hostname# inet usb-modem set route-metric none
  ```

  После удаления метрики для протокола PPP будет использоваться метрика по умолчанию.

Изменение метрики по умолчанию для маршрутов от DHCP/PPP-протокола

Если для маршрутов DHCP/PPP-протокола не задавались специфичные метрики, то для определения приоритета маршрутов этих протоколов будет использоваться метрика по умолчанию. Первоначально метрика по умолчанию равна 70. При необходимости вы можете изменить это значение. Для этого выполните команду:

```
hostname# inet dhcp client route-default-metric <1-255>
```

Просмотр настроек DHCP в режиме клиента

Вы можете просмотреть следующие параметры DHCP в режиме клиента:

- административная дистанция, которая задана для маршрутов, поступающих от DHCP-сервера (см. «Настройка административной дистанции для маршрутов DHCP-протокола» на стр. 191);
- метрика по умолчанию (см. «Настройка метрики для маршрутов DHCP-протокола» на стр. 191);

- сетевые интерфейсы, на которых включен режим DHCP;
- специфичные метрики на сетевых интерфейсах, если такие заданы;
- разрешение на автоматическое получение IP-адресов, адресов DNS- и NTP-серверов.

Чтобы посмотреть указанные настройки, выполните команду:

```
hostname> inet show dhcp client
```

В результате выполнения команды указанные настройки будут отражены в следующем виде:

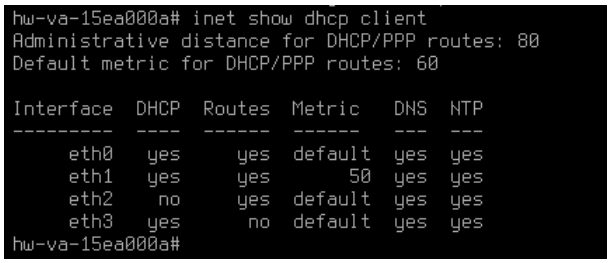

Рисунок 19. Просмотр настроек DHCP-режима

Просмотр маршрутов DHCP-сервера

Если вы хотите просмотреть список маршрутов, полученных от DHCP/PPP-сервера и добавленных в RIB (см. «Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW» на стр. 179), выполните команду:

```
hostname> inet show routing dhcp
```

В результате выполнения команды появится список маршрутов в следующем виде:

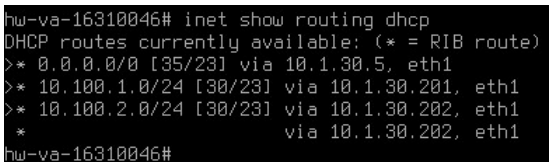

Рисунок 20. Просмотр маршрутов DHCP-сервера

В списке для каждого маршрута указаны: атрибут, IP-адрес назначения, маска сети, шлюз и имя сетевого интерфейса, на который поступил маршрут. Если маршрут добавлен в таблицу RIB (см. «Принципы формирования таблиц маршрутизации в ViPNet Coordinator HW» на стр. 179), то перед ним стоит атрибут «*».

Настройка параметров динамической маршрутизации по протоколу OSPF

Совет. Настройка динамической маршрутизации по протоколу OSPF имеет смысл в том случае, если в сети, в которой установлен ViPNet Coordinator HW, одновременно выполняются следующие условия:

- используются другие OSPF-маршрутизаторы и они напрямую связаны с ViPNet Coordinator HW;
- поддерживается многоадресное вещание (multicast);
- по требованиям безопасности вашей организации разрешена передача IP-трафика по протоколу OSPF между сетевыми узлами.

В процессе настройки маршрутизации по протоколу OSPF вам потребуется:

- включить использование протокола OSPF;
- указать сети, к которым подключен ViPNet Coordinator HW и которые будут участвовать в маршрутизации по протоколу OSPF, и области маршрутизации (см. глоссарий, стр. 270);
- создать ряд сетевых фильтров.

При настройке динамической маршрутизации по протоколу OSPF следует учитывать следующие рекомендации и ограничения:

- В случае сбоя на одном из маршрутов для переключения на альтернативный маршрут может потребоваться до 15 минут. Чтобы уменьшить время переключения, рекомендуется в файле `iplir.conf` в секциях `[id]`, соответствующих связанным координаторам ViPNet, задать более короткий период опроса, изменив значение параметра `checkconnection_interval` (подробнее см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам»). Например, можно задать для параметра значение 30 секунд. При этом необходимо учитывать, что сокращение периода опроса приведет к увеличению количества служебного трафика между координаторами, в результате чего может снизиться производительность координаторов.
- Если ваш ViPNet Coordinator HW непосредственно связан с другими координаторами ViPNet, рекомендуется в файле `iplir.conf` в секциях `[id]`, соответствующих этим координаторам, указать в параметре `accessiplist` метрику `1`. В этом случае при наличии нескольких альтернативных маршрутов защищенный трафик всегда будет передаваться по кратчайшему маршруту — через соседний координатор ViPNet.

При необходимости вы также можете настроить перераспределение маршрутов по протоколу OSPF.

Настройка протокола OSPF

Для настройки протокола OSPF выполните следующие действия:

1 Включите использование протокола с помощью команды:

```
hostname# inet ospf mode on
```

2 Укажите сеть, в которой осуществляется обмен информацией по протоколу OSPF, с помощью команды:

```
hostname# inet ospf network add <IP-адрес назначения> netmask <маска сети> area <0-4294967295>,
```

К данной сети должен быть подключен сетевой интерфейс ViPNet Coordinator HW.

С помощью параметра `area` задайте область маршрутизации, в которую входит ViPNet Coordinator HW (см. «Общие сведения для работы по протоколу OSPF» на стр. 184).

3 Создайте на ViPNet Coordinator HW следующие сетевые фильтры открытой сети:

- Фильтры, разрешающие входящий однонаправленный (unicast) IP-трафик и многоадресный (multicast) IP-трафик по протоколу OSPF от всех соседних OSPF-маршрутизаторов, с которыми взаимодействует ViPNet Coordinator HW:

  ```
  hostname# firewall local add 1 rule "Rule 1" src <IP-адрес OSPF-маршрутизатора> dst @local service @OSPF pass
  ```

  ```
  hostname# firewall local add 2 rule "Rule 2" src <IP-адрес OSPF-маршрутизатора> dst @multicast service @OSPF pass
  ```

  Фильтры должны быть созданы для каждого соседнего OSPF-маршрутизатора в области, в которой находится ViPNet Coordinator HW.

  Примечание. Если вы обновили ViPNet Coordinator HW с версии 4.1 и ниже, то системный объект `@OSPF` вам требуется создать вручную. В фильтрах вместо `service @OSPF` вы можете указать `proto 89`.

- Фильтр, разрешающий исходящий IP-трафик ViPNet Coordinator HW по протоколу OSPF на любой IP-адрес назначения:

  ```
  hostname# firewall local add 3 rule "Rule 3" src @local dst @any service @OSPF pass
  ```

Подробнее о создании сетевых фильтров см. в разделе Создание сетевого фильтра (на стр. 124).

Внимание! Если вы не настроите указанные фильтры, то ViPNet Coordinator HW не сможет работать по протоколу OSPF (см. «Общие сведения для работы по протоколу OSPF» на стр. 184).

Если впоследствии потребуется отказаться от использования протокола OSPF для маршрутизации, выполните команду:

```
hostname# inet ospf mode off
```

Если в подсети прекращено использование протокола OSPF, удалите ее. Для этого выполните команду:

```
hostname# inet ospf network delete <IP-адрес назначения> netmask <маска сети> area <0-4294967295>
```

При необходимости вы можете посмотреть параметры настройки протокола OSPF, а также информацию о маршрутизаторах, работающих по протоколу OSPF и связях между ними. Подробнее см. раздел Просмотр настроек протокола OSPF (на стр. 198).

Настройка перераспределения маршрутов

Протокол OSPF позволяет осуществлять перераспределение (redistribute) маршрутов. В каких случаях это может требоваться?

В одной разветвленной сети может быть образовано нескольких автономных систем (см. глоссарий, стр. 266) по причине использования в разных подсетях разных протоколов маршрутизации. Например, в одной подсети может использоваться статическая маршрутизация, в другой — динамическая маршрутизация по протоколу OSPF. В результате это образует две автономные системы. Чтобы системы могли взаимодействовать друг с другом, на маршрутизаторе, который установлен на границе этих систем, требуется настроить перераспределение маршрутов. В результате произойдет обмен маршрутной информацией, получаемой из этих систем, и они будут иметь связь друг с другом.

На рисунке ниже показана схема взаимодействия нескольких автономных систем.

Рисунок 21. Пример топологии сети с настроенным перераспределением маршрутов

В приведенном примере рассматривается 4 взаимодействующих подсети, 2 из которых входят в одну автономную систему, поскольку они работают по одному протоколу OSPF, остальные — в две другие автономные системы, поскольку они работают по протоколам Static и DHCP соответственно. При этом маршрутизатор 1 является граничным маршрутизатором (см. «Общие сведения для работы по протоколу OSPF» на стр. 184), поскольку он расположен на границе трех автономных систем. Чтобы маршрутизаторы 2 и 3 могли получить информацию о маршрутах, используемых на маршрутизаторах 4 и 5, и, соответственно, чтобы узлы сети филиала и сетей партнеров могли обмениваться IP-трафиком друг с другом, требуется настроить перераспределение маршрутов на маршрутизаторе 1. После настройки маршруты, полученные из сетей партнеров (от маршрутизаторов 4 и 5) на маршрутизаторе 1, будут передаваться на маршрутизатор 2, а с маршрутизатора 2 — на маршрутизатор 3. В результате маршрутизатор 3 будет знать статические маршруты, прописанные в сети партнера 1, и маршруты, выдаваемые DHCP-сервером, в сети партнера 2.

Чтобы включить перераспределение маршрутов на ViPNet Coordinator HW, выполните команду:

```
hostname# inet ospf redistribute add {static | dhcp}
```

указав один из параметров:

- `static` — включение перераспределения статических маршрутов;
- `dhcp` — включение перераспределения маршрутов от DHCP-сервера.

Если перераспределение указанного типа маршрутов было включено ранее, появится соответствующее сообщение.

Чтобы выключить перераспределение маршрутов, выполните команду:

```
hostname# inet ospf redistribute delete {static | dhcp}
```

В параметрах протокола OSPF указывается, настроено перераспределение маршрутов или нет. Подробнее см. раздел Просмотр настроек протокола OSPF (на стр. 198).

Просмотр настроек протокола OSPF

Вы можете просмотреть следующие параметры протокола OSPF:

- статус использования протокола OSPF: включен или выключен;
- список сетей, в которых ViPNet Coordinator HW осуществляет маршрутизацию по протоколу OSPF (IP-адреса, маски и области);
- статус перераспределения маршрутов: доступно или недоступно.

Чтобы посмотреть указанные настройки, выполните команду:

```
hostname> inet show ospf configuration
```

В результате выполнения команды указанные настройки будут отображены в следующем виде:

```
hw-va-16310045# inet show ospf configuration
OSPF protocol is enabled
OSPF networks defined:
Destination      Netmask          OSPF Area
-----------      -------          ---------
10.0.2.0         255.255.255.0    1
10.0.5.0         255.255.255.0    0
Redistribution of DHCP routes is enabled.
hw-va-16310045#
```

Рисунок 22. Просмотр настроек протокола OSPF

Просмотр информации базы данных состояний каналов связи по протоколу OSPF

Кроме настроек протокола OSPF вы также можете просмотреть информацию о состоянии каналов связей между всеми маршрутизаторами, работающими по протоколу OSPF. Данная информация содержится в базе данных (link state database) (см. «Общие сведения для работы по протоколу OSPF» на стр. 184) и синхронизируется на всех OSPF-маршрутизаторах. Именно на ее основе вычисляются динамические маршруты протокола OSPF. Для просмотра информации базы данных состояний каналов связи, выполните команду:

```
hostname> inet show ospf database
```

В результате выполнения команды отобразится содержимое базы данных в следующем виде:

```
hw-va-16310045# inet show ospf database

       OSPF Router with ID (10.0.3.2)

                Router Link States (Area 0.0.0.0)

Link ID         ADV Router      Age  Seq#       CkSum  Link count
10.0.3.2        10.0.3.2        155 0x8000017d 0x590a 2
10.1.30.5       10.1.30.5       220 0x8000029f 0x3fa0 2

                Net Link States (Area 0.0.0.0)

Link ID         ADV Router      Age  Seq#       CkSum
10.0.5.5        10.1.30.5       751 0x80000263 0x3541

                AS External Link States

Link ID         ADV Router      Age  Seq#       CkSum  Route
10.0.1.0        10.1.30.5      1551 0x80000182 0x9061 E2 10.0.1.0/24 [0x0]
10.0.2.0        10.1.30.5      1001 0x80000183 0x836c E2 10.0.2.0/24 [0x0]
10.0.3.0        10.1.30.5       210 0x80000182 0x7a75 E2 10.0.3.0/24 [0x0]
10.0.4.0        10.1.30.5       341 0x80000182 0x6f7f E2 10.0.4.0/24 [0x0]
10.100.1.0      10.1.30.5       911 0x8000029d 0xe1ad E2 10.100.1.0/24 [0x0]
10.100.2.0      10.1.30.5       180 0x8000029f 0xe0aa E2 10.100.2.0/24 [0x0]
192.168.0.0     10.1.30.5       821 0x80000182 0x6c27 E2 192.168.0.0/16 [0x0]


hw-va-16310045#
```

Рисунок 23. Просмотр базы данных состояний связи

Просмотр информации о соседних маршрутизаторах

При составлении маршрутов по протоколу OSPF участвуют все OSPF-маршрутизаторы, находящиеся в области маршрутизации ViPNet Coordinator HW (см. глоссарий, стр. 270). При

необходимости вы можете просмотреть сведения об OSPF-маршрутизаторах, которые являются соседними для ViPNet Coordinator HW (см. глоссарий, стр. 269). Для этого выполните команду:

```
hostname> inet show ospf neighbour
```

В результате выполнения команды отобразится информация обо всех соседних OSPF-маршрутизаторах:

```
hw-va-16310045# inet show ospf neighbour

    Neighbor ID Pri State           Dead Time Address         Interface            RXmtL RqstL DBsmL
10.1.30.5         1 Full/DR           35.310s 10.0.5.5        eth0:10.0.5.2            0     0     0
hw-va-16310045#
```

Рисунок 24: Просмотр информации о соседних OSPF-маршрутизаторах

По каждому маршрутизатору отображается следующая информация:

- уникальный идентификатор OSPF-маршрутизатора, под которым он известен другим маршрутизаторам при работе по протоколу OSPF;
- приоритет маршрутизатора;
- тип маршрутизатора (см. «Общие сведения для работы по протоколу OSPF» на стр. 184) (в случае, приведенном на рисунке, маршрутизатор-сосед является назначенным, на что указывает значение `DR` в поле `State`);
- интервал простоя маршрутизатора, по истечении которого он будет считаться неактивным (выключенным);
- IP-адрес маршрутизатора, по которому он доступен для обмена информацией по протоколу OSPF;
- IP-адрес активного сетевого интерфейса маршрутизатора;
- другие параметры.

Просмотр маршрутов, сформированных по протоколу OSPF

Если вы хотите просмотреть список динамических маршрутов, сформированных по протоколу OSPF и присутствующих в RIB, выполните команду:

```
hostname> inet show routing ospf
```

В результате выполнения команды появится списки маршрутов в трех разделах:

Рисунок 25. Просмотр маршрутов протокола OSPF

В каждом из разделов отображаются следующие маршруты:

- `OSPF network routing table` — маршруты, которые были сформированы по OSPF-протоколу в той области маршрутизации, к которой подключен ViPNet Coordinator HW (см. глоссарий, стр. 270). Если ViPNet Coordinator HW является межобластным маршрутизатором (ABR) (см. «Общие сведения для работы по протоколу OSPF» на стр. 184), то в этом разделе будут присутствовать маршруты из всех областей, которые он объединяет.
- `OSPF router routing table` — маршруты до соседних OSPF-маршрутизаторов (см. глоссарий, стр. 269), которые находятся в области маршрутизации ViPNet Coordinator HW и выполняют перераспределение маршрутов (см. «Настройка перераспределения маршрутов» на стр. 197).
- `OSPF external routing table` — маршруты в другие автономные системы, поступившие при перераспределении от соседних OSPF-маршрутизаторов.

13

Настройка транспортного модуля

Назначение и функциональность транспортного модуля

Транспортный модуль предназначен для надежной и безопасной передачи транспортных конвертов между связанными узлами сети ViPNet посредством протоколов TCP (этот канал передачи называется MFTP) и SMTP/POP3. Транспортный модуль входит в состав ViPNet Coordinator HW в виде демона mftpd и программы mailtrans. Он обеспечивает выполнение координатором функции транспортного сервера, принимает непосредственное участие в удаленном обновлении справочников и ключей на узлах, удаленном обновлении программного обеспечения, а также приеме политик безопасности из программы ViPNet Policy Manager (см. глоссарий, стр. 265).

Примечание. Транспортный модуль ViPNet Coordinator HW в исполнениях ViPNet Coordinator HW50 A, B и ViPNet Coordinator HW100 A, B имеет ограниченную функциональность. На таких ViPNet Coordinator HW не поддерживается функция транспортного сервера, то есть транспортный модуль выполняет только прием справочников, ключей и ПО ViPNet для собственного узла. Конверты с аналогичной информацией, адресованные другим узлам сети, им не принимаются.

Поэтому при формировании структуры сети в программе ViPNet Центр управления сетью на ViPNet Coordinator HW в исполнениях ViPNet Coordinator HW50 A, B и ViPNet Coordinator HW100 A, B не должны регистрироваться клиенты ViPNet. Транспортный модуль в составе ViPNet Coordinator HW с остальными ролями не имеет ограничений по функциональности, и на этих ViPNet Coordinator HW можно регистрировать клиенты.

Транспортные конверты представляют собой файлы с данными, которыми обмениваются между собой приложения, входящие в состав ПО ViPNet. Размер конверта, передаваемый транспортным модулем, не может превышать 2 Гбайт (2 097 151 Кбайт). Транспортный модуль передает конверты в соответствии с адресами получателей, прописанными в заголовках этих конвертов.

Для обмена транспортными конвертами на защищенных узлах могут использоваться каналы следующих типов:

- MFTP — при связи по такому каналу узел-отправитель конвертов устанавливает TCP-соединение с узлом-получателем конвертов, проводится взаимная аутентификация узлов и осуществляется передача конвертов между узлами через транспортные серверы. Данный тип канала по умолчанию устанавливается для защищенных сетевых узлов.
- SMTP/POP3 — при связи по такому каналу транспортный модуль переадресует конверты для отправки модулю почтового обмена MailTrans, который передает их через сервер SMTP, а также забирает с сервера POP3 конверты, предназначенные для данного узла. То есть канал SMTP/POP3 позволяет использовать для обмена транспортными конвертами почтовые серверы, что удобно, например, в случае передачи конвертов между сетевыми узлами разных организаций, в которых по каким-либо причинам затруднен выход в Интернет по TCP-каналам (см. «Настройка взаимодействия с модулем почтового обмена» на стр. 206).

- Viaroute — такой канал используется для связи с узлами доверенной сети (см. глоссарий, стр. 267), если настроено межсетевое взаимодействие (см. глоссарий, стр. 269). При этом транспортный модуль передает и получает конверты от узлов доверенной сети через шлюзовой координатор. Данный тип канала по умолчанию задается для координаторов доверенной сети при установлении межсетевого взаимодействия.

Настройка транспортного модуля выполняется путем редактирования конфигурационного файла `mftp.conf`. Подробнее о параметрах, содержащихся в файле `mftp.conf`, см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам».

Выбор канала передачи конвертов

По умолчанию для обмена конвертами с защищенными узлами, связанными с ViPNet Coordinator HW, используется канал MFTP. Чтобы изменить тип канала на SMTP/POP3, выполните следующие действия:

1. Завершите работу демона mftpd с помощью команды:

   ```
   hostname> mftp stop
   ```

2. Откройте конфигурационный файл `mftp.conf` для редактирования с помощью команды:

   ```
   hostname# mftp config
   ```

3. В секции `[channel]` нужного узла измените следующие параметры:

   - установите параметр `type` в значение `smtp`;
   - в параметре `reportaddress` укажите адрес электронной почты для отправки исходящих конвертов в формате: `<имя пользователя>@<сервер>.<домен>`

4. Сохраните изменения в файле `mftp.conf` и запустите демон mftpd с помощью команды:

   ```
   hostname> mftp start
   ```

Настройка взаимодействия с модулем почтового обмена

Если для каких-либо узлов в вашей организации предполагается использование каналов SMTP/POP3, настройте параметры взаимодействия транспортного модуля с модулем почтового обмена MailTrans. Для этого выполните следующие действия:

1. Завершите работу демона mftpd с помощью команды:

   ```
   hostname> mftp stop
   ```

2. Откройте конфигурационный файл `mftp.conf` для редактирования с помощью команды:

   ```
   hostname# mftp config
   ```

3. В секции `[mailtrans]` укажите значения следующих параметров:

   - `frommailbox` —адрес электронной почты отправителя SMTP-конвертов в формате: `<имя пользователя>@<сервер>.<домен>`
   - `inputmailbox` — адрес электронной почты, по которому модуль почтового обмена будет забирать конверты по протоколу POP3, в формате: `<имя пользователя>:<пароль>@<IP-адрес POP3-сервера>`
   - `outputmailbox` — IP-адрес SMTP-сервера, на который модуль почтового обмена будет отправлять конверты по протоколу SMTP.

4. Сохраните изменения в файле `mftp.conf` и запустите демон mftpd с помощью команды:

   ```
   hostname> mftp start
   ```

Настройка протоколирования событий транспортного модуля

Демон mftpd записывает информацию о событиях, происходящих в процессе работы транспортного модуля, в журнал устранения неполадок (см. «Работа с журналом устранения неполадок» на стр. 221).

Чтобы изменить параметры записи событий в журнал устранения неполадок транспортного модуля, выполните следующие действия:

1. Завершите работу демона mftpd с помощью команды:

   ```
   hostname> mftp stop
   ```

2. Откройте файл `mftp.conf` для редактирования с помощью команды:

   ```
   hostname# mftp config
   ```

3. В секции `[debug]` при необходимости измените значения следующих параметров:

   - `debuglevel` — уровень детализации информации, выводимой в журнал. Возможные значения: от `-1` до `5` (по умолчанию — `3`). Чем выше уровень детализации, тем более подробная информация выводится в журнал. Значение параметра `-1` выключает ведение журнала.
   - `debuglogfile` — источник информации, выводимой в журнал, в формате: `syslog:<facility.level>`, где:
     - `facility` — процесс, формирующий информацию. Возможные значения: `kern` (ядро), `user` (пользовательские программы), `mail` (почтовая система) или `daemon` (демоны).
     - `level` — уровень важности информации. Возможные значения: `err` (ошибка), `info` (информационное сообщение) или `debug` (отладочная информация).

     Значение параметра `debuglogfile` по умолчанию — `syslog:daemon.debug`.

4. Запустите демон mftpd с помощью команды:

   ```
   hostname> mftp start
   ```

Настройка прямой маршрутизации между сетями ViPNet

Если между двумя сетями ViPNet установлено межсетевое взаимодействие, то по умолчанию транспортные конверты от клиентов одной сети к клиентам другой сети проходят через шлюзовые координаторы данных сетей и обрабатываются на этих координаторах транспортным модулем MFTP. Для снижения нагрузки на шлюзовые координаторы администраторы могут настроить прямую маршрутизацию между координаторами своей и доверенной сетей. В этом случае конверты будут обрабатываться транспортным модулем MFTP на координаторах сетей и не будут обрабатываться на шлюзовых координаторах.

Если в качестве координаторов сетей используется ViPNet Coordinator HW, то для настройки прямой маршрутизации необходимо внести изменения в секции `[channel]`.

Рассмотрим пример взаимодействия двух сетей, в которых установлены ViPNet Coordinator HW HW_1 и HW_2, а координаторы Шлюз_1 и Шлюз_2 выполняют функции шлюзовых. Пусть требуется настроить прямую маршрутизацию без обработки конвертов на обоих шлюзовых координаторах.

Рисунок 26. Прямая маршрутизация в обход обоих шлюзовых координаторов

По умолчанию на каждом ViPNet Coordinator HW в секции `[channel]`, содержащей настройки канала обмена с ViPNet Coordinator HW другой сети, заданы следующие параметры:

Настройки на HW_1	Настройки на HW_2
`id = <идентификатор HW_2>`	`id = <идентификатор HW_1>`
`name = HW_2_<номер_сети_2>`	`name = HW_1_<номер_сети_1>`
`type = mftp`	`type = mftp`
`off_flag = yes`	`off_flag = yes`
`call_flag = no`	`call_flag = no`
`ip = <IP-адрес HW_2>`	`ip = <IP-адрес HW_1>`
`call_timeout = -1`	`call_timeout = -1`
`last_port = 5000`	`last_port = 5000`
`last_call = 0`	`last_call = 0`
`last_err = 0`	`last_err = 0`

Для настройки прямой маршрутизации между двумя ViPNet Coordinator HW в эти секции `[channel]` необходимо внести следующие изменения:

Настройки на HW_1	Настройки на HW_2
id = <идентификатор HW_2>	id = <идентификатор HW_1>
name = HW_2_<номер_сети_2>	name = HW_1_<номер_сети_1>
type = mftp	type = mftp
off_flag = no	off_flag = no
call_flag = yes	call_flag = yes
ip = <IP-адрес HW_2>	ip = <IP-адрес HW_1>
call_timeout = -1	call_timeout = -1
last_port = 5000	last_port = 5000
last_call = 0	last_call = 0
last_err = 0	last_err = 0
transit = <идентификатор HW_2>	transit = <идентификатор HW_1>

Таким образом, для настройки прямой маршрутизации надо включить канал обмена с ViPNet Coordinator HW другой сети и указать его идентификатор в параметре `transit`.

Пусть конверты, отправленные из первой сети во вторую, должны обрабатываться только на шлюзовом координаторе Шлюз_2, а конверты в обратном направлении должны передаваться без обработки на координаторах Шлюз_1 и Шлюз_2.

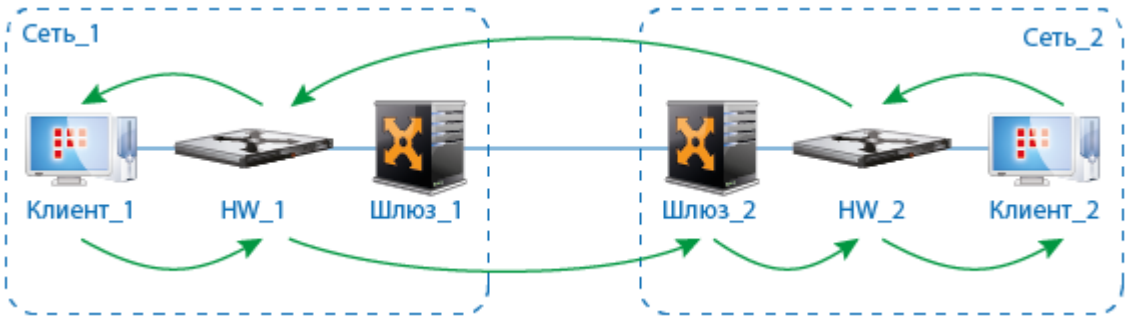

Рисунок 27. Прямая маршрутизация в обход одного шлюзового координатора

В этом примере настройки канала обмена на HW_2 должны быть такими же, как в первом примере, а на HW_1 необходимо внести следующие изменения:

Секция `[channel]` для канала обмена с HW_2	Секция `[channel]` для канала обмена со шлюзом Шлюз_2
id = <идентификатор HW_2>	id = <идентификатор Шлюз_2>
name = HW_2_<номер_сети_2>	name = Шлюз_2
type = mftp	type = mftp
off_flag = yes	off_flag = no

Секция `[channel]` **для канала обмена с HW_2**	**Секция** `[channel]` **для канала обмена со шлюзом Шлюз_2**
`call_flag = no`	`call_flag = yes`
`ip = <IP-адрес HW_2>`	`ip = <IP-адрес HW_1>`
`call_timeout = -1`	`call_timeout = -1`
`last_port = 5000`	`last_port = 5000`
`last_call = 0`	`last_call = 0`
`last_err = 0`	`last_err = 0`
`transit = <идентификатор Шлюз_2>`	`transit = <идентификатор Шлюз_2>`

Подробнее о параметрах настройки канала MFTP см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам», в разделе «Секция [channel]».

14

Ведение и просмотр журналов

Просмотр журнала регистрации IP-пакетов

Управляющий демон ведет журнал регистрации IP-пакетов, в который заносится информация обо всех IP-пакетах (зашифрованных и открытых), проходящих через сетевые интерфейсы узла.

В журнал заносится информация не о соединениях, а об IP-пакетах, которые проходят через сетевые интерфейсы, поэтому:

- каждый пропущенный и заблокированный локальный IP-пакет отображается в журнале один раз — на том интерфейсе, через который он прошел, или на том интерфейсе, на котором он был заблокирован, соответственно;
- каждый пропущенный транзитный IP-пакет отображается в журнале дважды: первый раз — на интерфейсе, через который он пришел, второй раз — на интерфейсе, через который он ушел.

Чтобы просмотреть записи из журнала регистрации IP-пакетов, выполните команду:

```
hostname# iplir view
```

В результате откроется окно **Set search parameters** для задания параметров поиска записей в журнале.

Внимание! При работе ViPNet Coordinator HW в режиме кластера горячего резервирования (см. глоссарий, стр. 268) журнал IP-пакетов можно просмотреть только на активном сервере кластера.

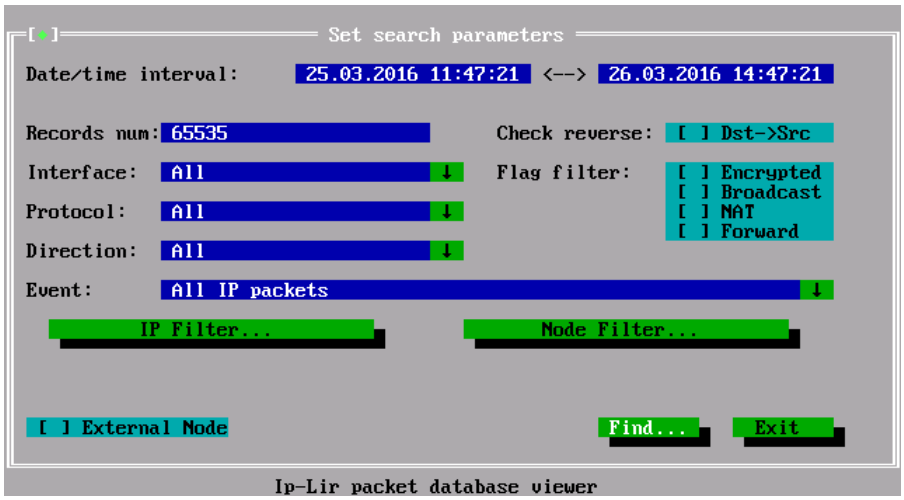

Рисунок 28. Задание параметров поиска записей в журнале регистрации IP-пакетов

По умолчанию предполагается просмотр записей обо всех пакетах, прошедших через сетевые интерфейсы собственного узла за последние сутки. Чтобы просмотреть такие записи, нажмите кнопку **Find**.

Чтобы выбрать записи из журнала регистрации IP-пакетов для просмотра, выполните следующие действия:

1 Укажите параметры пакетов:

- **Date/time interval** — период времени, в который были зарегистрированы пакеты в формате: `дд.мм.гггг чч:мм:сс <--> дд.мм.гггг чч:мм:сс`

  По умолчанию — текущие сутки.

- **Records num** — количество записей о зарегистрированных пакетах, которое будет отображено (по умолчанию — 65535).

- **Interface** — сетевой интерфейс, на котором были обработаны пакеты. По умолчанию — все доступные интерфейсы (**All**).

- **Protocol** — IP-протокол, по которому были переданы пакеты. Возможные значения:
  - **All** (по умолчанию) — любой протокол;
  - **ICMP** — только протокол ICMP;
  - **TCP** — только протокол TCP;
  - **UDP** — только протокол UDP.
  - **Except TCP, UDP, ICMP** — любой протокол, кроме TCP, UDP и ICMP.

- **Direction** — направление пакетов:
  - **All** (по умолчанию) — входящие и исходящие пакеты;
  - **Incoming** — входящие пакеты;
  - **Outgoing** — исходящие пакеты.

- **Event** — тип события, которому соответствует IP-пакет (см. «Типы событий в журнале регистрации IP-пакетов» на стр. 241). По умолчанию — события всех типов (**All**).

- **Check reverse** — включение в журнал ответных пакетов от получателя отправителю (**Dst -> Src**). Устанавливать этот флажок имеет смысл только, если в качестве отправителя и/или получателя пакетов указаны конкретные IP-адреса (**IP Filter**) или узлы (**Node Filter**).

- **Flag filter** — вид пакетов:
  - **Encrypted** — зашифрованные пакеты;
  - **Broadcast** — широковещательные пакеты;
  - **NAT** — транслированные пакеты;
  - **Forward** — транзитные пакеты.

  Вы можете выбрать один или несколько видов пакетов, установив соответствующие флажки. Транслированные (**NAT**) и транзитные (**Forward**) пакеты имеет смысл выбирать только для просмотра записей об IP-пакетах на координаторе.

2 Чтобы задать IP-адреса или порты отправителя или получателя пакетов, нажмите кнопку **IP Filter** и в открывшемся окне:

- если в списке **Protocol** вы выбрали все протоколы или все протоколы, кроме TCP, UDP и ICMP, укажите:
  - **Source IP address** — IP-адрес (**Value**) или диапазон IP-адресов (**Range**) отправителей пакетов. По умолчанию — любой адрес (**All**);
  - **Destination IP address** — IP-адрес (**Value**) или диапазон IP-адресов (**Range**) получателей пакетов. По умолчанию — любой адрес (**All**);
- если в списке **Protocol** вы выбрали протокол TCP или UDP, помимо IP-адресов отправителя и получателя, укажите:
  - **Source port** — порт (**Value**) или диапазон портов (**Range**) отправителей. По умолчанию — все порты (**All**);
  - **Destination port** — порт (**Value**) или диапазон портов (**Range**) получателей. По умолчанию — все порты (**All**);
- если в списке **Protocol** вы выбрали протокол ICMP, помимо IP-адресов отправителя и получателя, укажите:
  - **ICMP type** — тип (**Value**) или диапазон типов (**Range**) ICMP-пакетов.
  - **ICMP code** — код (**Value**) или диапазон кодов (**Range**) ICMP-пакетов.

3 Чтобы выбрать узел отправителя или получателя, нажмите кнопку **Node Filter** и в открывшемся окне:

3.1 Для выбора узла отправителя нажмите кнопку **Select source Node**, для выбора узла получателя — кнопку **Select destination Node**.

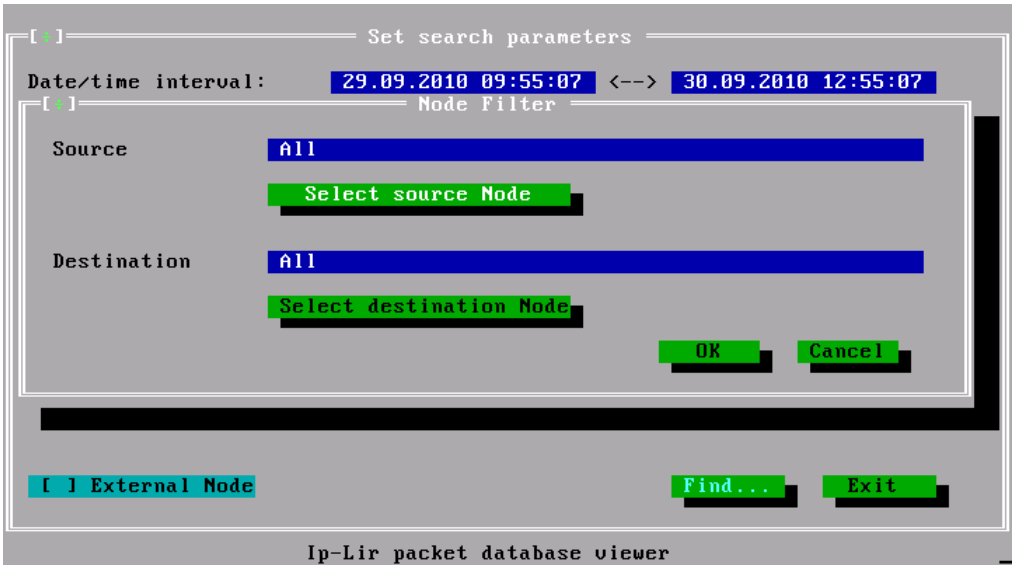

Рисунок 29. Выбор узла отправителя и получателя

В результате откроется таблица со списком защищенных узлов ViPNet, с которыми у данного узла есть связь. В левом столбце таблицы указан шестнадцатеричный идентификатор узла (**ID**), в правом — имя узла (**Name**).

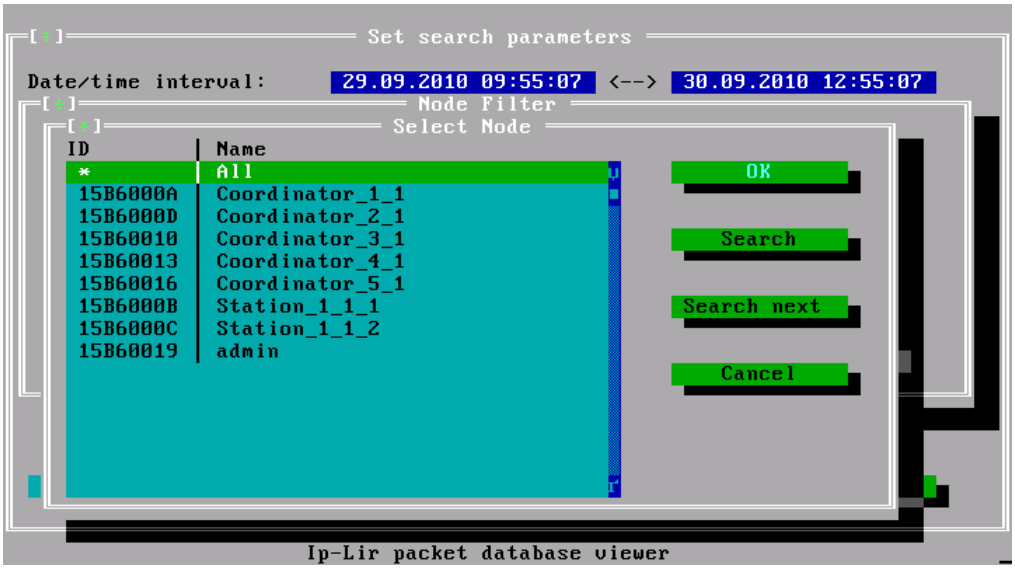

Рисунок 30. Выбор узла отправителя и получателя

3.2 Выберите нужный узел и нажмите кнопку **OK**.

Совет. Чтобы быстро найти нужный узел в списке, нажмите кнопку **Search**. В открывшемся окне введите часть идентификатора или имени узла и нажмите кнопку **OK**. В результате курсор будет переведен на строку первого узла, идентификатор или имя которого удовлетворяют условиям поиска. Для перехода к следующему такому узлу нажмите кнопку **Search next**.

---

4 Чтобы просмотреть журнал регистрации IP-пакетов удаленного узла:

Внимание! Для просмотра журнала регистрации IP-пакетов удаленного узла необходимо, чтобы этот узел был доступен.

4.1 Установите флажок **External Node** и введите пароль администратора узла ViPNet.

4.2 С помощью кнопки **Select** откройте окно со списком защищенных узлов ViPNet и выберите нужный узел.

5 Нажмите кнопку **Find**.

В результате откроется окно **View results**, содержащее таблицу со списком записей о пакетах, которые удовлетворяют выбранным параметрам. Записи упорядочены по времени регистрации. Таблица включает следующие столбцы:

- **Date/time** — дата и время регистрации пакета;
- **Dev** — интерфейс, на котором был обработан пакет;
- **Flags** — направление и флаги пакетов:
  - < — исходящий пакет;
  - > — входящий пакет;

- **C** — шифрованный пакет;
- **B** — широковещательный пакет;
- **D** — заблокированный пакет;
- **T** — транзитный пакет;
- **R** — пакет, который будет обработан правилами NAT открытой сети;
- **N** — пакет, который был обработан правилами NAT открытой сети;

- **Prot** — протокол, по которому был передан пакет;
- **Source IP** — IP-адрес отправителя пакета;
- **Port** — локальный порт для TCP- и UDP-пакетов или тип для ICMP-пакетов;
- **Destination IP** — IP-адрес получателя пакета;
- **Port** — порт удаленного компьютера для TCP- и UDP-пакетов или код для ICMP-пакетов.

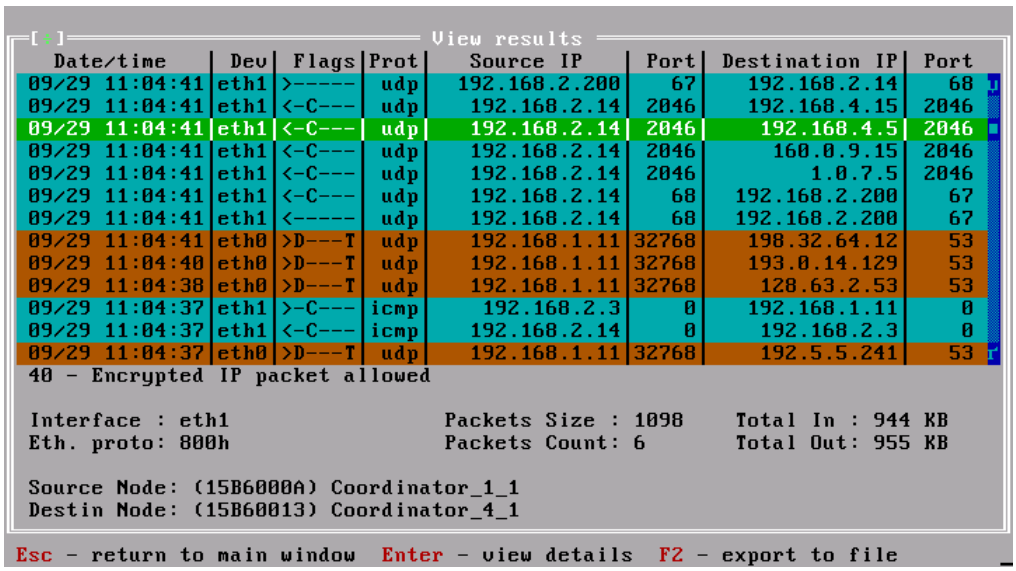

Рисунок 31. Просмотр записей из журнала регистрации IP-пакетов

Вы можете выполнить следующие действия:

- Экспортировать список записей в текстовый файл (например, для последующего анализа). Для этого в окне **View Results** нажмите клавишу **F2** и укажите файл для экспорта (см. «Экспорт журнала регистрации IP-пакетов» на стр. 218).
- Просмотреть краткую информацию о пакете, выбрав его в списке. В результате под списком отобразится следующая дополнительная информация:
  - тип события, которому соответствует IP-пакет (см. «Типы событий в журнале регистрации IP-пакетов» на стр. 241);
  - **Eth. proto** — шестнадцатеричный идентификатор протокола Ethernet;
  - **Packet Size** — размер пакетов, соответствующих данному типу события;

Примечание. Для зашифрованных пакетов в параметре **Packet size** отображается размер пакетов со всеми служебными заголовками.

- **Packet Count** — число пакетов, соответствующих выбранному типу события;
- **Total In** — суммарный размер всех входящих пакетов, соответствующих выбранному типу события;
- **Total Out** — суммарный размер всех исходящих пакетов, соответствующих выбранному типу события;

Примечание. Единицы изменения суммарного размера пакетов: **B** — байты, **KB** — килобайты, **MB** — мегабайты.

Символ «*» в поле **Total In** или **Total Out** пакета означает, что этот пакет не учтен в общем объеме входящего или исходящего трафика. Символ «N/A» — означает отсутствие информации о размере пакетов, соответствующих выбранному типу события.

- имена узла отправителя и узла получателя.

- Просмотреть подробную информацию о пакете, выбрав его в списке и нажав клавишу **Enter**. Подробная информация о пакете откроется в отдельном окне **Record details**.

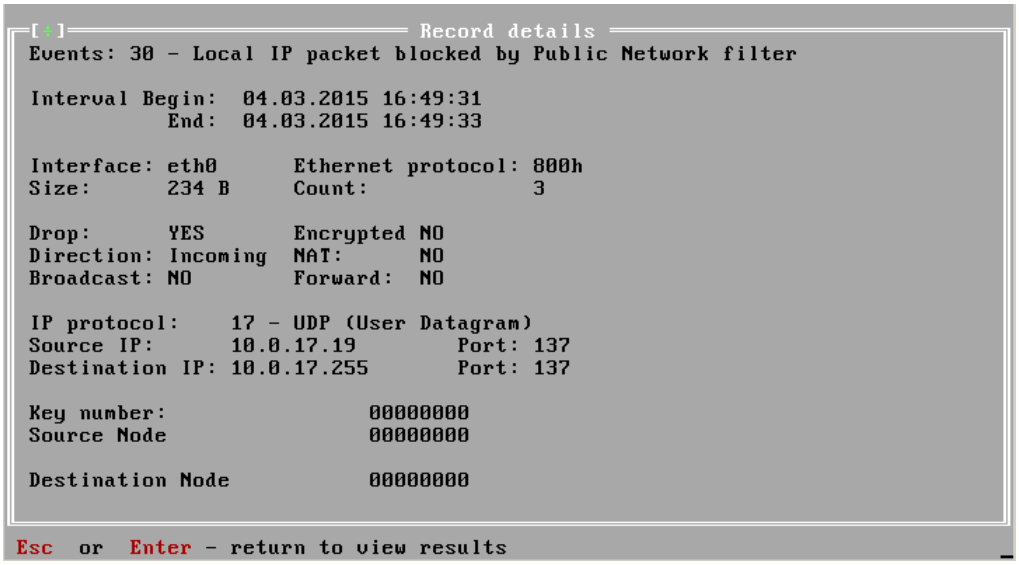

Рисунок 32. Просмотр подробной информации о пакете

- Для завершения просмотра журнала регистрации IP-пакетов или изменения параметров поиска пакетов в журнале нажмите клавишу **Esc**.

Экспорт журнала регистрации IP-пакетов

Чтобы экспортировать список записей из журнала регистрации IP-пакетов на USB-носитель, выполните следующие действия:

Внимание! Если вы используете удаленное подключение по SSH, то для корректного экспорта установите в настройках используемого ПО SSH-клиента тип терминала VT100+.

1 Выполните команду:

```
hostname> iplir view
```

2 В окне **Set search parameters** задайте параметры для выбора из журнала регистрации IP-пакетов нужных записей и нажмите кнопку **Find** (см. рисунок на стр. 212).

3 В окне **View results** со списком найденных записей нажмите клавишу **F2** (см. рисунок на стр. 216).

4 В открывшемся окне введите имя файла, в который вы хотите сохранить журнал, или выберите из списка существующий файл. Затем нажмите клавишу **Enter**.

5 Чтобы завершить просмотр журнала, нажмите клавишу **Esc**. Затем нажмите кнопку **Exit**.

6 Выполните команду:

```
hostname# admin export packetdb usb <имя файла>,
```

указав имя файла с журналом.

7 Подключите USB-носитель к одному из USB-разъемов ViPNet Coordinator HW и нажмите клавишу **Enter**.

8 В списке подключенных к ViPNet Coordinator HW USB-носителей и имеющихся на них разделов найдите нужный, введите его номер, нажмите клавишу **Enter** и дождитесь завершения экспорта.

В результате журнал регистрации IP-пакетов будет экспортирован в корневой каталог выбранного USB-носителя в виде архива `*.tar.gz`, содержащего текстовый файл. Данные в этом текстовом файле разделены символом «|».

Чтобы преобразовать журнал в удобный для чтения вид, распакуйте архив, откройте его с помощью программы Microsoft Excel и выполните импорт данных, указав в качестве разделителя символ «|». В результате данные отобразятся в таблице, содержащей следующие столбцы:

- **Time begin** — дата и время начала регистрации пакета;
- **Time end** — дата и время окончания регистрации пакета;
- **Dev** — интерфейс, на котором был обработан пакет;

- **Eth** — шестнадцатеричный идентификатор протокола Ethernet;
- **Flags** — направление и флаги пакета;
- **Prot** — протокол, по которому был передан пакет;
- **Source IP** — IP-адрес отправителя пакета;
- **Port** — локальный порт TCP- и UDP-пакетов или тип для ICMP-пакетов;
- **Destination IP** — IP-адрес получателя пакета;
- **Port** — порт удаленного компьютера для TCP- и UDP-пакетов или код для ICMP-пакетов.
- **Size** — размер пакета (или суммарный размер пакетов, если количество пакетов больше единицы);
- **Cnt** — количество пакетов, относящихся к данному событию;
- **SourceNode** — узел отправителя;
- **DestinNode** — узел получателя;
- **Event** — код и название события, присвоенного пакету.

Просмотр журнала транспортных конвертов MFTP

В процессе работы транспортный модуль MFTP записывает информацию о полностью принятых, отправленных, удаленных и поврежденных конвертах в специальный журнал.

Алгоритм работы с журналом конвертов основан на использовании ротации. В настройках транспортного модуля задается максимальный размер файла журнала в мегабайтах. При превышении данного размера наиболее ранние записи перезаписываются новыми. Таким образом, задавая максимальный размер журнала конвертов, вы можете регулировать число хранимых записей на текущий момент. При изменении максимального размера журнала в процессе работы происходит реконструкция журнала конвертов — в случае уменьшения размера журнала наиболее ранние записи удаляются из него.

Для просмотра журнала транспортных конвертов выполните команду:

```
hostname# mftp view
```

При экспорте информации из журнала конвертов в текстовый файл выводится следующая информация:

- имя конверта;
- размер конверта;
- имя узла-отправителя;
- имя узла-получателя;
- название события;
- имя прикладной задачи (роли) или тип узла-отправителя;
- описание конверта;
- дата и время события.

Работа с журналом устранения неполадок

В ViPNet Coordinator HW демоны iplircfg, mftpd, failoverd записывают информацию о событиях, происходящих в процессе их работы, в журнал устранения неполадок. Этот журнал позволяет контролировать работоспособность ViPNet Coordinator HW, своевременно выявлять и устранять причины сбоев в его работе.

Запись информации в журнал устранения неполадок осуществляется системными средствами с использованием протокола syslog. Этот журнал может храниться в системном журнале syslog на самом сервере ViPNet Coordinator HW (локальное протоколирование) или на удаленном сетевом узле (удаленное протоколирование).

Внимание! Во всех исполнениях ViPNet Coordinator HW по умолчанию настроено локальное протоколирование. При этом по умолчанию в исполнениях с одним дисковым накопителем (ViPNet Coordinator HW50 и ViPNet Coordinator HW100 на аппаратных платформах HW100 X1, X8) в журнал записывается информация первого уровня детализации, во всех остальных исполнениях в журнал записывается информация третьего уровня детализации. Подробнее об уровнях детализации см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам», в описании секций `[debug]` конфигурационных файлов демонов iplircfg, mftpd, failoverd.

Во избежание заполнения дискового пространства на аппаратных платформах HW100 X1 рекомендуется включать ежедневную перезагрузку ViPNet Coordinator HW с помощью команды:

```
hostname# machine set dailyreboot mode on
```

В случае локального хранения журнала предусмотрена автоматическая ротация файла журнала с целью экономии места на ViPNet Coordinator HW. Ротация выполняется, когда размер файла журнала достигает следующих размеров:

- 50 Мбайт — для всех исполнений с одним дисковым накопителем;
- 1 Гбайт — для остальных исполнений.

В процессе ротации заполненный файл журнала переименовывается, при этом другой файл журнала, который был переименован раньше остальных, удаляется. Это делается для того, чтобы на ViPNet Coordinator HW всегда было не более:

- 3 заполненных файлов журнала и 1 текущего файла, в который ведется запись событий, — для всех исполнений с одним дисковым накопителем;
- 10 заполненных файлов и 1 текущего файла — для остальных исполнений.

При локальном хранении журнала вы можете выполнить с ним следующие действия:

- Просмотреть содержимое журнала (см. «Просмотр журнала устранения неполадок» на стр. 222).
- Экспортировать файлы журнала на другой компьютер (см. «Экспорт на компьютер» на стр. 225) или на USB-носитель (см. «Экспорт журнала устранения неполадок на USB-носитель» на стр. 226). Файлы журнала экспортируются в виде архива `logs.tar.gz`.
- Вручную удалить все имеющиеся файлы журнала. Для этого выполните команду:

  ```
  hostname# admin remove logs
  ```

Внимание! Удаление файлов журнала требуется при завершении эксплуатации ViPNet Coordinator HW. В других случаях удалять файлы журнала настоятельно не рекомендуется.

Вы также можете настроить ряд параметров ведения журнала. Подробнее см. раздел Настройка параметров ведения журнала устранения неполадок (на стр. 223).

Перечень протоколируемых событий, связанных с аутентификацией и настройкой оборудования, приведен в приложении (см. «События журнала устранения неполадок, связанные с аутентификацией и настройкой оборудования» на стр. 246).

Просмотр журнала устранения неполадок

Чтобы просмотреть записи журнала устранения неполадок, выполните команду:

```
hostname# machine show logs [reversed] [since <время>] [filtered {<демон> | string <строка>}],
```

указав следующие параметры:

- `reversed` — вывод списка записей в обратном хронологическом порядке.
- `<время>` — вывод списка записей, начиная с указанного момента времени. Момент времени задается в формате `YYYY-MM-DD hh:mm:ss`.
- `<демон>` — вывод записей только для указанного демона в составе ПО ViPNet Coordinator HW (см. «Список демонов в составе ПО ViPNet Coordinator HW» на стр. 255).
- `<строка>` — вывод записей журнала по части строки. При поиске по части строки можно использовать символы `A-Z, a-z, 0-9`, а также следующие символы:

  ```
  !# $ % & ( ) * + , - . / : ; < = > @ [ ] _ { | } ~
  ```

  Также можно использовать пробел, в этом случае необходимо взять часть строки в двойные кавычки (`""`).

Внимание! В одной команде `machine show logs` можно одновременно использовать только одну из лексем: `since` или `filtered`. При использовании лексемы `filtered`

можно указать только один из ее параметров: <демон> или <строка>.

Например, чтобы найти все записи журнала устранения неполадок, начиная с 14:50 22 ноября 2016 года, и отобразить их в обратном порядке, выполните команду:

```
hostname# machine show logs reversed since 2016-11-22 14:50:00
```

Настройка параметров ведения журнала устранения неполадок

Вы можете выполнить следующие настройки параметров ведения журнала устранения неполадок:

- Чтобы выбрать место хранения журнала устранения неполадок, выполните команду:

  ```
  hostname# machine set loghost {<IP-адрес> | local},
  ```

  указав IP-адрес удаленного сетевого узла, на котором будет храниться журнал, или параметр `local` для локального протоколирования.

  Внимание! Если для удаленного протоколирования будет использоваться открытый узел, на ViPNet Coordinator HW вручную задайте фильтр открытой сети, разрешающий исходящий трафик по протоколу UDP на 514-й порт этого открытого узла.

  

  Не рекомендуется использовать удаленное протоколирование на ViPNet Coordinator HW в режиме кластера горячего резервирования, так как на удаленный сетевой узел не будут передаваться журналы с пассивного сервера (то есть часть информации о работе ViPNet Coordinator HW будет утеряна).

  Если все же необходимо настроить удаленное протоколирование при работе в режиме кластера горячего резервирования, задайте параметры протоколирования на обоих серверах кластера, так как эти параметры не передаются с активного сервера на пассивный по каналу резервирования.

- Чтобы отключить ведение журнала устранения неполадок, выполните команду:

  ```
  hostname# machine set loghost null
  ```

  

  Внимание! Отключать ведение журнала не рекомендуется.

- Чтобы изменить уровень детализации информации, выводимой в журнал, выполните настройку параметра `debuglevel` в секции `[debug]` конфигурационных файлов `iplir.conf`, `failover.ini`, `mftp.conf`. Описание параметров секции `[debug]` в данных файлах см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам».

Совет. В большинстве случаев изменять уровень детализации, выбранный по умолчанию, не рекомендуется. Особенно не рекомендуется повышать уровень детализации в исполнениях с одним дисковым накопителем (ViPNet Coordinator HW50 и ViPNet Coordinator HW100 на аппаратных платформах HW100 X1, X8). При повышении уровня детализации необходимо учитывать, что размер файла журнала будет быстро расти, вследствие чего, будет чаще производиться его ротация при достижении максимального размера (см. «Работа с журналом устранения неполадок» на стр. 221).

- Чтобы сообщения о работе демона выводились на консоль командного интерпретатора, выполните команду:

  `hostname# debug on <facility.level>`,

  указав выбранные значения параметров `facility` и `level`.

  Параметры `facility` и `level` должны быть заданы в секции `[debug]` конфигурационных файлов `iplir.conf`, `failover.ini`, `mftp.conf`. Описание параметров секции `[debug]` в данных файлах см. в документе «ViPNet Coordinator HW. Справочное руководство по конфигурационным файлам».

- Чтобы выключить вывод сообщений на консоль командного интерпретатора, выполните команду `debug off`. С помощью этой команды вы можете выключить как вывод всех сообщений, так и вывод только тех сообщений, которые соответствуют конкретным значениям параметров `facility` и `level`.

  Например, вы можете настроить протоколирование так, чтобы следить за работой разных демонов на разных терминалах. Для этого выполните следующее:

  8.1 В конфигурационном файле одного из демонов в секции `[debug]` задайте нужные параметры, например:

  ```
  [debug]
  debuglogfile = syslog:daemon.debug
  debuglevel = 3
  ```

  И на одном из терминалов выполнить команду:

  ```
  hostname# debug on daemon.debug
  ```

  8.2 В конфигурационном файле другого демона в секции `[debug]` задайте параметры протоколирования, указав другое значение `facility`, например:

  ```
  [debug]
  debuglogfile = syslog:local0.debug
  debuglevel = 3
  ```

  И на другом терминале выполнить команду:

  ```
  hostname# debug on local0.debug
  ```

  В результате на первый терминал будут поступать сообщения только от первого демона, на второй терминал — только от второго демона. Чтобы на первом терминале можно было следить за работой обоих демонов, на нем в дополнение к первой команде выполните команду:

  ```
  hostname# debug on local0.debug
  ```

Примечание. Выводимые на консоль командного интерпретатора сообщения о работе демонов могут смешиваться с сообщениями самого интерпретатора и командами, набираемыми на консоли. Для просмотра набранной к данному моменту части команды и продолжения ее ввода используйте комбинацию клавиш **Ctrl+L**.

Экспорт на компьютер

Внимание! Во время экспорта журналов на другой компьютер завершается работа всех демонов и выгружаются все драйверы ViPNet, то есть ViPNet Coordinator HW будет незащищенным.

При экспорте журналов устранения неполадок с ViPNet Coordinator HW на другой компьютер используется стандартная служба TFTP. В ОС Windows XP эта служба по умолчанию включена. В ОС Windows Vista эта служба по умолчанию выключена и ее необходимо включить вручную. Для включения службы в ОС Windows Vista выполните следующее:

1. Выберите **Пуск** (**Start**) > **Панель управления** (**Control Panel**) > **Программы и компоненты** (**Programs and Features**).
2. Зайдите в меню **Включение или отключение компонентов Windows** (**Turn Windows features on or off**) и установите флажки рядом с названием служб **Клиент TFTP** (**TFTP Client**) и **Простые службы TCPIP** (**Simple TCPIP services**).

На время выполнения экспорта автоматически изменяются настройки сетевых интерфейсов ViPNet Coordinator HW: на интерфейсе Ethernet1 устанавливается IP-адрес `169.254.241.1`, все остальные интерфейсы выключаются. После успешного завершения экспорта настройки интерфейсов ViPNet Coordinator HW автоматически восстанавливаются.

Внимание! Во избежание потери удаленного доступа к ViPNet Coordinator HW запрещается выполнять экспорт журналов на другой компьютер в удаленной SSH-сессии.

Чтобы экспортировать журналы устранения неполадок на компьютер, выполните следующие действия:

1. Подключите ноутбук к порту Ethernet1 ViPNet Coordinator HW с помощью кросс-кабеля Ethernet.
2. Установите вручную на сетевом интерфейсе ноутбука IP-адрес 169.254.241.5.
3. Подключите к ViPNet Coordinator HW обычную консоль (монитор и клавиатуру) или COM-консоль.
4. Завершите работу демонов iplircfg, failoverd и mftpd.

5 Выполните команду:

```
hostname# admin export logs tftp
```

При выполнении команды производится архивирование журналов устранения неполадок. Чтобы прервать экспорт во время архивирования, нажмите сочетание клавиш **Ctrl+C**.

После завершения архивирования появится сообщение, содержащее имя архива и предложение скачать его.

6 Перенесите файл с архивом на ноутбук по TFTP. Для этого выполните на ноутбуке команду:

```
tftp –i 169.254.241.1 get <имя файла>
```

7 Нажмите клавишу **Enter**.

8 Запустите демоны iplircfg, failoverd и mftpd.

В результате архив будет скопирован на компьютер.

Экспорт журнала устранения неполадок на USB-носитель

Чтобы экспортировать журналы устранения неполадок на USB-носитель, выполните следующие действия:

1 На ViPNet Coordinator HW выполните команду:

```
hostname# admin export logs usb
```

При выполнении команды производится архивирование журналов устранения неполадок. Чтобы прервать экспорт во время архивирования, нажмите сочетание клавиш **Ctrl+C**.

После завершения архивирования появляется предложение подключить USB-носитель и подтвердить выполнение экспорта.

2 Подключите USB-носитель к USB-разъему ViPNet Coordinator HW (или к компьютеру, на котором развернут виртуальный образ ViPNet Coordinator HW).

В результате архив будет скопирован на USB-носитель.

15

Мониторинг ViPNet Coordinator HW

Мониторинг с помощью ПК ViPNet StateWatcher

Вы можете осуществлять мониторинг работы ViPNet Coordinator HW с помощью программного комплекса ViPNet StateWatcher. Данный комплекс предназначен для наблюдения за состоянием узлов сети ViPNet (в том числе ViPNet Coordinator HW). ПК ViPNet StateWatcher также собирает информацию о состоянии установленных на узлах компонентов ПО ViPNet, о текущей загруженности аппаратного обеспечения узлов, о состоянии сетевых интерфейсов, о статусе связей между узлами.

В случае возникновения сбоев или неполадок на узлах, вы будете оперативно проинформированы об этом. Вы можете настроить удобный вам способ оповещения о событиях: непосредственно в веб-интерфейсе ПК ViPNet StateWatcher (визуальные и звуковые оповещения, оповещения на карте и информационной панели), по электронной почте (в том числе ViPNet Деловая почта), по SMS, по протоколу Syslog. События на узлах выявляются на основе анализа их состояния специальными правилами. Вы можете использовать встроенные правила анализа или создать собственные правила, с помощью которых будут анализироваться важные для вас параметры узлов.

Подробнее о работе с ПК ViPNet StateWatcher см. в документе «Программный комплекс ViPNet StateWatcher. Сервер мониторинга. Руководство администратора».

Мониторинг по протоколу SNMP

С помощью SNMP-агента, входящего в состав ViPNet Coordinator HW, вы можете удаленно получать информацию о времени работы системы, активности сетевых интерфейсов и количестве пропущенных и блокированных пакетов на них, а также уведомлять удаленную станцию управления сетью (NMS) о наиболее важных событиях в системе (например, выходе из строя активного сервера при работе ViPNet Coordinator HW в режиме кластера горячего резервирования).

Примечание. ViPNet Coordinator HW поддерживает протокол SNMP версий 1 и 2. Параметры протокола `rocommunity` и `trapcommunity` по умолчанию имеют значение `public`.

При работе ViPNet Coordinator HW в режиме кластера горячего резервирования на удаленную станцию передается информация только о работе активного сервера кластера. Слежение за работой пассивного сервера кластера не производится.

Для получения информации о сетевых узлах ViPNet по протоколу SNMP необходимо выделить компьютер и установить на него специальное программное обеспечение управления сетью (NMS), например WhatsUp Gold. Чтобы получение информации было возможно, импортируйте файл `VIPNET-MIB.txt`, в котором описаны используемые ПО ViPNet объекты SNMP, в NMS (подробнее см. в документации по используемой NMS).

В результате на компьютер будет приходить информация о состоянии ViPNet Coordinator HW. Для получения этой информации должна использоваться ветка:

```
.iso.org.dod.internet.private.enterprises.infotecs.vipnet (.1.3.6.1.4.1.10812.1)
```

Опрос ViPNet Coordinator HW по протоколу SNMP рекомендуется выполнять с защищенных узлов ViPNet. Соответствующие сетевые фильтры по умолчанию включены в список фильтров защищенной сети. Если опрос будет осуществляться с открытого узла, то в список локальных фильтров открытой сети необходимо добавить аналогичные разрешающие фильтры, указав в них IP-адрес этого узла.

Примечание. Чтобы SNMP-запросы обрабатывались корректно, необходимо отправлять их на IP-адрес ViPNet Coordinator HW, который доступен с опрашивающего узла по кратчайшему маршруту.

ViPNet Coordinator HW поддерживает базу управляющей информации (MIB) для ПО ViPNet (см. «Описание SNMP-параметров для ПО ViPNet Coordinator HW» на стр. 231) и некоторые стандартные базы управляющей информации (см. «Поддерживаемые базы управляющей информации SNMP» на стр. 233).

По умолчанию SNMP-агент выключен и не запускается автоматически при загрузке ViPNet Coordinator HW. При необходимости вы можете изменить параметры запуска SNMP-агента:

- Чтобы включить автоматический запуск SNMP-агента, выполните команду:

```
hostname# inet snmp mode {on | off}
```

- Чтобы запустить или остановить SNMP-агент, выполните команду:

```
hostname# inet snmp {start | stop}
```

Чтобы использовать оповещения (SNMP Traps), выполните настройку следующих параметров:

- Настройте IP-адрес удаленного узла, на который будут отправляться асинхронные оповещения о различных событиях при работе ПО ViPNet, с помощью команды:

```
hostname# inet snmp trapsink <IP-адрес>
```

Примечание. Выключить использование оповещений SNMP Traps вы можете с помощью команды:

```
hostname# inet snmp trapsink none
```

- Чтобы на удаленной станции управления сетью при приеме оповещений SNMP Traps проводилась авторизация, задайте пароль с помощью команды:

```
hostname# inet snmp community trap <пароль>
```

- При необходимости задайте пароль доступа к SNMP-параметрам вашего ViPNet Coordinator HW с помощью команды:

```
hostname# inet snmp community ro <пароль>
```

Примечание. Пароли должны содержать от 6 до 18 символов. В паролях вы можете использовать символы латинского алфавита, цифры, а также следующие специальные символы: «.», «*», «/», «-», «:», «_», «=», «@», «&».

Просмотреть текущие значения паролей вы можете с помощью команды:

```
hostname> inet show snmp community
```

Информация о работе SNMP-агента записывается в системный журнал. Чтобы настроить уровень детализации записываемой информации, выполните команду:

```
hostname# inet snmp debug-level <уровень детализации>
```

В качестве уровня детализации вы можете задать следующие значения:

- `info` (по умолчанию) — записывается только информация, касающаяся инициализации SNMP-агента. Этот уровень задан .
- `debug` — записывается служебная информация, используемая при отладке.
- `error` — записываются ошибки, после которых SNMP-агент может продолжать работу.
- `critical` — записываются критические ошибки, после которых SNMP-агент не может продолжить работу.
- `off` — запись выключена.

Чтобы получить информацию о текущем состоянии SNMP-агента, IP-адресе удаленного узла, на который отправляются оповещения (SNMP Traps), а также о заданном уровне детализации, выполните команду:

```
hostname> inet show snmp
```

Описание SNMP-параметров для ПО ViPNet Coordinator HW

ПО ViPNet использует следующие объекты, для которых возможно только чтение данных:

- .1.3.6.1.4.1.10812.1.1.1 — шестнадцатеричный идентификатор сети ViPNet.
- .1.3.6.1.4.1.10812.1.1.2 — шестнадцатеричный идентификатор сетевого узла ViPNet.
- .1.3.6.1.4.1.10812.1.1.3 — имя пользователя узла ViPNet.
- .1.3.6.1.4.1.10812.1.1.4 — время работы ViPNet Coordinator HW (обнуляется каждый раз при перезапуске управляющего демона).
- .1.3.6.1.4.1.10812.1.1.5 — шестнадцатеричный идентификатор, включающий в себя идентификаторы узла ViPNet и сети ViPNet.
- .1.3.6.1.4.1.10812.1.1.7 — имя узла ViPNet.
- .1.3.6.1.4.1.10812.1.2.1 — количество сетевых интерфейсов.
- .1.3.6.1.4.1.10812.1.2.2 — последовательность объектов, описывающих сетевые интерфейсы.
- .1.3.6.1.4.1.10812.1.2.2.1 — объекты, описывающие сетевые интерфейсы:
  - .1.3.6.1.4.1.10812.1.2.2.1.1 — номер интерфейса;
  - .1.3.6.1.4.1.10812.1.2.2.1.2 — системное имя интерфейса;
  - .1.3.6.1.4.1.10812.1.2.2.1.3 — устаревший параметр, всегда равен 0;
  - .1.3.6.1.4.1.10812.1.2.2.1.4 — число пропущенных входящих нешифрованных пакетов;
  - .1.3.6.1.4.1.10812.1.2.2.1.5 — число блокированных входящих нешифрованных пакетов;
  - .1.3.6.1.4.1.10812.1.2.2.1.6 — число пропущенных исходящих нешифрованных пакетов;
  - .1.3.6.1.4.1.10812.1.2.2.1.7 — число блокированных исходящих нешифрованных пакетов;
  - .1.3.6.1.4.1.10812.1.2.2.1.8 — число пропущенных входящих шифрованных пакетов;
  - .1.3.6.1.4.1.10812.1.2.2.1.9 — число блокированных входящих шифрованных пакетов;
  - .1.3.6.1.4.1.10812.1.2.2.1.10 — число пропущенных исходящих шифрованных пакетов;
  - .1.3.6.1.4.1.10812.1.2.2.1.11 — число блокированных исходящих шифрованных пакетов.
- .1.3.6.1.4.1.10812.1.7 — последовательность объектов, описывающих состояние демонов iplircfg, failoverd, mftpd, algd, webgui-fcgi-server (подробнее о демонах ViPNet Coordinator HW см. «ViPNet Coordinator HW. Общее описание», раздел «Состав ПО ViPNet Coordinator HW»).
- .1.3.6.1.4.1.10812.1.7.1 — объекты, описывающие состояние демонов ViPNet Coordinator HW:
  - .1.3.6.1.4.1.10812.1.7.1.1 — индекс демона;
  - .1.3.6.1.4.1.10812.1.7.1.2 — имя демона:

        - `Alg` — демон algd,
        - `Failover` — демон failoverd,
        - `Iplir` — демон iplircfg,
        - `Mftp` — демон mftpd,
        - `Webgui` — демон webgui-fcgi-server;
    - .1.3.6.1.4.1.10812.1.7.1.3 — состояние демона:
        - `unknown(0)` — состояние неизвестно,
        - `shutdown(1)` — работа демона завершена,
        - `running(2)` — демон работает;
    - .1.3.6.1.4.1.10812.1.7.1.4 — время работы демона.
- .1.3.6.1.4.1.10812.4.3.1 — версия агента мониторинга.
- .1.3.6.1.4.1.10812.4.3.2 — уникальный идентификатор установки агента мониторинга.
- .1.3.6.1.4.1.10812.4.4 — таблица плагинов агента мониторинга.
- .1.3.6.1.4.1.10812.4.5 — таблица функциональных возможностей агента мониторинга.
- .1.3.6.1.4.1.10812.4.20.5 — уникальный идентификатор устройства.

ПО ViPNet использует следующие оповещения SNMP (SNMP Traps):

- .1.3.6.1.4.1.10812.1.0.1 — запуск управляющего демона.
- .1.3.6.1.4.1.10812.1.0.2 — завершение работы управляющего демона.
- .1.3.6.1.4.1.10812.1.0.3 — запуск демона mftpd.
- .1.3.6.1.4.1.10812.1.0.4 — завершение работы демона mftpd.
- .1.3.6.1.4.1.10812.1.0.7 — запуск демона failoverd (только в активном режиме).
- .1.3.6.1.4.1.10812.1.0.8 — завершение работы демона failoverd (только в активном режиме).

Примечание. Как правило, NMS позволяет настроить ответное действие при возникновении каждого из этих оповещений (например, отправка письма). Подробнее см. в документации по используемой NMS.

Поддерживаемые базы управляющей информации SNMP

ViPNet Coordinator HW частично поддерживает следующие стандартные базы управляющей информации (MIB):

- MIB-II System (.1.3.6.1.2.1.1) — база управляющей информации для системных параметров. Включает в себя объекты, предназначенные для получения информации о различных системных параметрах компьютера и операционной системы ViPNet Coordinator HW (например, имя сетевого узла, название ОС, название аппаратной платформы, количество и параметры сетевых интерфейсов). Подробнее см. в RFC 1213 (https://tools.ietf.org/html/rfc1213).
- IF-MIB (.1.3.6.1.2.1.2) — база управляющей информации для сетевых интерфейсов ViPNet Coordinator HW. Включает в себя объекты, предназначенные для получения информации о различных параметрах сетевых интерфейсов ViPNet Coordinator HW (например, тип интерфейса, физический адрес, скорость). Подробнее см. в RFC 2863 (https://tools.ietf.org/html/rfc2863).
- IP-MIB (.1.3.6.1.2.1.4) — база управляющей информации для протокола IP. Включает в себя объекты, предназначенные для получения информации о различных характеристиках работы ViPNet Coordinator HW по протоколу IP (например, параметры сетевых интерфейсов, статистика для IP-трафика, таблица адресов). Подробнее см. в RFC 4293 (https://tools.ietf.org/html/rfc4293).
- TCP-MIB (.1.3.6.1.2.1.6) — база управляющей информации для протокола TCP. Включает в себя объекты, предназначенные для получения информации о различных характеристиках работы ViPNet Coordinator HW по протоколу TCP (например, максимальный размер сегмента, количество активных соединений). Подробнее см. в RFC 4022 (https://tools.ietf.org/html/rfc4022).
- UDP-MIB (.1.3.6.1.2.1.7) — база управляющей информации для протокола UDP. Включает в себя объекты, предназначенные для получения информации о различных характеристиках работы ViPNet Coordinator HW по протоколу UDP (например, общее количество полученных и отправленных датаграмм). Подробнее см. в RFC 4113 (https://tools.ietf.org/html/rfc4113).
- UCD-SNMP-MIB (.1.3.6.1.4.1.2021) — база управляющей информации для протокола SNMP. Включает в себя объекты, предназначенные для получения информации о состоянии компьютера и операционной системы (например, состояние оперативной памяти и жесткого диска). Подробнее см. на веб-ресурсе http://www.net-snmp.org/docs/mibs/UCD-SNMP-MIB.txt.
- SNMP-FRAMEWORK-MIB (.1.3.6.1.6.3.10) — база управляющей информации, которая включает в себя параметры для получения информации о динамически загружаемых модулях SNMP в ViPNet Coordinator HW. Подробнее см. на веб-ресурсе http://www.net-snmp.org/docs/mibs/snmpFrameworkMIB.html.

A

Сетевые фильтры по умолчанию

Сетевые фильтры, в том числе фильтры системы защиты от сбоев, создаваемые в ViPNet Coordinator HW по умолчанию, перечислены в таблицах ниже.

Таблица 12. Фильтры защищенной сети (vpn)

Название	Источник	Назначение	Протокол	Действие
В активном режиме кластера (нередактируемый)				
ViPNet Service	@any	@any	udp: from 2060 to 2060	pass
В пассивном режиме кластера (нередактируемый)				
Block all vpn traffic	@any	@any	@any	drop
Общие фильтры				
Allow DHCP service	@any	@any	udp: from 67 to 68, from 68 to 67	pass
Allow DHCP-Relay service	@any	@any	udp: from 67 to 67	pass
Allow ViPNet base services	@any	@any	udp: to 2046, from 2048 to 2048, from 2050 to 2050	pass
Allow ViPNet DBViewer	@any	@any	tcp: to 2047	pass
Allow ViPNet StateWatcher	@any	@local	tcp: to 5100, 10092	pass
Allow ViPNet MFTP	@any	@any	tcp: to 5000-5003	pass

Название	Источник	Назначение	Протокол	Действие
Allow ViPNet WebGUI	@any	@local	tcp: to 8080	pass
Allow ViPNet SGA	@any	@local	tcp: to 10095, tcp: to 5103, tcp: to 10093	pass
Allow ICMP Ping	@any	@any	icmp8	pass
Allow SSH	@any	@any	tcp: to 22	pass
Allow DNS	@any	@any	udp: to 53	pass
Allow NTP	@any	@any	udp: to 123	pass
Allow UPS service	@any	@any	tcp: to 3493	pass
Примечание. В исполнении ViPNet Coordinator HW VA данный фильтр отсутствует.				
Allow syslog outgoing	@local	@any	udp: to 514	pass
Allow SNMP	@any	@local	udp: to 161	pass
Allow SNMP traps	@local	@any	udp: to 162	pass
Блокирующий фильтр (нередактируемый)				
Block All Traffic	@any	@any	@any	drop

Таблица 13. Локальные фильтры открытой сети (`local`)

Название	Источник	Назначение	Протокол	Действие
В одиночном режиме системы защиты от сбоев, в активном и пассивном режимах кластера (нередактируемые)				
ViPNet Service CommonIn	@any	@local	tcp/udp: to 2046, 2047, 10095, 10096, 5100, 5103, 10092, 10093	drop
ViPNet Service CommonOut	@local	@any	tcp/udp: from 2046	drop
В активном режиме кластера (нередактируемые)				
Failover test IP	@local	<список значений параметров `testip` из `failover.ini`>	icmp	pass
Failover Channel	@local	iface <значение параметра `device` из `failover.ini`>	@any	pass

Название	Источник	Назначение	Протокол	Действие
Failover Channel	<значение параметра `activeip` второго сервера из `failover.ini`>	iface <значение параметра `device` из `failover.ini`>	@any	pass
В пассивном режиме кластера (нередактируемые)				
Failover Channel	@local	iface <значение параметра `device` из `failover.ini`>	@any	pass
Failover Channel	<значение параметра `activeip` второго сервера из `failover.ini`>	iface <значение параметра `device` из `failover.ini`>	@any	pass
ARP requests to active	@local	<список значений параметров `activeip` из `failover.ini`>	udp: to <значение `connect_port` из `failover.ini`>	pass
Block all local traffic	@any	@any	@any	drop
Общие фильтры				
Allow DHCP service	@any	@any	udp: from 67 to 68, from 68 to 67	pass
Allow DHCP-Relay service	@any	@any	udp: from 67 to 67	pass
Allow ICMP Ping	@local	@any	icmp8	pass
Allow DNS	@local	@any	udp: to 53	pass
Allow NTP	@local	@any	udp: to 123	pass
Блокирующий фильтр (нередактируемый)				
Block All Traffic	@any	@any	@any	drop

Таблица 14. Транзитные фильтры открытой сети (`forward`)

Название	Источник	Назначение	Протокол	Действие
В пассивном режиме кластера				
Block all forward traffic	@any	@any	@any	drop
Блокирующий фильтр (нередактируемый)				

Название	Источник	Назначение	Протокол	Действие
Block All Traffic	@any	@any	@any	drop

Таблица 15. Фильтры туннелируемых узлов (`tunnel`)

Название	Источник	Назначение	Протокол	Действие
В пассивном режиме кластера (нередактируемый)				
Block all tunnel traffic	@any	@any	@any	drop
Общий фильтр				
To all tunnel nodes	@any	@tunneledip	@any	pass
From all tunnel nodes	@tunneledip	@any	@any	pass
Блокирующий фильтр (нередактируемый)				
Block All Traffic	@any	@any	@any	drop

B

Пользовательские группы протоколов по умолчанию

По умолчанию в ViPNet Coordinator HW настроены пользовательские группы протоколов, перечисленные в таблице ниже.

Таблица 16. Пользовательские группы протоколов, настроенные по умолчанию

Имя группы протоколов	Состав группы протоколов
DHCP	UDP:from 67-68 to 67-68
CITRIX	TCP:to 1494
DNS	UDP:to 53
FTP	TCP:to 21
GRE	IP:47
H323	TCP:to 1720
HTTP	TCP:to 80, TCP:to 8080
HTTP-Proxy	TCP:to 3128
HTTPS	TCP:to 443
IGMP	IP:2
IKE	UDP:to 500
IMAP	TCP:to 143
IPSecESP	IP:50

Имя группы протоколов	Состав группы протоколов
Kerberos	TCP:to 88, TCP:to 749, UDP:to 88, UDP:to 749
L2TP	UDP:to 1701
LDAP	TCP:to 389, UDP:to 389
LotusNotes	TCP:to 1352
MS-SQL	TCP:to 1433-1434, UDP:to 1433-1434
MySQL	TCP:to 3306
NetBIOS-DGM	UDP:from 138 to 138
NetBIOS-NC	UDP:from 137 to 137
NetMeeting	TCP:to 1503
NTP	UDP:to 123
PING	ICMP:8
POP3	TCP:to 110
Postgres	TCP:to 5432
PPTP	TCP:to 1723
RADIUS	UDP:to 1812-1813
RDP	TCP:to 3389
RTSP	TCP:to 554
SCCP	TCP:to 2000
SIP	TCP:to 5060, UDP:to 5060
SMTP	TCP:to 25
SNMP	UDP:to 161
SNMP-Traps	UDP:to 162
SSH	TCP:to 22
Syslog	UDP:to 514
Telnet	TCP:to 23
TFTP	UDP:to 69
UPnP	TCP:to 1900, TCP:to 2869, UDP:to 1900, UDP:to 2869
MFTP	TCP:to 5000-5003
StateWatcher	TCP:to 2047, TCP:to 5100, TCP:to 10092
ViPNetBase	UDP:to 2046, UDP:from 2048 to 2048, UDP:from 2050 to 2050
Cluster	UDP:from 2060 to 2060

Имя группы протоколов	Состав группы протоколов
ClusterMonitoring	UDP:from 2060 to 2065, UDP:from 2065 to 2060
SGA	TCP:to 5103, TCP:to 10093, TCP:to 10095
WindowsMobileDevices	TCP:to 990, TCP:to 999, TCP:to 5678, TCP:to 5721, TCP:to 26675
WindowsMobileDevices2	UDP:to 5679
VNC	TCP:to 5900
OSPF	IP:89

C

Типы событий в журнале регистрации IP-пакетов

Типы событий, регистрируемых в журнале IP-пакетов ViPNet Coordinator HW, можно поделить на следующие группы и подгруппы:

Рисунок 33. Классификация событий в журнале IP-пакетов

Описание типов событий каждой подгруппы приведено в таблицах.

*Таблица 17. События подгруппы **Все IP-пакеты/Блокированные IP-пакеты/IP-пакеты, блокированные сетевыми фильтрами защищенной сети***

Номер события	Описание события
1	Не найден ключ для передачи пакета сетевому узлу с идентификатором, указанным в пакете
2	Неверное значение имитозащитной вставки пакета. Защищенные данные или открытая информация в пакете были изменены при передаче
3	Входящий зашифрованный или предназначенный для шифрования исходящий открытый пакет заблокирован фильтром защищенной сети
4	Слишком большой тайм-аут пакета, то есть время его отправки отличается от времени его получения на величину, большую указанной в соответствующей настройке
5	Входящий пакет отправлен сетевым узлом с версией драйвера ViPNet, не совместимой с версией драйвера ViPNet на сетевом узле получателя
7	Метод шифрования, код которого указан во входящем пакете, не поддерживается
8	Недопустимые параметры в расшифрованном пакете
9	IP-пакет блокирован главным фильтром защищенной сети
10	Неизвестен идентификатор сетевого узла отправителя пакета
13	Превышено максимальное время пребывания пакета в сети
14	Неверный адрес получателя пакета
15	Превышено допустимое количество одновременно обрабатываемых фрагментированных пакетов
16	Исчерпана лицензия на количество туннелируемых узлов
17	Неверный IP-адрес получателя
18	Неизвестный IP-адрес получателя
19	Подмена узла отправителя
70	Транзитный IP-пакет заблокирован фильтром защищенной сети

*Таблица 18. События подгруппы **Все IP-пакеты/Блокированные IP-пакеты/IP-пакеты, блокированные сетевыми фильтрами открытой сети***

Номер события	Описание события
20	Заблокирован открытый пакет
21	Идентификатор отправителя пакета неизвестен
22	От защищенного узла получен открытый пакет
23	От защищенного узла получен открытый широковещательный пакет

Номер события	Описание события
24	На порт, используемый одним из демонов ViPNet, получен открытый пакет
30	Локальный пакет заблокирован фильтром открытой сети
31	Транзитный пакет заблокирован фильтром открытой сети
32	Широковещательный пакет заблокирован фильтром открытой сети
33	Пакет заблокирован фильтром антиспуфинга
37	Пакет заблокирован фильтром туннелируемых ресурсов
39	Пакет заблокирован фильтром по умолчанию при загрузке компьютера

*Таблица 19. События подгруппы **Все IP-пакеты/Блокированные IP-пакеты/IP-пакеты, блокированные по другим причинам***

Номер события	Описание события
80	Размер заголовка IP-пакета меньше допустимого
81	Недопустимая версия протокола IP (поддерживается только IPv4)
82	Длина заголовка IP-пакет меньше допустимой
83	Длина IP-пакета меньше указанной в IP-заголовке
84	Значение контрольной суммы в заголовке IP-пакета отличается от ее значения в IP-пакете
85	Размер TCP-заголовка меньше допустимого
86	Размер UDP-заголовка меньше допустимого
87	Обработаны не все фрагменты IP-пакета
88	Широковещательный адрес отправителя в IP-пакете
89	Перекрытие фрагментов IP-пакета. Наиболее устаревший из перекрываемых фрагментов IP-пакета отброшен
90	IP-пакет не был обработан, так как для его обработки недостаточно вычислительных ресурсов
91	IP-пакет получен во время инициализации драйвера ViPNet
92	Размер IP-пакета превышает 48 Кбайт
93	По истечении допустимого времени получены не все фрагменты IP-пакета
95	Получены два IP-пакета от узлов с разными IP-адресами и одинаковыми идентификаторами
99	Заблокирован фрагментированный IP-пакет

Номер события	Описание события
101	Транзитный IP-пакет не может быть маршрутизирован
102	Прикладной пакет не обработан, так как не загружен модуль обработки на прикладном уровне
103	Количество установленных соединений превышает допустимое значение, заданное в соответствующих настройках
104	Соединение заблокировано, так как параметры исходящих пакетов (socketpair) для этого соединения совпадают с такими параметрами для ранее установленного соединения
105	Не удалось выделить динамический порт для правила трансляции адресов (в пуле нет свободных портов)
111	Не найден ключ обмена
112	Неверное значение имитозащитной вставки в незашифрованном пакете версии 4.2
113	Неизвестный идентификатор сетевого узла отправителя
115	Не удалось найти маршрут для IP-пакета в таблице маршрутизации
116	Не найден сетевой адаптер
117	Не удалось определить MAC-адрес получателя пакета по его IP-адресу
118	Ошибка при шифровании исходящего IP-пакета для защищенного узла
119	Получен зашифрованный IP-пакет неизвестного формата, который не может быть расшифрован
120	Ошибка при отправке IP-пакета защищенному узлу из-за проблем с доступом к этому узлу
121	Ошибка в работе кластера горячего резервирования
122	Получен IP-пакет неизвестного протокола канального уровня
130	Отсутствуют свободные динамические порты, необходимые для создания соединения
131	Ошибка обработки прикладных протоколов: не удалось построить маршрут
132	Ошибка обработки прикладных протоколов: отсутствуют свободные динамические порты UDP
133	Ошибка обработки прикладных протоколов: порты источника и назначения IP-пакета принадлежат разным прикладным сервисам
134	Ошибка обработки прикладных протоколов: ошибка в таблице соответствия между прикладными сервисами и сетевыми протоколами, портами
137	Шифрование исходящего IP-пакета с использованием данного алгоритма запрещено (тесты алгоритма не прошли)

*Таблица 20. События группы **Все IP-пакеты/Все пропущенные IP-пакеты/Пропущенные зашифрованные IP-пакеты***

Номер события	Описание события
40	Пропущен зашифрованный IP-пакет
41	Пропущен зашифрованный широковещательный IP-пакет
44	Маршрутизация зашифрованного транзитного IP-пакета с подменой адреса получателя
45	Пакет пропущен на туннелирующем координаторе, так как он поступил от туннелируемого узла или предназначен для такого узла

*Таблица 21. События группы **Все IP-пакеты/Все пропущенные IP-пакеты/Пропущенные незашифрованные IP-пакеты***

Номер события	Описание события
60	Пропущен незашифрованный IP-пакет
61	Пропущен незашифрованный широковещательный IP-пакет
62	Пропущен незашифрованный транзитный IP-пакет
63	IP-пакет пропущен фильтром для туннелируемых ресурсов
64	IP-пакет пропущен при запуске операционной системы
65	Пропущен зашифрованный IP-пакет для незарегистрированного узла ViPNet

*Таблица 22. События группы **Все IP-пакеты/Служебные события***

Номер события	Описание события
42	Изменился IP-адрес сетевого узла
46	Изменились параметры доступа к сетевому узлу
47	Истек интервал отправки IP-пакетов, передаваемых сетевым узлом своему серверу соединений. Событие может возникать только для тех узлов, которые работают через межсетевой экран с динамической трансляцией адресов
48	Узел доступен по широковещательному адресу
49	Изменился IP-адрес собственного сетевого узла
110	Новый IP-адрес сетевого узла зарегистрирован на DNS-сервере
114	DNS-имя узла не зарегистрировано на DNS-сервере

D

События журнала устранения неполадок, связанные с аутентификацией и настройкой оборудования

Ниже приведены протоколируемые события ViPNet Coordinator HW, связанные с аутентификацией и настройкой оборудования, и соответствующие им записи в журнале устранения неполадок.

Таблица 23. Протоколируемые события операционной системы Linux

Событие	Пример записей в журнале
Запуск операционной системы	`vmunix: Linux version 3.10.92`

Завершение работы операционной системы	`vmunix: [] drviplir: ViPNet IpLir driver unloaded`
	`vmunix: [] itcscrpt: ViPNet Crypto Driver unloaded`
	`vmunix: [] itcswd: ViPNet WatchDog driver unloaded`
	`vmunix: [] itcskrniface: Kernel interface driver is unloaded`
	`acpid: exiting`

Событие	Пример записей в журнале
	`syslogd: exiting on signal 15`

Таблица 24. События, протоколируемые при работе пользователя или администратора ViPNet Coordinator HW

Событие	Пример записей в журнале
Завершение работы операционной системы по команде	`rvpn_shell[1380]: <HALT> Command 'machine halt'.`
Перезагрузка операционной системы по команде	`rvpn_shell[1356]: <REBOOT> Command: machine reboot`
Выход из командного интерпретатора, работающего в режиме пользователя	`login[20446]: pam_unix(login:session): session closed for user user` `rvpn_shell[2190]: <EXIT> Command 'exit'`
Успешный удаленный запуск командного интерпретатора по протоколу SSH	`login[20446]: pam_unix(sshd:session): session opened for user user` `sshd[20448]: Accepted password for user from 13.0.1.94 port 55650 ssh2`
Не удалось запустить командный интерпретатор при удаленном подключении по протоколу SSH	`sshd[20448]: Failed password for user from 13.0.1.94 port 55650 ssh2`
Завершение удаленного подключения по протоколу SSH	`sshd[20448]: pam_unix(sshd:session): session closed for user user`
Успешный переход в режим администратора	`rvpn_shell[1334]: <ENABLE> Command 'enable' – Administrator has logged on.`
Не удалось перейти в режим администратора	`rvpn_shell[2190]: <ENABLE> Command 'enable' - Failed to log on as administrator.`
Переход из режима администратора в режим пользователя	`rvpn_shell[2190]: <EXIT> Command 'exit'`
Вход в системную командную оболочку	`rvpn_shell[2190]: <I_SHELL> You have exited to theLinux system shell.`
Выход из системной командной оболочки	Запись о событии не заносится в журнал.
Принудительное завершение сессии командного интерпретатора	`rvpn_shell[1324]: <A_KICK> Command: 'admin kick'.`

Событие	Пример записей в журнале

Успешное изменение файла `iplir.conf`	`rvpn_shell: <I_CFG> Command: iplir config - starting`
	`rvpn_shell: <I_CFG> Command: iplir config. Config file edited successfully.`
	Отображение изменения:
	`rvpn_shell: 15655c15655`
	`rvpn_shell: < visibility= auto`
	`rvpn_shell: ---`
	`rvpn_shell: > visibility= real`
	`rvpn_shell: <I_CFG> Command: iplir config - iplir.conf has been edited successfully.`

Не удалось изменить содержимое файла `iplir.conf`	`rvpn_shell: <I_CFG> Command: iplir config - starting`
	`rvpn_shell: <I_CFG> Command: iplir config. Incorrect config file.`
	или
	`rvpn_shell: <I_CFG> Command: iplir config - starting`
	`rvpn_shell: <I_CFG> Command: iplir config Cannot edit config while iplir is running`

Успешная смена пароля пользователя	`rvpn_shell[1949]: <PSW> Command: 'admin password'. The new password has been set successfully.`
Не удалось сменить пароль пользователя	`rvpn_shell[1949]: <PSW> Command: 'admin password' could not be executed. The old password is incorrect.`
Удаление справочников и ключей	`rvpn_shell[1664]: <RMKEYS> Command 'admin remove keys'`
Удаление файлов журнала устранения неполадок и журнала транспортных конвертов MFTP	`syslogd (GNU inetutils 1.9): restart` `rvpn_shell[1664]: <ADM_LOG> 'admin remove logs' (remove the logs).`
Успешная установка справочников и ключей	`root: Keys will be installed from usb/tftp` `root: 'abn_000a.dst' has been successfully unpacked.`
Добавление ключей РНПК	`rvpn_shell[1949]: <ADM_IMP> Command 'admin add spare keys'.`
Изменение способа аутентификации пользователя на «Устройство»	`rvpn_shell[1664]: command : admin authentication-type-token` `rvpn_shell[1664]: The two-factor authentication has been enabled successfully.`
Локальное обновление ПО	`rvpn_shell[1664]: Command: admin upgrade, starting upgrade script`

Событие	Пример записей в журнале
Удаленное обновление ПО	mftpd[5776]: StartVUpgrade: Calling './vupgrade /opt/vipnet/ccc' mftpd[5776]: run command: ./vupgrade. Arguments: ……. vupgrade: Check /mnt/main/opt/vipnet/user/mftpd-config.crg success. vupgrade: Check /mnt/main/etc/failoverd-config.crg success.
Задание IP-адреса для сетевого интерфейса	rvpn_shell[1949]: <IFCONF> Command: inet ifconfig eth2 10.0.0.1 netmask 255.255.255.0 Old ifconfig value: BROADCAST MULTICAST MTU:1500 Metric:1
Сброс настроек сетевого интерфейса	rvpn_shell[1949]: Interface eth2 is reset.
Включение сетевого интерфейса	rvpn_shell[1949]: <IFCONF> The command ‘inet ifconfig eth2 up‘ executed.
Выключение сетевого интерфейса	rvpn_shell[1949]: <IFCONF> The command ‘inet ifconfig eth2 down‘ executed.
Установка параметров скорости сетевого интерфейса	rvpn_shell[1949]: <IFCONF_SPD> Command: inet ifconfig eth2 speed 10 duplex full
Добавление статического маршрута	rvpn_shell[1949]: <RT_ADD> Command: inet route add 123.45.67.89 netmask 255.255.255.0 gateway 192.168.32.50. Static route has been added successfully.
Удаление статического маршрута	rvpn_shell[1949]: <RT_DEL> Command: inet route delete 123.45.67.89. Static route has been deleted successfully.
Создания виртуального интерфейса VLAN	rvpn_shell[1949]: <I_VLAN_ADD> Command: ‘inet ifconfig eth2 vlan add 2‘. The new vlan interface eth2.2 has been created.
Удаление виртуального интерфейса VLAN	rvpn_shell[1949]: <I_VLAN_DELETE> Command: ‘inet ifconfig eth2 vlan delete 2‘. The vlan interface eth2.2 has been deleted.
Сохранение копии конфигурации VPN	rvpn_shell[1949]: <CFG_SV> Command: admin config save test_conf.
Загрузка копии конфигурации VPN	rvpn_shell[1949]: <CFG_LD> Command: admin config load test_conf.
Экспорт журнала регистрации IP-пакетов на USB-носитель	rvpn_shell[1949]: <ADM_EXP> Command: ‘admin export packetdb usb’.
Экспорт файлов журнала устранения неполадок	rvpn_shell[1949]: <ADM_EXP> Command: admin export logs usb
Экспорт справочников, ключей и настроек	rvpn_shell[1949]: <ADM_EXP> Command: ‘admin export binary-encrypted’.

Событие	Пример записей в журнале

Изменение сетевых фильтров и правил трансляции адресов	Отображение изменения:
	`iplircfg[21590]: User rules dataname has been changed.`
	`iplircfg[21590]: <Rules>`
	`iplircfg[21590]: + <Rule IsActive="true" xsi:type="vn:LocalRule">`
	`iplircfg[21590]: + <RuleId>100001</RuleId>`
	`iplircfg[21590]: </Rule>`

Включение вывода сообщений о событиях	`rvpn_shell[1949]: <DBG_ON> Command 'debug on -device tty1, facility daemon, level debug'.`
Выключение вывода сообщений о событиях	`rvpn_shell[1949]: <DBG_OFF> Command 'debug off -device tty1'.`

Таблица 25. События, связанные с запуском или остановкой демонов ViPNet Coordinator HW

Событие	Пример записей в журнале
Загрузка драйверов и запуск демонов ViPNet Coordinator HW	`rvpn_shell[1399]: <ADM_VSTART> Command 'vpn start'.`
Выгрузка драйверов и завершение работы демонов ViPNet Coordinator HW	`rvpn_shell[1399]: <ADM_VSTOP> Command 'vpn stop'.`
Запуск демона failoverd, отвечающего за функционирование системы защиты от сбоев	`failoverd[1479]: Starting failover in single/cluster mode` `rvpn_shell[1399]: <F_STR> Command: failover start`
Завершение работы демона failoverd, отвечающего за функционирование системы защиты от сбоев	`failoverd[1217]: Caught quit signal, exiting` `rvpn_shell[1399]: <F_STP> Command: failover stop`
Запуск управляющего демона iplircfg	`iplircfg[1448]: IPLIR daemon is running` `rvpn_shell[1399]: <I_STR> Command: iplir start`
Завершение работы управляющего демона iplircfg	`iplircfg[1363]: Exiting daemon` `rvpn_shell[1399]: <I_STP> Command: iplir stop`
Запуск транспортного модуля MFTP	`rvpn_shell[1399]: <M_STR> Command: mftp start` `mftpd[1463]: Init MFTP manager ... OK`
Завершение работы транспортного модуля MFTP	`mftpd[1229]: Exiting MFTP manager` `rvpn_shell[1399]: <M_STP> Command: mftp stop`
Перезапуск демона algd, отвечающего за обработку прикладных протоколов	`rvpn_shell[1399]: <ALG_RST> Command: 'alg restart'.`
Перезапуск веб-сервера ViPNet	`rvpn_shell[1399]: <WUI_RST> Command: webui`

Событие	Пример записей в журнале
Coordinator HW	`restart`

Таблица 26. События, связанные с работой системы защиты от сбоев

Событие	Пример записей в журнале
Смена режима работы сервера кластера из пассивного в активный	`failoverd[1284]: switching to active mode`

Перезагрузка сервера кластера из-за сбоя на сетевом интерфейсе	`failoverd[1258]: failure detected on interface eth1`
	`failoverd[1258]: one or more subsystems failed, rebooting`
	`failoverd[1258]: Rebooted due to network error on eth1 at <дата и время>`

Перезагрузка сервера кластера из-за конфликта режимов	`failoverd[1411]: Rebooted due conflict mode at <дата и время>`
Перезагрузка сервера кластера из-за обнаружения демоном failoverd ошибки другого контролируемого демона	`failoverd[1411]: [02-01 21:12:54] CheckApp: [mentor.cpp]: Reached maximum count of application restarts [iplircfg] pid 0` `failoverd[1411]: Rebooted due to watcher detected error at <дата и время>`

Таблица 27. События, связанные с обменом служебной информацией между узлами ViPNet

Событие	Пример записей в журнале
Получение сообщения о доступности узла ViPNet (сообщения рассылаются при включении узла, а затем периодически)	`iplircfg[13787]: [11-16 16:40:44] ProcessMonChannel: [src/ipserver.cpp]: got message: type 1 from ID 1C2000A (msg name '85FTKIK3.IAY')`
Получение сообщения о доступности узла ViPNet (сообщения рассылаются при первом обращении к серверу IP-адресов сети ViPNet)	`iplircfg[13787]: [11-16 16:40:39] ProcessMonChannel: [src/ipserver.cpp]: got message: type 3 from ID 1C2000A (msg name '773PYNG3.2HH')`
Первое обращение к серверу соединений, если сервер не является также сервером IP-адресов сети ViPNet	`iplircfg[13787]: [11-16 16:40:39] ProcessMonChannel: [src/ipserver.cpp]: got message: type 23 from ID 1C2000A (msg name '773PYNG3.2HH')`
Отключение узла ViPNet от сервера IP-адресов сети ViPNet	`iplircfg[13787]: [11-16 16:40:44] ProcessMonChannel: [src/ipserver.cpp]: got message: type 2 from ID 1C2000A (msg name '85FTKIK3.IAY')`

Событие	Пример записей в журнале
Проверка соединения с узлом ViPNet (запрос)	`iplircfg[13787]: [11-16 16:40:44] ProcessMonChannel: [src/ipserver.cpp]: got message: type 5 from ID 1C2000A (msg name '85FTKIK3.IAY')`
Подтверждение получения сообщения (кроме уведомлений об изменении адреса доступа удаленного узла ViPNet)	`iplircfg[13787]: [11-16 16:40:44] ProcessMonChannel: [src/ipserver.cpp]: got message: type 7 from ID 1C2000A (msg name '85FTKIK3.IAY')`
Уведомление об изменении адреса доступа удаленного узла ViPNet	`iplircfg[13787]: [11-16 16:40:44] ProcessMonChannel: [src/ipserver.cpp]: got message: type 50 from ID 1C2000A (msg name '85FTKIK3.IAY')`
Подтверждение получения уведомления об изменении адреса доступа удаленного узла ViPNet	`iplircfg[13787]: [11-16 16:40:44] ProcessMonChannel: [src/ipserver.cpp]: got message: type 70 from ID 1C2000A (msg name '85FTKIK3.IAY')`

Изменение параметров адреса доступа удаленного узла ViPNet	`iplircfg[10937]: [12-07 15:33:50] CConfig.SubstNatSettings: subst proxy 01C3000A natsettings for changed ID 01C30B4D: firewallIp 79.133.171.237, port 55777`
	`iplircfg[10937]: [12-07 15:33:46] CConfig::SubstNatSettingsByProxy: subst changed proxy 01D60032 natsettings for ID 01D6004D: firewallIp 10.54.100.10, port 55777`
	`iplircfg[10937]: [12-07 15:35:26] CConfig.SetIpsForId: load natsettings for 5C1A37: firewallip = 88.151.200.50, forwardip = 88.151.200.50, port = 55777, timeout = 0, virtualip = 0, proxyid = 5C05CF, DRV_ALWAYS_USE_SERVER = yes DRV_GREEN_NODE = yes DRV_STOP_FIREWALL_IP=no`

Таблица 28. События, связанные с ошибками датчика случайных чисел (ДСЧ) и шифратора

Событие	Пример записей в журнале

Проверка при запуске и регламентная проверка ДСЧ в демонах iplircfg и mftpd	`iplircfg[2736]: NewRngInitializer failure: NewRandInit has failed: 100`
	`iplircfg[2736]: Failed to initialize CSP`
	`iplircfg[2736]: Please, restart application manually.`

Проверка при запуске и регламентная проверка шифраторов в демонах iplircfg и mftpd	`iplircfg[2370]: InitCSP has failed`
	`iplircfg[2370]: Failed to initialize CSP`
	`iplircfg[2370]: Please, restart application manually.`

Событие	Пример записей в журнале

Периодическая проверка ДСЧ в демонах	`mftpd[1950]: [04-25 17:15:12] CPGenRandom: CPGenRandom unsuccessful, length 8, ErrorCode - 2146893795`
	`mftpd[1950]: [04-25 17:15:12] EncryptData: FD 15: Can't encrypt data for alien ID 28750011: 'CCryptoApi::Encrypt()'`
	`mftpd[1950]: [04-25 17:15:12] FormINF8Data: FD 15: Can't encrypt data`
	`mftpd[1950]: [04-25 17:15:12] Init: FD 15: Can't form INF8 data`
	`mftpd[1950]: [04-25 17:15:12] SendERRCommand: FD 15 -> "ERR"`
	`mftpd[1950]: [04-25 17:15:12] ProcessMftpConnection: Can't init MFTP connection. Sock 15.`
	`mftpd[1950]: [04-25 17:15:12] CMftpConnection.Close: Close connection. Sock 15.`

Периодическая проверка шифраторов в демонах	`mftpd[1950]: [04-25 17:15:12] EncryptData: FD 15: Can't encrypt data for alien ID 28750011: 'CCryptoApi::Encrypt()'`
	`mftpd[1950]: [04-25 17:15:12] FormINF8Data: FD 15: Can't encrypt data`
	`mftpd[1950]: [04-25 17:15:12] Init: FD 15: Can't form INF8 data`
	`mftpd[1950]: [04-25 17:15:12] SendERRCommand: FD 15 -> "ERR"`
	`mftpd[1950]: [04-25 17:15:12] ProcessMftpConnection: Can't init MFTP connection. Sock 15.`
	`mftpd[1950]: [04-25 17:15:12] CMftpConnection.Close: Close connection. Sock 15.`

Проверка при запуске и регламентная проверка ДСЧ в криптодрайвере	`vmunix: [   34.573918] itcscrpt: Run CSP test.` `vmunix: [   34.573920] itcscrpt: Driver selftest: TestLinealRecurrentGenerator failed`

Проверка при запуске и регламентная проверка шифраторов в криптодрайвере	`vmunix: [   34.206950] itcscrpt: Run CSP test.`
	`vmunix: [   34.206957] itcscrpt: Driver selftest: TestLowStreamImit failed`
	`vmunix: [   34.206957] itcscrpt: Driver selftest: TestLowCloseStreamImit failed`
	`vmunix: [   34.206958] itcscrpt: Driver selftest: TestLowStreamEnCryptCFB failed`
	`vmunix: [   34.206959] itcscrpt: Driver selftest: TestLowStreamDeCryptCFB failed`

Событие	Пример записей в журнале

Периодическая проверка ДСЧ в криптодрайвере	`vmunix: [  652.427598] itcscrpt: Driver selftest: TestLowStreamImit failed`
	`vmunix: [  652.427600] itcscrpt: Driver selftest: TestLowCloseStreamImit failed`
	`vmunix: [  652.427601] itcscrpt: Driver selftest: TestLowStreamEnCryptCFB failed`
	`vmunix: [  652.427602] itcscrpt: Driver selftest: TestLowStreamDeCryptCFB failed`
	`vmunix: [  652.427606] itcscrpt: Driver selftest: There are tests with failures.`

Примечание. Записи об успешной работе ДСЧ или шифратора не протоколируются.

Таблица 29. События, связанные с работой антивируса и контроля содержимого трафика прокси-сервера

Событие	Пример записей в журнале
Сработало запрещающее правило контроля содержимого трафика по HTTP-методу	`0 192.168.56.121 TCP_DENIED/403 4293 GET http://www.address.com/ - NONE - text/html`
Сработало запрещающее правило контроля содержимого трафика по MIME-типу	`657 192.168.56.121 TCP_DENIED_REPLY/403 3974 GET http://www.address.com/file.zip - FIRST_UP_PARENT/10.0.25.3 text/html`
Антивирус заблокировал скачивание зараженного файла	`Nov 17 20:49:01 hw-va-164a0013 icapserver[6069]: RESPMOD 127.0.0.1 - 192.168.15.233 http://www.rexswain.com/eicar.zip INFECTED EICAR-Test-File`

E

Список демонов в составе ПО ViPNet Coordinator HW

Ниже приведен список демонов в составе ПО ViPNet Coordinator HW, для которых можно настроить фильтрацию вывода событий журнала устранения неполадок (см. «Просмотр журнала устранения неполадок» на стр. 222):

Демоны ViPNet:

- algd — демон ALG NAT прокси;
- bonding_helper — скрипт управления модуля ядра, реализующего функционал агрегированных каналов;
- drviplir — драйвер перехвата сетевого трафика (модуль ядра);
- failover, failoverd — демон слежения за основными службами ViPNet и организации кластера;
- http_proxy — управляющие скрипты прокси-сервера squid;
- l2overip — драйвер, организующий прозрачное соединение сегментов сети на уровне L2 (модуль ядра);
- Iplircfg — демон, управляющий настройками драйвера перехвата сетевого трафика, ведения журнала IP пакетов, обновления справочно-ключевой информации;
- Iplirpasswd — утилита проверки паролей пользователя и администратора ViPNet;
- Itcscrpt — крипто драйвер шифрования IP пакетов (модуль ядра);
- mftp, mftpd — демон транспортного сервиса ViPNet;
- ntp_service — управляющий скрипт NTP сервиса;
- rvpn_shell — командный интерпретатор ViPNet;

- session_watcher — демон, отслеживающий сессии пользователей ViPNet;
- unmerge — утилита распаковки дистрибутива справочников и ключей;
- vipnet-web-gui — демон веб-интерфейса ViPNet Coordinator HW;
- vpn-snmpd — демон SNMP;
- vupgrade — утилита обновления ПО ViPNet Coordinator HW.

Демоны Linux:

- acpid (ACPI event daemon) — демон для реакции на ACPI события, например, реакция на нажатие кнопки питания;
- bonding — модуль ядра, реализующий функционал агрегированных каналов;
- chat — скрипт управления службы установки соединения по модему;
- crond — демон для запуска задач по расписанию;
- hostapd — демон wi-fi AP;
- login — демон для аутентификации пользователя и открытия его сессии;
- named — демон DNS;
- ntpd — демон NTP;
- pcscd — демон pcsc для работы с токенами;
- pppd — демон установки соединения по модему;
- sshd — демон SSH;
- squid — демон прокси-сервера squid;
- syslogd — демон системного журнала;
- udhcpc — демон DHCP-клиента;
- udhcpd — демон DHCP-сервера;
- upsd — демон UPS;
- upsmon — демон монитора UPS;
- vmunix — ядро системы;
- watchquagga — демон слежения за службой динамической маршрутизации;
- wpa_cli — демон Wi-Fi-клиента;
- wpa_supplicant — скрипт управления демоном Wi-Fi-клиента;
- xinetd — демон, запускающий сетевые серверные процессы;
- zebra — демон динамической маршрутизации;
- zdhcpd — демон протокола DHCP службы динамической маршрутизации;
- zospfd — демон протокола OSPF службы динамической маршрутизации.

F

Поддерживаемые типы содержимого

Ниже приведен список MIME-типов, которые можно указать в параметре `<тип содержимого>` при настройке правил фильтрации содержимого трафика (см. «Настройка фильтрации содержимого трафика» на стр. 170).

- **Application — Внутренний формат прикладной программы**
  - application/atom+xml: Atom
  - application/EDI-X12: EDI X12 (RFC 1767)
  - application/EDIFACT: EDI EDIFACT (RFC 1767)
  - application/json: JavaScript Object Notation JSON (RFC 4627)
  - application/javascript: JavaScript (RFC 4329)
  - application/octet-stream: двоичный файл без указания формата (RFC 2046)
  - application/ogg: Ogg (RFC 5334)
  - application/pdf: Portable Document Format, PDF (RFC 3778)
  - application/postscript: PostScript (RFC 2046)
  - application/soap+xml: SOAP (RFC 3902)
  - application/font-woff: Web Open Font Format
  - application/xhtml+xml: XHTML (RFC 3236)
  - application/xml-dtd: DTD (RFC 3023)
  - application/xop+xml:XOP

- application/zip: ZIP
- application/gzip: Gzip
- application/x-bittorrent: BitTorrent
- application/x-tex: TeX

- **audio — Аудио**
  - audio/basic: mulaw аудио, 8 кГц, 1 канал (RFC 2046)
  - audio/L24: 24bit Linear PCM аудио, 8-48 кГц, 1-N каналов (RFC 3190)
  - audio/mp4: MP4
  - audio/aac: AAC
  - audio/mpeg: MP3 или др. MPEG (RFC 3003)
  - audio/ogg: Ogg Vorbis, Speex, Flac или др. аудио (RFC 5334)
  - audio/vorbis: Vorbis (RFC 5215)
  - audio/x-ms-wma: Windows Media Audio
  - audio/x-ms-wax: Windows Media Audio перенаправление
  - audio/vnd.rn-realaudio: RealAudio
  - audio/vnd.wave: WAV(RFC 2361)
  - audio/webm: WebM
- **image — Изображение**
  - image/gif: GIF(RFC 2045 и RFC 2046)
  - image/jpeg: JPEG (RFC 2045 и RFC 2046)
  - image/pjpeg: JPEG
  - image/png: Portable Network Graphics[9](RFC 2083)
  - image/svg+xml: SVG
  - image/tiff: TIFF(RFC 3302)
  - image/vnd.microsoft.icon: ICO
  - image/vnd.wap.wbmp: WBMP
- **message — Сообщение**
  - message/http (RFC 2616)
  - message/imdn+xml: IMDN (RFC 5438)
  - message/partial: E-mail (RFC 2045 и RFC 2046)
  - message/rfc822: E-mail; EML файлы, MIME файлы, MHT файлы, MHTML файлы (RFC 2045 и RFC 2046)
- **model — Для 3D моделей**

- model/example: (RFC 4735)
- model/iges: IGS файлы, IGES файлы (RFC 2077)
- model/mesh: MSH файлы, MESH файлы (RFC 2077), SILO файлы
- model/vrml: WRL файлы, VRML файлы (RFC 2077)
- model/x3d+binary: X3D ISO стандарт для 3D компьютерной графики, X3DB файлы
- model/x3d+vrml: X3D ISO стандарт для 3D компьютерной графики, X3DV VRML файлы
- model/x3d+xml: X3D ISO стандарт для 3D компьютерной графики, X3D XML файлы
- multipart
- multipart/mixed: MIME E-mail (RFC 2045 и RFC 2046)
- multipart/alternative: MIME E-mail (RFC 2045 и RFC 2046)
- multipart/related: MIME E-mail (RFC 2387 и используемое MHTML (HTML mail))

- **multipart — составные файлы**
  - multipart/form-data: MIME Webform (RFC 2388)
  - multipart/signed: (RFC 1847)
  - multipart/encrypted: (RFC 1847)
- **text — Текст**
  - text/cmd: команды
  - text/css: Cascading Style Sheets (RFC 2318)
  - text/csv: CSV (RFC 4180)
  - text/html: HTML (RFC 2854)
  - text/javascript (Obsolete): JavaScript(RFC 4329)
  - text/plain: текстовые данные (RFC 2046 и RFC 3676)
  - text/php: Скрипт языка PHP
  - text/xml: Extensible Markup Language (RFC 3023)
- **video — Видео**
  - video/mpeg: MPEG-1 (RFC 2045 и RFC 2046)
  - video/mp4: MP4 (RFC 4337)
  - video/ogg: Ogg Theora или другое видео (RFC 5334)
  - video/quicktime: QuickTime
  - video/webm: WebM
  - video/x-ms-wmv: Windows Media Video
  - video/x-flv: FLV
  - video/3gpp: .3gpp .3gp

- video/3gpp2: .3gpp2 .3g2

- **vnd — Вендорные файлы**
  - application/vnd.oasis.opendocument.text: OpenDocument
  - application/vnd.oasis.opendocument.spreadsheet: OpenDocument
  - application/vnd.oasis.opendocument.presentation: OpenDocument
  - application/vnd.oasis.opendocument.graphics: OpenDocument
  - application/vnd.ms-excel: Microsoft Excel файлы
  - application/vnd.openxmlformats-officedocument.spreadsheetml.sheet: Microsoft Excel 2007 файлы
  - application/vnd.ms-powerpoint: Microsoft Powerpoint файлы
  - application/vnd.openxmlformats-officedocument.presentationml.presentation: Microsoft Powerpoint 2007 файлы
  - application/msword: Microsoft Word файлы
  - application/vnd.openxmlformats-officedocument.wordprocessingml.document: Microsoft Word 2007 файлы
  - application/vnd.mozilla.xul+xml: Mozilla XUL файлы
  - application/vnd.google-earth.kml+xml: KML файлы (например, для Google Earth)
- **x — Нестандартные файлы**
  - application/x-www-form-urlencoded Form Encoded Data
  - application/x-dvi: DVI
  - application/x-latex: LaTeX файлы
  - application/x-font-ttf: TrueType (не зарегистрированный MIME-тип, но наиболее часто используемый)
  - application/x-shockwave-flash: Adobe Flash
  - application/x-stuffit: StuffIt
  - application/x-rar-compressed: RAR
  - application/x-tar: Tarball
  - text/x-jquery-tmpl: jQuery
  - application/x-javascript:
- **x-pks — PKCS**
  - application/x-pkcs12: p12 файлы
  - application/x-pkcs12: pfx файлы
  - application/x-pkcs7-certificates: p7b файлы
  - application/x-pkcs7-certificates: spc файлы

- application/x-pkcs7-certreqresp: p7r файлы
- application/x-pkcs7-mime: p7c файлы
- application/x-pkcs7-mime: p7m файлы
- application/x-pkcs7-signature: p7s файлы

G

Дополнительные рекомендации для администратора, использующего ViPNet Coordinator HW 4 в качестве межсетевого экрана типа «А» четвертого класса защиты

- Запретите трафик от нежелательных приложений. Для этого создайте сетевые фильтры, запрещающие передачу данных через порты, которые используют эти приложения (см. «Создание сетевого фильтра» на стр. 124).

  Разрешите трафик от доверенных приложений. Для этого создайте сетевые фильтры, разрешающие передачу данных через порты, которые используют эти приложения.

- Для ассоциации пользователей сети и IP-адресов:
  - для каждого сетевого узла задайте статический IP-адрес;

- занесите в специальный журнал ФИО физических лиц, имеющих учетные записи на сетевых узлах;
- организуйте фиксацию времени работы пользователей на сетевых узлах, где зарегистрировано несколько учетных записей, в специальном журнале.

Глоссарий

COM-консоль

Ноутбук, подключенный к COM-порту, который используется для локальной настройки ViPNet Coordinator HW.

DHCP (Dynamic Host Configuration Protocol)

Сетевой протокол прикладного уровня, позволяющий компьютерам автоматически получать IP-адреса и другие параметры, необходимые для работы в сети TCP/IP. К таким параметрам относятся маска подсети, IP-адрес шлюза, IP-адреса серверов DNS, IP-адреса серверов WINS.

DHCP-сервер

Сервер, автоматически администрирующий IP-адреса DHCP-клиентов и выполняющий соответствующую настройку для сети.

DNS-сервер

Сервер, содержащий часть базы данных DNS, используемой для доступа к именам компьютеров в интернет-домене. Например, ns.domain.net. Как правило, информация о домене хранится на двух DNS-серверах, называемых «Primary DNS» и «Secondary DNS» (дублирование делается для повышения отказоустойчивости системы).

Также DNS-сервер называют сервером доменных имен, сервером имен DNS.

L2OverIP

Технология, которая позволяет организовать защиту удаленных сегментов сети, использующих одно и то же адресное пространство, на канальном уровне модели OSI. В результате узлы из разных сегментов смогут взаимодействовать друг с другом так, как будто они находятся в одном

сегменте с прямой видимостью по MAC-адресам. В основе технологии лежит перехват на канальном уровне модели OSI Ethernet-кадров, отправленных из одного сегмента сети в другой.

NTP-сервер

Сервер точного времени, который необходим для синхронизации времени компьютеров, рабочих станций, серверов и прочих сетевых устройств. Этот сервер играет роль посредника между эталоном времени и сетью. Он получает время от эталона по специальному каналу (интерфейсу) и выдает его для любого узла сети, обеспечивая тем самым синхронизацию устройств.

OSPF (Open Shortest Path First)

Протокол динамической маршрутизации, основанный на технологии отслеживания состояния канала для нахождения кратчайшего маршрута. Распространяет информацию о доступных маршрутах внутри автономной системы.

PPP (Point-to-Point Protocol)

Протокол канального уровня, использующийся для установления прямой связи между двумя узлами сети.

TCP-туннель

Способ соединения клиентов ViPNet, находящихся во внешних сетях, с другими узлами сети ViPNet по протоколу TCP. Используется в том случае, если соединение по протоколу UDP заблокировано провайдерами услуг Интернета.

TCP-туннель настраивается на координаторе, который является для клиента сервером соединений.

ViPNet Policy Manager

Программа, которая входит в состав программного комплекса ViPNet. Предназначена для централизованного управления политиками безопасности узлов защищенной сети ViPNet.

ViPNet Центр управления сетью (ЦУС)

ViPNet Центр управления сетью — это программа, входящая в состав программного обеспечения ViPNet Administrator. Предназначена для создания и управления конфигурацией сети и позволяет решить следующие основные задачи:

- построение виртуальной сети (сетевые объекты и связи между ними, включая межсетевые);
- изменение конфигурации сети;
- формирование и рассылка справочников;
- рассылка ключей узлов и ключей пользователей;
- формирование информации о связях пользователей для УКЦ;
- задание полномочий пользователей сетевых узлов ViPNet.

VLAN

Виртуальная локальная компьютерная сеть, представляет собой группу узлов с общим набором требований, которые взаимодействуют так, как если бы они были подключены к широковещательному домену, независимо от их физического местонахождения. VLAN имеет те же свойства, что и физическая локальная сеть, но позволяет узлам группироваться вместе, даже если они не находятся в одной физической сети.

Автономная система

Один или несколько сегментов сети, в которых осуществляется маршрутизация по одному протоколу (OSPF, IGRP, EIGRP, IS-IS, RIP, BGP, Static). Также может трактоваться как домен маршрутизации — группа маршрутизаторов сети, работающих по одинаковым протоколам маршрутизации.

Агрегированный сетевой интерфейс

Логический сетевой интерфейс, образованный из нескольких физических интерфейсов Ethernet, объединенных на канальном уровне сетевой модели OSI.

Административная дистанция

Характеристика маршрута (см. глоссарий, стр. 269). Позволяет определить меру доверия к маршруту. Задается для любого маршрута в виде целого числа в диапазоне от 1 до 255.

Администратор сети ViPNet

Лицо, отвечающее за управление сетью ViPNet, создание и обновление справочников и ключей для сетевых узлов ViPNet, настройку межсетевого взаимодействия с доверенными сетями и обладающее правом доступа к программе ViPNet Центр управления сетью и (или) ViPNet Удостоверяющий и ключевой центр.

Вариант персонального ключа пользователя

Номер персонального ключа пользователя из резервного набора персональных ключей (РНПК). Изменение варианта персонального ключа пользователя означает, что номер персонального ключа был увеличен на единицу и для создания новых ключей пользователя будет использоваться следующий персональный ключ из РНПК.

Вес

Параметр, который задается для шлюза в статическом маршруте (см. глоссарий, стр. 269) в виде целого числа в диапазоне от 1 до 255. Позволяет настроить балансировку IP-трафика между шлюзами в одинаковый адрес назначения. Определяет долю IP-трафика, который должен передаваться по маршруту на указанный шлюз.

Виртуальная защищенная сеть

Технология, позволяющая создать логическую сеть, чтобы обеспечить множественные сетевые соединения между компьютерами или локальными сетями через существующую физическую сеть. Уровень доверия к такой виртуальной сети не зависит от уровня доверия к физическим сетям благодаря использованию средств криптографии (шифрования, аутентификации и средств персонального и межсетевого экранирования).

Виртуальный IP-адрес

IP-адрес, который приложения на сетевом узле ViPNet (А) используют для обращения к ресурсам сетевого узла ViPNet (Б) или туннелируемых им узлов вместо реального IP-адреса узла. Виртуальные IP-адреса узлу ViPNet (Б) назначаются непосредственно на узле А. На других узлах узлу ViPNet (Б) могут быть назначены другие виртуальные адреса. Узлу ViPNet (Б) назначается столько виртуальных адресов, сколько реальных адресов имеет данный узел. При изменении реальных адресов у узла Б выделенные ему виртуальные адреса не изменяются. Виртуальные адреса туннелируемых узлов привязываются к реальным адресам этих узлов и существуют, пока существует данный реальный адрес. Использование виртуальных адресов позволяет избежать конфликта реальных IP-адресов в случае, если узлы работают в локальных сетях с пересекающимся адресным пространством, а также использовать эти адреса для аутентификации удаленных узлов в приложениях ViPNet.

Динамический сетевой интерфейс

Разновидность сетевого интерфейса, который добавляется в процессе работы при наступлении некоторого события (например, при подключении встроенного или USB-модема, предоставляющего данный интерфейс).

Динамические интерфейсы объединяются в группы по типу интерфейса. Поэтому иногда может встречаться термин «групповой динамический интерфейс».

Существуют следующие группы динамических интерфейсов:

- `ppp` — группа интерфейсов для подключения к мобильной сети через встроенный модем;
- `wifi` — группа интерфейсов для подключения к беспроводной сети Wi-Fi.

Дистрибутив ключей

Файл с расширением `.dst`, создаваемый в программе ViPNet Удостоверяющий и ключевой центр для каждого пользователя сетевого узла ViPNet. Содержит справочники, ключи и файл лицензии, необходимые для обеспечения первичного запуска и последующей работы программы ViPNet на сетевом узле. Для обеспечения работы программы ViPNet дистрибутив ключей необходимо установить на сетевой узел.

Доверенная сеть

Сеть ViPNet, с узлами которой узлы своей сети ViPNet осуществляют защищенное взаимодействие.

Домен коллизий

Часть сети Ethernet, все узлы которой конкурируют за общую разделяемую среду передачи и, следовательно, каждый узел которой может создать коллизию с любым другим узлом этой части сети.

Сеть Ethernet, построенная на повторителях, всегда образует один домен коллизий. Мосты, коммутаторы и маршрутизаторы делят сеть Ethernet на несколько доменов коллизий.

Источник бесперебойного питания (UPS)

Автоматическое электронное устройство с аккумуляторной батареей, предназначенное для бесперебойного кратковременного снабжения электрической энергией компьютера и его компонентов с целью корректного завершения работы и сохранения данных в случае резкого падения или отсутствия входного питающего напряжения системы.

Класс сетевого интерфейса

Признак, определяющий назначение сетевого интерфейса. В ViPNet Coordinator HW интерфейсам можно назначить следующие классы: `access`, `trunk`, `slave`.

По умолчанию сетевому интерфейсу назначен класс `access`. Если требуется, чтобы интерфейс Ethernet или агрегированный интерфейс обрабатывал трафик из нескольких VLAN, ему необходимо назначить класс `trunk`. Чтобы объединить несколько интерфейсов Ethernet в агрегированный интерфейс, каждому из таких интерфейсов необходимо предварительно назначить класс `slave`.

Кластер горячего резервирования

Кластер горячего резервирования состоит из двух взаимосвязанных серверов ViPNet Coordinator HW, один из которых (активный) выполняет функции координатора сети ViPNet, а другой сервер (пассивный) находится в режиме ожидания. В случае сбоев, критичных для работоспособности ПО ViPNet на активном сервере, пассивный сервер переключается в активный режим для выполнения функций сбойного сервера. При этом сбойный сервер перезагружается и становится пассивным.

Клиент (ViPNet-клиент)

Сетевой узел ViPNet, который является начальной или конечной точкой передачи данных. В отличие от координатора клиент не выполняет функции маршрутизации трафика и служебной информации.

Ключ защиты

Ключ, на котором шифруется другой ключ.

Ключи узла ViPNet

Совокупность ключей, с использованием которых производится шифрование трафика, служебной информации и писем программы ViPNet Деловая почта.

Командный интерпретатор

Командная оболочка, предназначенная для администрирования программного обеспечения ViPNet Coordinator HW с помощью ряда специальных команд.

Координатор (ViPNet-координатор)

Сетевой узел, представляющий собой компьютер с установленным программным обеспечением координатора (ViPNet Coordinator) или специальный программно-аппаратный комплекс. В рамках сети ViPNet координатор выполняет серверные функции, а также маршрутизацию трафика и служебной информации.

Маршрут

Путь следования IP-трафика при передаче в сети от одного узла другому.

Маршрут по умолчанию

Путь следования IP-пакетов, для которых не был найден подходящий маршрут в таблице маршрутизации.

Маршрутизация

Процесс выбора пути для передачи информации в сети.

Маршрутизатор-сосед

OSPF-маршрутизатор, находящиеся в одной области маршрутизации с другими маршрутизаторами этого типа.

Межсетевое взаимодействие

Информационное взаимодействие, организованное между сетями ViPNet. Позволяет узлам различных сетей ViPNet обмениваться информацией по защищенным каналам. Для организации взаимодействия между узлами различных сетей ViPNet администраторы этих сетей обмениваются межсетевой информацией.

Межсетевой экран

Устройство на границе локальной сети, служащее для предотвращения несанкционированного доступа из одной сети в другую. Межсетевой экран проверяет весь входящий и исходящий IP-трафик, после чего принимается решение о возможности дальнейшего направления трафика к пункту назначения. Межсетевой экран обычно осуществляет преобразование внутренних адресов в адреса, доступные из внешней сети (выполняет NAT).

Метрика адреса доступа

Определяет задержку (в миллисекундах) отправки тестовых пакетов при выполнении опроса узла для определения доступности адреса. Предназначена для задания приоритета использования каналов связи.

Метрика маршрута

Предназначена для задания приоритета маршрута передачи IP-трафика.

Область маршрутизации

Одна или несколько IP-сетей, в которых осуществляется обмен информацией по определенному протоколу, в частности, по протоколу OSPF (см. глоссарий, стр. 265).

Протокол OSPF рассматривает межсетевую среду как множество областей, соединенных друг с другом через некоторую базовую область (backbone area). Для идентификации областей каждой из них выделяется специальный идентификатор (area ID), представляющий собой 32-разрядное число, которое записывается так же, как и IP-адрес — в десятично-точечном формате (в виде четырех однобайтовых чисел, разделенных точками).

Обычная консоль

Монитор и клавиатура, которые используются для локальной настройки ViPNet Coordinator HW.

Открытый узел

Узел, с которым обмен информацией происходит в незашифрованном виде.

Пароль администратора сетевого узла ViPNet

Пароль для входа на сетевом узле ViPNet в режим администратора, в рамках которого становятся доступны дополнительные возможности настройки приложений ViPNet. Пароль администратора сетевого узла ViPNet может быть создан администратором сети ViPNet в программе ViPNet Удостоверяющий и ключевой центр (в сетях, которые администрируются при помощи ПО ViPNet Administrator) или ViPNet Network Manager (в сетях, которые администрируются при помощи ПО ViPNet Network Manager).

Пароль пользователя

Индивидуальный пароль пользователя для работы в приложениях ViPNet на сетевом узле ViPNet. Первоначально создается администратором сети ViPNet в программе ViPNet Удостоверяющий и ключевой центр или ViPNet Network Manager. Этот пароль может быть изменен пользователем на сетевом узле ViPNet.