InfiNet Wireless Router

Руководство по ОС WANFlex

Обновлено: 11 Февраля 2008

Версия ОС: V4.29

Copyright © 2004-2007 by InfiNet Wireless Limited.

Copyright © 2004-2007 by InfiNet Wireless Limited.
All rights reserved.

Содержание

I. ВВЕДЕНИЕ 1
1. Общие принципы 1
2. Формат IP-адресов 1
II. ОБЩИЕ КОМАНДЫ 2
1. Help 2
2. System 2
3. Config (манипуляции с конфигурацией) 4
4. Set (указание TIME ZONE) 5
5. Flashnet (загрузка/выгрузка ПО) 5
6. Restart 6
7. Ping 6
8. Telnet 7
9. Tracert 7
10. Httpd 7
11. Rshd (Remote Shell) 8
12. Ipstat (модуль сбора IP-статистики) 9
13. Acl (Access Control Lists) 11
14. Sntp 11
15. Date 13
III. КОМАНДЫ УПРАВЛЕНИЯ КАНАЛЬНЫМ ЗВЕНОМ 14
1. Rfconfig (конфигурация радиоинтерфейса) 14
2. RMA (конфигурация протокола RMA) 19
Описание протокола 19
Конфигурация абонентского блока 21
Конфигурация базовой станции 21
Тестовые режимы 23
Управление процессом 24
Автоматическое управление скоростью 24
Режим работы с централизованной раздачей полномочий 25
3. Muffer 28
4. Macf 31
5. Sppp 33
6. Arp 35
IV. КОМАНДЫ УПРАВЛЕНИЯ СЕТЕВЫМ ЗВЕНОМ 37
1. Ifconfig (конфигурирование интерфейсов) 37
2. Tun 39
3. Qm (конфигурация Quality-of-Service) 41
4. Route (статические маршруты) 46

Copyright © 2004-2007 by InfiNet Wireless Limited.

5. Rip (модуль динамической маршрутизации)..... 47
6. OSPFv2 (модуль протокола динамической маршрутизации)..... 51
Введение..... 51
Командный язык. Общие принципы..... 52
Запуск/останов OSPF маршрутизатора..... 54
Идентификатор маршрутизатора..... 54
Объекты фильтрации..... 55
Анонсирование каналов..... 57
Метрика канала..... 61
Деление OSPF системы на области..... 61
Отношения смежности. Соседи..... 65
Аутентификация. Проверка подлинности..... 67
Команды просмотра состояния маршрутизатора..... 69
7. Netstat..... 73
8. Ipfw (IP Firewall)..... 74
Общее описание..... 75
Интерфейс..... 76
Фильтры..... 77
Правила обработки пакетов пакетным фильтром..... 81
Примеры..... 82
9. Loadm (монитор загрузки каналов)..... 86
10. Bpf (Berkeley Packet Filter)..... 87
11. Snmpd (SNMP daemon)..... 88
12. Td (Telnet daemon)..... 89
13. Nat (Network Address Translation)..... 89
Команды..... 90
Перенаправление пакетов..... 93
NAT и H.323 телефония..... 96
14. Trapd (SNMP traps)..... 98
15. DHCP Server..... 99
Командный язык DHCP сервера..... 99
16. DHCP relay. Команда dhcpr..... 117
Общее описание..... 117
Описание команд..... 118
17. DHCP клиент. Команда dhcpc..... 119
Общее описание..... 119
Опции..... 119
Команды..... 120
Примеры..... 120
18. VRRP сервер. Команда vrrp..... 121

Copyright © 2004-2007 by InfiNet Wireless Limited.

Общее описание..... 121
Описание команд..... 122
V. СПЕЦИАЛЬНЫЕ КОМАНДЫ..... 127
1. Ctl (управление внешними устройствами)..... 127
VI. НЕДОКУМЕНТИРОВАННЫЕ СЛУЖЕБНЫЕ КОМАНДЫ..... 129
VII. UNIFY SOFTSWITCH 130
1. Gatekeeper 130

Copyright © 2004-2007 by InfiNet Wireless Limited.

I. Введение

1. Общие принципы

Для управления и конфигурации используется простой и понятный командный язык, по структуре напоминающий систему взаимодействия ОС Unix. Каждая команда начинает действовать сразу после ввода. Однако действие каждой команды ограничивается только одним сеансом, до первой перезагрузки.

Чтобы сохранить действие команд нужно записать текущую конфигурацию в постоянную память командой **config save**.

Несколько команд можно группировать в одну строку, разделяя их символом "**;**" (точка с запятой). Если в строке встретится команда с неверным синтаксисом, то она игнорируется, остаток строки проверяется до конца. Имя команды может быть сокращено до любого недвусмысленного значения.

Если ваш терминал поддерживает стандарт **VT100** или **ANSI**, то вы можете использовать клавиши редактирования и перемещаться по списку ранее выполненных команд стандартными клавишами клавиатуры. Пронумерованный список ранее введённых команд можно просмотреть командой "**!h**". Любую строку из этого списка можно сделать текущей, с помощью команды "**!НОМЕР**". Клавиша **TAB** подставляет в командную строку последнюю похожую команду (поиск подстроки).

Команда **Ctrl/R** обновляет содержимое строки ввода, если оно было нарушено выводом на экран системных сообщений. Команда, выполненная без аргументов, печатает краткую подсказку о своих ключах и синтаксисе. Контекстную подсказку можно получить нажав клавишу "**?**" в любом месте строки.

Скорость диагностического порта - 38400 (может быть изменена командой **dport**).

Кроме того, настройку маршрутизатора можно выполнять с помощью протоколов **Telnet** и **HTTP**. Маршрутизатор поддерживает одновременно до 3 Telnet сеансов (порт 23) и одно соединение HTTP (порт 80).

2. Формат IP-адресов

Множество команд операционной системы требуют указания IP-адресов.

IP-адреса могут указываться в традиционном формате. Маска может быть записана как в количество бит (количество битов с единицей) или в традиционном формате (подобно IP-адресу). IP-адрес 0/0 подразумаевает все возможные IP-адреса.

Например:

nn.nn.nn.nn (без маски)

nn.nn.nn.nn/N (N – длина маски)

nn.nn.nn.nn:xxx.xxx.xxx.xxx (xxx.xxx.xxx.xxx – другая запись маски)

Адрес **192.168.9.0/24** описывает сеть 192.168.9.0 с маской длиной 24.

Запись **192.168.9.0:255.255.255.0** по смыслу полностью совпадает с предыдущей.

Copyright © 2004-2005 by InfiNet Wireless

II.Общие команды

1. Help

Команда выводит список всех команд маршрутизатора.

Синтаксис:

help

Описание:

Выводит список всех команд маршрутизатора. Так же выполняется в случае если пользователь ввёл неопознанную команду.

2. System

Команда служит для просмотра и установки системных параметров.

Синтаксис:

system [аргументы]

Описание:

- **system name [имя системы]**

  Если *имя системы* не указано, то выдается текущее.
  Например:

  system name Revolution.

- **system location [строка, местонахождение устройства]**

  Необязательная информационная строка, используемая в протоколе SNMP.
  Например:

  system location На крыше дома, где живёт Карлсон.

- **system gpsxy [E|W]XX.XXXXX [N|S]YY.YYYYY**

  Географические координаты устройства (долгота широта) в градусах. Этот параметр может быть использован для автоматической корректировки параметров радиомодуля при подключении к сети.

  Например:

  system gpsxy 60.40056 56.82857

- **system user *имя пользователя***

  Имя, под которым администратор будет заходить на маршрутизатор с консоли или по сети по протоколу telnet/http.
  Например:

  system user root

- **system password *пароль***

  Пароль администратора.

  Например:

  system password qwerty

Copyright © 2004-2005 by InfiNet Wireless

Если пароль в системе не установлен, то авторизация не проверяется вообще и войти можно под любым именем и паролём.

- **system prompt любое_слово**

  Позволяет изменить приглашение (prompt) клавиатурного монитора. Допустимо любое слово до 16 букв.

  Приглашение буде выглядеть в виде: Prompt#ttyN>

  Например:

  system prompt MyHost

- **system guest *ключевое слово/фраза***

  Ключевое слово для входа в гостевой режим. Вводится в качестве login, пароль можно использовать любой. В этом режиме невозможно изменить никакие настройки маршрутизатора и посмотреть параметры, влияющие на безопасность.

  Например:

  system guest for_members_only

- **system [no]fastroute**

  Включить/выключить режим ускоренной маршрутизации
  В этом режиме устройство становится невидимым для средств трассирования сети (traceroute), но при этом выполняет все функции маршрутизатора.
  Не рекомендуется включать эту опцию одновременно на нескольких устройствах, если они находятся в одном сегменте Ethernet, поскольку это может вызвать шторм IP пакетов.

- **system icmplimit XX**

  Устанавливает ограничение на количество исходящих ICMP пакетов в секунду (по умолчанию 200). Позволяет избежать перегрузки устройства при работе программ сканирования сети.
  При установке в 0 (ноль), все ограничения отключаются.

- **system** [**no**]**sendredirects**

  Включает (отключает) способность системы отправлять сообщения **icmp redirect** для подавления источника пакетов при неправильно настроенной маршрутизации.

- **system** [**no**]**dropredirects**

  Включает (отключает) способность системы принимать сообщения **icmp redirect** для корректировки маршрутных таблиц при неправильно настроенной маршрутизации.

- **system uptime**

  Показывает время работы системы с момента последней перезагрузки.

- **system cpu**

  Отображает текущую загрузку центрального процессора в процентах за несколько интервалов времени

- **system search [seconds]**

  Включает режим мигания всей световой индикации для поиска устройства среди группы однотипных.
  По умолчанию режим отключается через 10 секунд.

Copyright © 2004-2005 by InfiNet Wireless

- **system version**

  Показывает версию программного обеспечения.

- **system OfficialAddress IP**

  Команда "sys OfficialAddress IP" устанавливает адрес, который будет использоваться в качестве источника во всех исходящих с устройства соединениях.

- **system log [args]**

  Управление системным журналом событий Аргументы команды:

  - **on** - выводить сообщения на текущую консоль
  - **off** - отключить вывод сообщений на консоль
  - **IP-ADDR** - указать адрес UNIX машины для передачи сообщений в системный журнал по протоколу **syslog**. В настройках syslogd указать facility.level равным "user.notice" или просто числовое значение 15.
  - **"-" -** отключить регистрацию на удалённой машине
  - **show** - показать системный архив сообщений (самое свежее внизу, время в секундах/миллисекундах от текущего)
  - **clear** - очистить архив сообщений
  - **[no]filter -** эта опция удаляет из системного журнала одинаковые соседние строки, оставляя только один экземпляр каждого сообщения и счётчик их повторений (по умолчанию включена)

- **sys serialCD** [**no**]**log** [**no**]**trap**

  Устанавливает реакцию на появление/пропадание сигнала Carrier Detect (CD) диагностического порта (консоли). Опция "**log**" включает запись событий в системный журнал (sys log). Опция "**trap**" включает генерацию SNMP-trap сообщений. Сообщение об исчезновении сигнала CD имеет oid 1.3.6.1.4.1.3942.0.103
  Сооообщение о появлении сигнала CD имеет oid 1.3.6.1.4.1.3942.0.104 Для того, чтобы посылать сооообщения в формате SNMP необходимо запустить сервис trapd командой "trapd on" и установить необходимые адреса (см. команду **trapd**).

Любой строковый параметр можно удалить задав значение "-". Установка и отмена любых параметров может быть сохранена в постоянной конфигурации командой **config save.**

3. Config (манипуляции с конфигурацией)

Манипуляции с конфигурацией

Синтаксис:

co[nfig] show [patterns]

co[nfig] save

co[nfig] export|import login:password@host/file

Copyright © 2004-2005 by InfiNet Wireless

Описание:

- **show** - показывает текущую рабочую конфигурацию системы. Любое действие по изменению параметров системы немедленно отображается в результатах команды **config show**. Однако все изменения действительны только в течение текущего сеанса работы (до первой перезагрузки). В качестве дополнительных аргументов можно указывать начальные буквы системных команд. **Например:**

co show rma rip

покажет конфигурацию протоколов RMA и RIP

co show r !rip

покажет конфигурацию всех команд, начинающихся на "r" , кроме "rip".

- **save** - сохраняет текущую конфигурацию системы в энергонезависимую память маршрутизатора для последующего постоянного использования.
- **export, import** - позволяют сохранить или загрузить конфигурацию маршрутизатора на удалённом сервере. Операция выполняется по протоколу FTP. Файл должен быть задан полным именем в формате используемой файловой системы.

Например:

config export user:secret@192.168.1.1/var/conf/test.cfg

4. Set (указание TIME ZONE)

Команда служит для установки часового пояса. Доступна начиная с версии 3.36.

Синтаксис:

set TZ TIMEZONE

Например:

set TZ EST+5EDT,M4.1.0/2,M10.5.0/2

set TZ EKT+5

Подробнее о формате задания часового пояса: http://www.rt.com/man/tzset.3.html

5. Flashnet (загрузка/выгрузка ПО)

Команда служит для загрузки свежего ПО на маршрутизатор

Синтаксис:

flashnet get/put login[:password]@хост/файл

Описание:

Команда **flashnet** позволяет загрузить в маршрутизатор новую версию программного обеспечения – для этого используется опция **get**. Загрузка выполняется по протоколу FTP. В качестве имени файла следует указать полное имя в формате используемой файловой системы.

Например:

flashnet get upgrade@192.168.1.1/conf/rwr/rwr300.bin

Copyright © 2004-2005 by InfiNet Wireless

Процесс загрузки делится на две фазы:

- считывание файла с удалённого сервера и проверка его целостности
- запись образа системы в память маршрутизатора.

Вторая фаза отображается на экране чередующимися знаками "O" и ".". **Не прерывайте** этот процесс, иначе устройство придёт в полную негодность и восстановить его можно будет только в условиях фирмы-производителя.

***Не прерывайте** этот процесс, иначе устройство придёт в полную негодность и восстановить его можно будет только в условиях фирмы-производителя.*

Используя опцию **put** можно выгрузить программное обеспечение на внешний сервер.

6. Restart

Команда служит для перезагрузки маршрутизатора.

Синтаксис:

restart [y]

restart SECONDS

restart stop

Описание:

Команда **restart** выполняет полную перезагрузку и переинициализацию устройства также как и в случае повторного включения питания. Может быть использована для восстановления исходной конфигурации после серии безуспешных попыток понять, что именно сделано не так, а также после загрузки новой версии программного обеспечения. Если указан аргумент "**y"**, то рестарт выполняется сразу, без запроса подтверждения.
Команда может быть использована для отложенной перегрузки устройства через заданное количество секунд (**restart 300**). Это может быть полезно в случае проведения рискованных манипуляций с конфигурацией, когда есть опасность потерять контроль над устройством. Система будет периодически предупреждать об оставшемся времени до перезагрузки сообщением в системный журнал. Повторное выполнение команды начинает отсчёт времени сначала. Отменить отложенную перезагрузку можно командой **restart stop**.

7. Ping

Посылка тестовых пакетов

Синтаксис:

ping IP [size|-s SIZE] [count|-c COUNT] [source|-S SRC_IP]

Описание:

Посылает тестовые пакеты (ICMP_ECHO_REQUEST) на заданный адрес (IP). Позволяет оценить достижимость и время отклика прохождения пакетов. Дополнительно можно задать длину тестового пакета в диапазоне от 10 до 8000 байт (по умолчанию 64) и количество пакетов которые требуется получить в ответ (по умолчанию 5).

Copyright © 2004-2005 by InfiNet Wireless

8. Telnet

Telnet клиент.

Синтаксис:

telnet IP_адрес

Описание:

Команда **telnet** используется для установления соединения с удалённой машиной в режиме эмуляции терминала. В данной реализации используется простая прозрачная трансляция потока символов без какой-либо промежуточной интерпретации, поэтому тип терминала будет определяться тем терминалом с которого была выполнена данная команда. Аварийное прерывание сеанса выполняется при нажатии клавиш **Ctrl/D.**

9. Tracert

Трассировка маршрута.

Синтаксис:

tracert [-s SourceAddress] HostAddress

Описание:

Трассирует путь прохождения пакетов до заданного узла `HostAddress`. Команда использует различные значения поля протокола IP 'time to live' в исходящих пакетах и анализирует ответы `ICMP TIME_EXCEEDED`, поступающие от всех маршрутизаторов, находящихся на пути к выбранному узлу.
По умолчанию, в качестве адреса источника подставляется адрес интерфейса через который отправляются пакеты. С помощью опции **-s** можно подставить любой другой IP-адрес.

Трассировка ограничивается 30-ю промежуточными узлами. Длина отправляемых пакетов равна 36 байтам. Для каждого промежуточного узла выполняется 3 попытки.

Результат трассировки содержит адрес промежуточного марщрутизатора и время отклика для каждой попытки в миллисекундах.

Кроме того, могут появляться специальные символы, отображающие коды специфических ответов протокола ICMP:

- **!** - Порт недостижим
- **!N** - Сеть недостижима
- **!H** - Узел недоступен
- **!P** - Недопустимый протокол
- **!F** - Пакет превышает допустимую длину
- **!X** - Административый запрет на доступ к узлу (фильтр, прокси, и т.д.)
- *** -** Нет отклика.

10. Httpd

Поддержка протокола HTTP.

Синтаксис:

httpd enable | disable

Copyright © 2004-2005 by InfiNet Wireless

Описание:

Наличие поддержки протокола HTTP позволяет выполнять конфигурацию маршрутизатора с помощью любого доступного WEB броузера. Мини сервер, реализованный в маршрутизаторе, позволяет выполнить любую команду операционной системы WANFleX, а также некоторые предустановленные шаблоны.

Пример:

httpd enable

11.Rshd (Remote Shell)

Remote Shell сервер (RSH).

Синтаксис

rshd enable | disable RemoteUSER RemoteHOST LocalUSER

rshd start | stop | flush

Описание:

RSH сервер обеспечивает удалённое исполнение команд с помощью программы **rsh**.

Идентификация основана на использовании привилегированных TCP портов и списка разрешённых узлов.

По умолчанию сервер RSH не активен.

Запуск и остановка сервера выполняются с помощью команд **rshd start** и **rshd stop**. После запуска сервер не будет обслуживать запросы на исполнение команд, до тех пор, пока не будет определён хотя бы один разрешённый вход систему.

Вход в систему определяется командой **rshd enable** с тремя параметрами:

- **RemoteUSER** - имя удалённого пользователя (до 16 символов)
- **RemoteHOST** – IP-адрес удалённого узла
- **LocalUSER** - имя локального пользователя (до 16 символов).

Запрос на выполнение команды будет выполнен только в том случае, если все три параметра в запросе совпали с заданными. Может быть указано до 6-ти независимых входов. Имя локального пользователя никак не связано с основной системой авторизации WANFleX и может рассматриваться как ключевое слово.

Для отмены входа в систему выполняется команда **rshd disable** с теми же параметрами.

Команда **rshd flush** полностью очищает конфигурацию **rsh** сервера.

Сервер RSH удобно использовать для периодического снятия накопленной **статистики** с маршрутизатора

(`rsh -l mysecretuser RWR.domain.ru ipstat get`).

Пример:

rshd enable admin 195.38.44.1 mysecretuser

rshd enable root 195.38.45.123 mysecret2

rshd start

Copyright © 2004-2005 by InfiNet Wireless

12.Ipstat (модуль сбора IP-статистики)

Модуль IP статистики.

Синтаксис:

ipstat enable [items] | disable

ipstat clear | getclear

ipstat fixit | fixget | fixclear

ipstat strict | -strict

ipstat add [ifname] rule...

ipstat del N

ipstat traffic

Описание:

Модуль статистики позволяет накапливать информацию о проходящих через маршрутизатор потоках данных для последующего анализа и/или тарификации.

Накапливаемые данные хранятся в оперативной памяти маршрутизатора в виде записей с тремя полями: адрес источника, адрес приёмника и количество переданных байт.

По умолчанию, считаются только исходящие пакеты, в момент отправки через какой-либо физический интерфейс.

Число записей ограничено обьёмом имеющейся свободной оперативной памяти и может быть изменено командой "**ipstat enable** items", где "items" - число записей. По умолчанию число записей - 1000, этого количества обычно хватает на 15-20 минут на типовом абонентском блоке. Каждая запись занимает 12 байт.

Отображение накопленной информации выполняется на текущий терминал (или сеанс **rsh**), командами:

- **ipstat getclear** - показать и очистить накопленную статистику.
- **ipstat clear** - очистить накопленную статистику.

Более надёжный способ удалённого снятия статистики достигается следующими командами :

- **ipstat fixit** - делает дамп текущей статистики в промежуточный буфер. Текущая статистика обнуляется и начинает накапливаться снова.
- **ipstat fixget** - показывает содержимое сохранённого дампа статистики. Эту команду можно выполнять несколько раз и сколь угодно долго без потери текущей статистики.
- **ipstat fixclear** - очищает временный дамп статистики.

Листинг статистики содержит время с последней очистки таблицы, число использованных записей и их общее количество, сумму потерянных байт из-за переполнения таблицы и информацмию по каждой записи.

В случае переполнения таблицы записей или нехватке памяти, в системный журнал выводится соответствующее предупреждение и вновь поступающие данные не учитываются в статистике. Если указана опция **enable strict**, то, при возниковении такой ситуации, на устройстве отключается сквозная маршрутизация. При этом само устройство остаётся доступным по любому протоколу.

Copyright © 2004-2005 by InfiNet Wireless

Команда "**ipstat add** [ifname] rule..." позволяет ограничить число пакетов, попадающих в статистику, только теми, которые удовлетворяют правилу "rule". Дополнительно может быть указано имя интерфейса, через который пакет попадает в систему (ifname). Синтаксис правил, описывающих пакет, полностью соответствует синтаксису команды **ipfw**.

Команда "**ipstat del** N" удаляет правило N из списка.

Команда "**ipstat traf**fic [XX]" позволяет визуально посмотреть процесс накопления статистики в реальном времени. Отображается не более XX строк (по умолчанию 23), отсортированых по убыванию скорости потоков.

Примерный скрипт для надёжного снятия статистики с устройства с помощью **rsh**:

```
#!/usr/bin/perl -w
for(;;)
{
   my $stat;
   do
   {
      $stat = system("rsh -t 30 -n -l root IWR_IP
 ips fixit >/dev/null");
      if(int($stat) != 0) { sleep(5); }
   } while (int($stat) != 0);
   do
   {
      $stat = system("rsh -t 30 -n -l root IWR_IP
 ips fixget >stat.tmp");
      if(int($stat) != 0) { sleep(5); }
   } while (int($stat) != 0);
   do
   {
      $stat = system("rsh -t 30 -n -l root IWR_IP
 ips fixclear >/dev/null");
      if(int($stat) != 0) { sleep(5); }
   } while (int($stat) != 0);

   system("cat stat.tmp >>stat.txt");

   sleep(300);
}
```

Copyright © 2004-2005 by InfiNet Wireless

13.Acl (Access Control Lists)

Списки контроля доступа (Access Control Lists).

Синтаксис:

acl add $NAME TYPE params...

acl del $NAME [params...]

acl ren $NAME1 $NAME2

acl flush

Описание:

В практике сетевого планирования довольно часто возникает необходимость группировки некоторых однотипных параметров для использования их в качестве в качестве списка допустимых значений различных фильтров (например **ipfw**, **qm**, **ipstat**). Списки контроля доступа (ACL) позволяют эффективно решить эту задачу.

Команда **acl add** создаёт именованный список доступа NAME типа TYPE. Имена списков **должны** начинаться с символа **$** и могут содержать до 7 алфавитно-цифровых символов, исключая пробелы и символ "**;**" (точка с запятой). Одновременно в команде может быть указано несколько параметров типа TYPE, которые будут помещены в созданный список. Если список с указанным именем уже существовал ранее, то перечисленные параметры будут добавлены к уже имеющемуся списку.

Команда **acl del** позволяет удалить перечисленные параметры из списка NAME. Если не задано ни одного параметра, то будет удалён весь список.

Команда **acl ren**ame позволяет изменить имя списка с NAME1 на NAME2.

Команда **acl flush** удаляет все списки вместе с их содержимым.

Допустимые типы списков (TYPE):

net - содержит сетевые адреса в dot формате:

xxx.xxx.xxx.xxx или xxx.xxx.xxx.xxx/MASKLEN или

xxx.xxx.xxx.xxx/xxx.xxx.xxx.xxx
Список типа **net** оптимизирует сохраняемые параметры путём исключения дубликатов и поглощения более мелких сетей более крупными. Например, если в списке содержался параметр 1.1.1.1 , то включение в него сети 1.1.1.0/24 приведёт к исключению из списка адреса 1.1.1.1.

Примеры:

acl add $LIST1 net 10.0.0.0/8 192.168.0.0/16 5.5.5.5

acl del $LIST1 100.100.100.100/28

14.Sntp

Управление параметрами SNTP.

Наличие поддержки SNTP позволяет системе WANFlex синхронизировать время с указанным NTP сервером, используя четвёртую версию протокола SNTP **RFC 2030**.

Клиент работает в режиме **unicast** опроса сервера в заданные промежутки времени.

Синтаксис:

sntp [options] [command]

Команды:

Copyright © 2004-2005 by InfiNet Wireless

- **start** – запустить сервис
- **stop** - остановить сервис

Опции:

- -**server**={ipaddr} - задать адрес SNTP сервера
- -**interval**={seconds} – задать интервал опроса в секундах
- -**debug**={on|off} - включить/выключить отображение отладочной информации

Пример:

sntp -interval=3600 -debug=on

sntp -server=9.1.1.1 start

Команды:

- **start**

  Активизировать процесс синхронизации времени.

  Пример:

  sntp start

- **stop**

  Остановить процесс синхронизации времени.

  Пример:

  sntp stop

Опции:

Опции могут быть указаны в любой последовательности как вместе с командой так и без неё.

- **server**

  С помощью опции server необходимо указать IP-адрес Вашего NTP сервера.

  Пример:

  sntp -server=9.1.1.1

- **interval**

  С помощью опции interval задаётся промежуток времени в секундах, через который клиент будет опрашивать NTP сервер, по умолчанию имеет значение 3600.

  Пример:

  sntp -interval=5000

- **debug**

  Опция позволяет включить/выключить печать отладочной информации (пакетов) в журнал операционной системы WANFlex.

  Пример:

  sntp -debug=on

  sntp -debug=off

Copyright © 2004-2005 by InfiNet Wireless

15.Date

Управление датой и временем.

Команда показывает или установливает дату и время в системе WANFlex. При установке даты и времени изменяется не только системное время (kernel clock), но и hardware clock (если такая поддержка есть в конкретном устройстве).

Синтаксис:

date [[[[[cc]yy]mm]dd]HH]MM[.ss]]

cc	Век 19 или 20 добавляется перед годом
yy	Год в сокращённом виде (то есть 89 для 1989, 05 для 2005)
mm	Месяц в числовом виде от 1 до 12
dd	День, от 1 до 31
HH	Час, от 0 до 23
MM	Минуты, от 0 до 59
ss	Секунды, от 0 до 61 (59 плюс максимально две leap seconds)

Пример:

date 20040210053004

Tue Feb 10 05:30:04 2004

date

Tue Feb 10 05:30:10 2004

Copyright © 2004-2005 by InfiNet Wireless

III.Команды управления канальным звеном

1. Rfconfig (конфигурация радиоинтерфейса)

Команда служит для настройки радиоинтерфейса маршрутизатора

Синтаксис:

rf*config IFNAME [****sid*** *идентификатор] [****bitr*** *скорость] [****freq*** *частота] options*

Options:

- [[**-**]**burst**]
- [[**-**]**wocd**]
- [**pwr** *мощность*] [[-]**pwrctl**]
- [**mod**ulation **MOK** | **CCK** | **OFDM**]
- [**rts**threshold XXX]
- **band** {**full** | **half** | **quarter**}
- [**stat**istics]
- [**chntime** XXX]
- [**cap**abilities]
- [[**-**]**long**]
- [**dist**ance XXX | **auto**]
- [[**-**]**bcsid**]
- **txrt** XXX
- **txvrt** XXX
- **noise** XXXX

Не все радиоинтерфейсы имеют одинаковый набор параметров и опций, поскольку они зависят от типа и стандарта используемого радиомодуля. Полный список опций поддерживаемых конкретным интерфейсом можно посмотреть непосредственно на устройстве командой "rf rfNAME ?" . Тип радиомодуля и перечень его свойств можно посмотреть командой "rf rfNAME cap".

Описание:

- **sid** - идентификатор сети, число от 1H до FFFFFFH. Всем маршрутизаторам, которые должны видеть друг друга в радиоканале, необходимо иметь одинаковый идентификатор. Для совместимости с радио картами ARLAN 655, идентификатор должен быть чётным числом.
- **bitr** - битовая скорость в радиоканале, возможные значения: Для 2Mb интерфейса - 2000, 1000, 500, 354 бит в секунду. Для 11Mb интерфейса - 11000, 5500, 2000, 1000 бит в секунду.

  Для устройств на 5-6 ГГц: 6000, 9000, 12000, 24000, 36000, 48000, 54000 – в зависимости от модели устройства

  Если включен режим autobitrate протокола RMA, то для базовой станции параметр **bitr** будет задавать верхний предел для изменения скорости передачи в сторону абонента
- **freq** - рабочая частота, задается в мегагерцах, возможные значения:

Copyright © 2004-2005 by InfiNet Wireless

Для 2Mb интерфейса: 2412, 2427, 2442, 2457, 2465, [*2472, 2484*]. Для 11Mb интерфейса: 2412, 2417, 2422, 2427, 2432, 2437, 2442, 2447, 2452, 2457, 2462, [*2467, 2472, 2484*].

Для устройств 5-6 ГГц см. информацию на сайте **www.infinet.ru**

- **burst** включает поддержку протокола BURST. Этот режим позволяет существенно сократить количество коротких пакетов в эфире и уменьшить времена отклика при интенсивных потоках коротких пакетов, характерных, например, для игры Quake. Включение опции burst относится ко всему радио-интерфейсу и означает только то, что вы *хотите* использовать этот режим в данном устройстве. Реально протокол BURST будет работать только между теми устройствами которые используют протоколы RMA и BURST с обеих сторон. Включение этой опции не влияет на работу остальных устройств функционирующих в сети. Некоторую статистику использования протокола можно посмотреть по команде **muf stat**.
- **wocd** отключает контроль несущей при передаче. Может использоваться только на базовой станции совместно с **маркерным доступом**. Позволяет уменьшить число повторных пакетов и увеличить производительность соты. (*только для 2Mb и 54Mb радиомодулей*)
- **pwr** задаёт мощность излучения передатчика в милливаттах. Конкретные значения допустимых уровней мощности могут меняться в зависимости от типа установленного радиомодуля.

  Полный список уровней мощности поддерживаемых данным радиомодулем можно посмотреть командой "**rf rfX cap**abilities". В случае несовпадения введённого числа со списком допустимых значений, будет установлено ближайшее меньшее.
- **chtime**. Channel Burst Time. Время занятия канала в режиме **burst** рассылки, в микросекундах (0 ... 5000). По умолчанию 0 (когда 0, в конфигурации не отображается).
- **pwrctl** включает режим автоматического управления мощностью передатчика. Этот режим рекомендуется использовать на абонентских блоках с модулями Cisco 350 (с расширенным диапазоном управления мощностью), которые работают с базовой станцией, имеющей две антенны (RWR3513) и включенную опцию antenna diversity. В этом случае абонентский блок будет стремиться поддерживать оптимальную мощность передатчика (амплитуда 4-7 на стороне приёмника), обеспечивающую уверенный приём своих пакетов базовой станцией и исключающую ложное срабатывание алгоритма выбора антенны. Этот механизм **не будет правильно работать** при использовании внешнего усилителя.
- **modulation** устанавливает тип модуляции приёмо-передатчика (OFDM / CCK). Для устройств 5-6 ГГц доступна только модуляция OFDM.

  Начиная с версии 4.15 для устройств 2.4+ также доступен режим OFDM при котором поддерживаются битрейты 6, 9, 12, 18, 24, 36, 48 и 54 Мб/с.

  Устройства с разными типами модуляции не могут работать друг с другом.
- **rtsthreshold** определяет длину пакетов, больше которой будет использоваться механизм резервирования среды RTS/CTS стандарта 802.11. (по умолчанию 2048)
- **statistics** отображает на экран текущую статистику радиомодуля с интервалом в 1 секунду.

Copyright © 2004-2005 by InfiNet Wireless

Опция **stat1** выводит статистику только один раз без очистки экрана. Нижеприведенные таблицы описывают выводимые данной командой параметры

Вывод команды "**rf stat**" для устройств 5ГГц	
Параметр	**Значение**
Broadcast rate	текущее значение Bitrate для Broadcast и Multicast пакетов на базовой станции зависит от скорости подключения самого медленного абонента.
Voice Mode	Есть / нет Voice пакеты . Если есть, то автоматически включается режим их приоритетной обработки
Bytes Received	число принятых битов с учетом заголовков
Bytes Transmitted	число переданных битов с учетом заголовков
Packets Received OK	число правильно принятых пакетов
Packets Transmitted OK	число правильно переданных пакетов
Duplicate Received	число принятых дублей пакетов из-за издержек протокола
Total Retries	общее число переповторов
FIFO Overrun	число ситуаций переполнения очередей в радио при приеме
FIFO Underrun	число ситуаций опаздывания записи очередного пакета очередь в радио при передаче
CRC Errors	число принятых пакетов с нарушением контрольной суммы
Excessive Retries	число пакетов, которые не были переданы даже за максимальное количество перепередач (переповторов)
Noise Floor	уровень шума на входе, измеряется раз в 10 секунд
Noise Floor Threshold	установленный уровень для ситуации Carrier Detect

Вывод команды "**rf stat**" для устройств 2.4ГГц	
Параметр	**Описание**
BeaconsReceived	число принятых служебных пакетов 802.11 (не используется)
BeaconsTransmitted	число принятых служебных пакетов 802.11 (не используется)
AckPackets Transmitted	число переданных квитанций о приеме
RTS Packets Transmitted	число переданных пакетов RTS
CTS PacketsTransmitted	число переданных пакетов CTS
PLCP CRCErrors	счетчик ошибок контрольной суммы
Single Collisions	счеткик единичных коллизий

Copyright © 2004-2005 by InfiNet Wireless

PLCP FormatErrors	количество ошибок в формате PLCP
Polling cache aged	количество пакетов, у которых истекло время нахождения в кэше
PLCP LengthErrors	количство ошибок длины PLCP
NoDeferral	количество пакетов, отправленых без задержки
MAC CRCErrors	количество ошибок проверки контрольной суммы
Deferred Protocol	количество протокольных задержек передачи пакета
PartialReceived	число пакетов принятых неполностью
Deferred EnergyDetect	количество задержек передачи пакета по причине занятости среды
SSIDMismatches	количество ошибочных SSID (бесполезный параметр)
RetryLong	количество перепередач пакетов размер которых больше порога RTS
APMismatches	некорректные AP (бесполезный параметр)
RetryShort	количество перепередач пакетов размер которых меньше порога RTS
Data RateMismatches	количество ошибок битрейта
AuthenticationRejects	количество отвергнутых аутентификаций
AckReceived	количество принятых подтверждений на отправку пакетов
AuthenticationT/O	количство таймаутов аутентификации 802.11
No AckReceived	количество не подтвержденных пакетов при передаче
AssociationRejects	количство отвергнутых запросов на ассоциацию
CTSReceived	количтво принятых CTS пакетов
AssociationT/O	количество таймаутов ассоциации
No CTSReceived	количество не приняты CTS пакетов
PacketsAged	количество пакетов у которых превышено максимальное время жизни после приема

- **capabilities** отображает внутреннюю информацию о свойствах используемого радиомодуля, в том числе допустимые значения мощности передатчика, перечень поддерживаемых частот и т.д.
- **distance** - устанавливает точное значение расстояния между устройствами (в километрах). Этот параметр изменяет значения некоторых задержек и таймаутов протокола 802.11a/b/g, что делает возможным работу на повышенных дальностях с плавной регулировкой.

  Параметр **distance** работает только при отправке пакетов, поэтому он должен быть установлен на каждом устройстве сети.

  Существует несколько способов управления этим параметром:

Copyright © 2004-2005 by InfiNet Wireless

- Если установлено точное числовое значение, то именно это значение и будет использоваться независимо от способа подключения.
- Если на абонентском блоке вместо числового значения указано ключевое слово **auto** (установлено по умолчанию), то абонентский блок будет корректировать свои параметры по командам базовой станции. Достаточно установить числовое значение на базовой станции (задав его равным расстоянию до наиболее удалённого абонента), при этом остальные устройства автоматически скорректируют свою работу. При просмотре конфигурации, после параметра auto может показываться текущее установленное значение: **auto(XX).**
- Если для устройства известны точные географические координаты места установки (например по GPS), то можно указать эту информацию в команде **sys gpsxy**, а параметр **distance** установить в **auto** на всех устройствах, включая базовую станцию. В этом случае устройства будут автоматически корректировать свои настройки, подбирая оптимальное значение параметра **distance**. Базовая станция будет вычислять расстояние до наиболее удалённого абонента, а абонентский блок рассчитает собственное расстояние до базовой станции. Если абонентский блок имеет информацию о координатах линка, то он будет пользоваться этой информацией, иначе будет использоваться расстояние полученное с базовой станции).

  При установке значения 0 параметр distance используются настройки радиомодуля по умолчанию.

- **band** {**full** | **half** | **quarter**}. Данная опция устанавливает ширину полосы излучения: full – 20 MHz, half – 10 MHz, quarter – 5 MHz. Возможность переключения между различными ширинами полос определяется лицензией. В пределах одной беспроводной сети устройства должны иметь одинаковые настройки данного параметра.
- **long** включает режим LongRange, позволяющий устанавливать связь на дальних расстояниях.
  К сожалению, реализация протокола 802.11b в радиомодулях Aironet/Cisco рассчитана для использования на сравнительно небольших расстояниях и не позволяет устанавливать устойчивую связь на дальностях больше 20 км. Режим LongRange устраняет этот недостаток, полностью снимая зависимость от расстояния (в пределах прямой видимости).

  Для устройств 802.11b рекомендуется включение режима на расстояниях более 23-24 км.

  LongRange включается опцией **long**, выключается **-long**.

  Режим LongRange должен быть включен **на всех** устройствах сети. Если используется протокол RMA, то, при включении режима LongRange на базовой станции, все абоненты автоматически переключатся на работу в этом режиме и наоборот. Однако, не рекомендуется смешивать в одном сегменте сети устройства работающие в разных режимах. Максимальная скорость в режиме LongRange на 40-45% ниже, чем в стандартном режиме, поэтому использование режима LongRange оправдано в тех случаях, когда установить связь обычным способом не удаётся. LongRange работает на любых расстояниях, при этом скорость остаётся практически постоянной.

Copyright © 2004-2005 by InfiNet Wireless

- **bcsid** включает широковещательную рассылку идентификатора SID в служебных beacon пакетах, которые периодически рассылаются радиомодулем. Это свойство представляет собой потенциальную дыру в безопасности, поскольку разрешает устройству отвечать на запросы абонентских радиокарт с неустановленным SID (значение ANY). И хотя протоколы RMA и MINT исключают возможность несанкционированного подключения к сети, такое поведение устройства часто приводит к нестабильной работе, сбоям и задержкам в канале связи, особенно в условиях большого города с достаточным парком установленного радиооборудования и мультипровайдерности.
  Это безусловно большой недостаток протокола 802.11b, который, к тому же, был включен по умолчанию во всех радиомодулях Cisco/Aironet.
  Но теперь, начиная с версии 3.23, с этим покончено. Отныне широковещательная рассылка SID отключена по умолчанию. Помимо существенного повышения устойчивости сети отключение рассылки SID делает невозможным перехват идентификатора сети с помощью программ сканирования эфира типа NetStumbler, AirTraf, AiroPeak и т.п.
- **txrt** (transmit retries) задаёт максимальное количество переповторов при отправке unicast пакетов, по умолчанию 16
- **txvrt** (transmit voice retries) максимальное количество переповторов для пакетов данных (не голосовых) в голосовом режиме, по умолчанию 5. Голосовой режим (Voice Mode) включается автоматически при появлении трафика VoIP. Максимально допустимое значение - 64 переповтора.
- **noise** задаёт шумовой порог для радиоинтерфейса (Noise floor Threshold). Измеряется в Децибелах. По умолчанию 25 ДБ. Шумовой порог задаётся как положительное смещение относительно текущего уровня шума, который измеряется устройством. Устройство осуществляет передачу только в том случае, когда в эфире отсутствуют сигналы, уровень которых выше этого порогового значения. Посмотреть измеренный уровень шума (Noise floor) и установленный шумовой порог (Noise floor Threshold) можно командой "*rf IFNAME stat*".

Например:

rfconfig rf4.0 sid 01010101 bitr 24000 freq 5260 burst

2. RMA (конфигурация протокола RMA)

Команда предназначена для конфигурации протокола RMA.

Описание протокола

Каждый из маршрутизаторов может быть сконфигурирован для работы в режиме абонентского блока **AB** или базовой станции **BS**.

Каждый линк между **BS** и **AB** описывается как двухточечное соединение (подсеть состоящая из двух узлов).

Таким образом, с точки зрения внешней сети, каждый **AB** будет виден как подсеть с длиной маски 30 и информация о достижимости этой подсети и всех сетей подключенных к **AB** будет распространятся всеми **BS**, на которых он (**AB**) будет описан.

RMA предоставляет механизм, позволяющий переключать поток маршрутной информации так, чтобы в каждый момент времени она исходила только от той **BS**, к которой реально подключен данный **AB**.

Copyright © 2004-2005 by InfiNet Wireless

Доставка самой маршрутной информации выполняется с помощью любого подходящего протокола маршрутизации (в данном случае RIP-2).

Командный интерфейс к протоколу RMA:

rma ab ...	конфигурация базовой станции
rma bs ...	конфигурация абонентского блока
rma IFNAME [**-**]**noab** **rma** IFNAME [**-**]**nobs**	запрещает регистрацию абонентских блоков и/или поиск базовой станции на указанном интерфейсе.
rma IFname **test|autotest| autobs**	группа команд управления тестовыми режимами.
rma IFname **loamp=XX**	Минимальная амплитуда сигнала от абонентского блока, при которой базовая станция будет разрешать подключение, 0-16, по умолчанию 0
rma IFNAME [-]autobitrate	управление режимом автоматического управления скоростью передачи
rma IFname [**-**]**defroutebs** [**metric** N]	Устанавливает маршрут по умолчанию на адрес активной базовой станции.
rma start | stop | clear	управление процессом
rma IFname **poll|poll2|poll3|-poll**	включение/выключение маркерного доступа (Polling)
rma key XXX	Персональный секретный ключ устройства, используется для подписи служебных пакетов при авторизации на базовой станции (в том числе через RAPS) и сквозной шифрации передаваемых данных (опционально). Если этот параметр задан на абонентском блоке, то его должна знать и базовая станция (через RAPS или статически через rma ab key=XXX). Строка до 32 символов без пробелов.
rma bckey XXX	Ключ для подписи и шифрации широковещательных пакетов (опционально). Общий для всей сети. Строка до 32 символов без пробелов.
rma IFNAME [**-**]**crypt**	Включает режим сквозной шифрации данных (опционально, заказывается отдельно) (*если используется, то должна*

	быть включена на всех устройствах сети).
r**ma id** XXXX **rma server** IP[:PORT]	Команды службы разрешения удалённого доступа **RAPS**

Конфигурация абонентского блока

На абонентском блоке задаётся набор параметров базовых станций с которыми он потенциально может взаимодействовать.

Этот набор указывается ввиде пары значений (S,F) , где S -SID, F- частота.

Синтаксис:

rma bs *IFNAME SID|SPEED[/MINSPEED] FREQ[,FREQ...] [-**del**]*

где:

- **IFNAME** - имя используемого интерфейса
- **SID** - идентификатор сети.
- **SPEED** - требуемая скорость соединения в Kbit/sec (при включенном режиме autobitrate, этот параметр задаёт верхний разрешённый предел скорости соединения). Список поддерживаемых скоростей зависит от типа используемого радиомодуля и может быть получен командой: "rf rfX bitr"
- **MINSPEED** – опциональный параметр. Задает минимальное значение скорости, которое может быть установлено в режиме работы с autobitrate. Если данное значение не установлено в режиме autobitrate, то по умолчанию используется минимально возможное значение скорости для данного радиомодуля. Параметр задает минимальную скорость передачи как для абонента, так и для базовой станции
- **FREQ** - рабочая частота базовой станции. Перечень частот указывается через запятую, либо в виде интервалов `F1-F2`. (возможна комбинация этих вариантов).

Например:

rma bs rf4.0 020202/18000 5260,5340

rma bs rf3.0 050608/11000 2427,2442,2465

...

rma start

rip start

Будучи выполненной без дополнительных параметров, команда **rma bs** отображает состояние выбранной базовой станции, а также текущее качество соединения.

Конфигурация базовой станции

Чтобы перевести маршрутизатор в режим **BS**, достаточно на нём описать таблицу абонентских блоков, которым разрешено работать с данной базовой станцией.

Как уже было сказано, каждый абонентский блок описывается отдельной двухточечной подсетью (с длиной маски - 30).

Кроме того, должен быть известен MAC-адрес каждого абонентского блока.

Copyright © 2004-2005 by InfiNet Wireless

Синтаксис:

rma ab IFNAME MAC ip=MYIP [name=NAME] [key=KEY] [options]

rma ab IFNAME MAC [options]

rma abN IFNAME [options]

options: -disable | -enable | -del | -dumb | -nodumb

где:

- **IFNAME** - имя используемого интерфейса (rf0, rf1)
- **MAC** – MAC-адрес абонентского блока.
- **MYIP** – IP-адрес соединения со стороны базовой станции (длина маски - 30).
- **NAME** - Смысловое название абонента (до 20 символов без пробелов).
- Опции -dumb|-nodumb включают и выключают режим поддержки радиокарт не поддерживающих протокола RMA (только для модулей 802.11b). Если в качестве абонентского блока используется радиокарта не поддерживающая протокол RMA, то необходимо включить режим -dumb.

Например:

rma ab rf4.0 00:00:00:00:00:11 10.10.1.1 Abonent1

rma ab rf4.0 00:00:00:00:00:55 10.10.1.5 Abonent2

rma ab rf4.0 00:00:00:00:00:99 10.10.1.9 Abonent3

...

rma start

rip start

Будучи выполненной без дополнительных параметров, команда **rma ab** отображает состояние абонентских блоков, а также текущее качество соединения.

1	2	3	4	5	6	7
ab1	00028ae1d72a	NewRWR.0.1	9.9.0.1->2	11/8	A/B/P	<18/36>
ab2	00028ae1d795	NewRWR.0.5	9.9.0.5			

Описание выводимых полей:

№поля	Описание
1	Имя, присвоенное системой. Может быть использовано вмсесто MAC-адреса устройства в процессе конфигурации.
2	MAC-адрес абонента
3	Мнемоническое имя устройства
4	IP-адреса: (базовая станция)>(абонент). Адрес абонента автоматически включается в таблицу после успешной регистрации на базовой станции.

Copyright © 2004-2005 by InfiNet Wireless

5	Текущее качество соединения, измеренное по относительной шкале от 1 до 16. 1 – самый слабый сигнал, 16 – самый сильный. <сигнал от абонента>/<сигнал от Базовой Станции>
6	Поле статуса. A – активный абонент B – включен режим burst P – включен поллинг
7	<скорость на прием>/<скорость на передачу>

Тестовые режимы

Маршрутизатор имеет три тестовых режима.

1. Ручное тестирование канала по известному MAC-адресу. в этом режиме можно тестировать любое устройство непосредственно наблюдая результаты тестирования на консоли и на светодиодной индикации.

   Синтаксис:

   ***rma** IFNAME **test** MAC [LEN] [**mcast**]*

   Вместо MAC-адреса можно указывать ключевые слова **ab**N или **bs**, но такая форма команды будет работать, когда связь между объектами уже установлена (абонентский блок зарегистрирован на базовой станции).

2. Автоматическое тестирование канала по известному MAC-адресу. это режим отложенного выполнения. MAC-адрес блока подлежащего тестированию сохраняется в памяти устройства и тест выполняется сразу после включения питания.

   Синтаксис:

   ***rma** IFNAME **autotest** MAC [len] [**mcast**]*

   config save

3. Автоматическое тестирование канала с регистрацией на базовой станции. Аналогичен предыдущему режиму, но перед началом тестирования осуществляется поиск и регистрация на базовой станции.

   Синтаксис:

   ***rma** IFNAME **autobs** [len] [**mcast**]*

   config save

   Автоматические режимы отключаются с консоли командами:

   rma IFNAME -autotest

   config save

   Длина пакета по умолчанию 256 байт.

   Ключевое слово **mcast** позволяет выполнить тестирование широковещательными (broadcast) пакетами. В этом случае результаты

Copyright © 2004-2005 by InfiNet Wireless

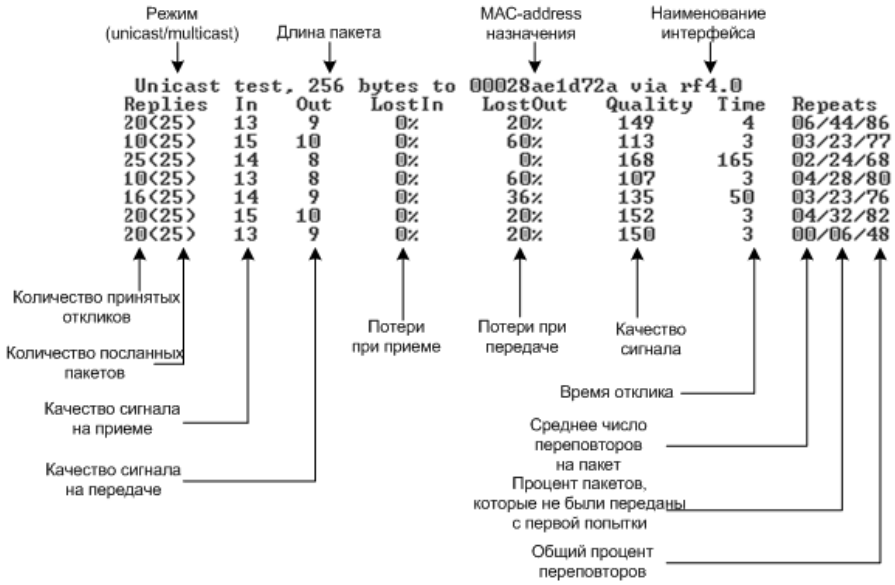
могут быть существенно хуже обычного режима, однако именно этот режим даёт наиболее точную оценку состояния линка.

Протокол тестирования доступен всегда, независимо от режима работы маршрутизатора (в том числе и от RMA), однако если абонентское устройство находится в режиме поиска базовой станции и постоянно меняет параметры радиомодуля, то ничего хорошего не получится. Поэтому на абонентском устройстве тест лучше запускать когда соединение с базовой станцией уже установлено, либо выключать протокол RMA (не забудьте проверить при этом текущие параметры радиомодуля).

Для получения более достоверных результатов не следует запускать более одного теста одновременно, особенно навстречу друг другу.

Автоматические режимы тестирования нужны только для изготовления автономных блоков при настройке антенн. Не забывайте отключать их в рабочей конфигурации устройства.

Приводим пример вывода команды **rma test**:



Управление процессом

rma start

rma stop

Включает/выключает поддержку протокола RMA.

Текущее состояние процесса можно посмотреть по команде **config show**.

rma clear

Полностью очищает всю конфигурацию RMA.

Автоматическое управление скоростью

rma IFname [-]autobitrate

Включает/отключает режим автоматического управления скоростью передачи.

Традиционные сценарии использования маршрутизаторов Revolution/InfiNet предполагают фиксированную скорость соединения между абонентским

Copyright © 2004-2005 by InfiNet Wireless

устройством и базовой станцией. Абонентский блок в процессе поиска и регистрации на базовой станции выбирает оптимальный режим из числа разрешённых в конфигурации и в дальнейшем не меняет параметров соединения даже если характеристики радиоканала (уровни сигнала, наличие помех) меняются со временем. Скорость передачи в обоих направлениях устанавливается одинаковой, той которую выбрало абонентское устройство. Если условия радиообмена ухудшаются настолько, что работа становится невозможной, соединение разрывается и производится поиск новой базовой станции или условий работы. Такая схема оправдана в тех случаях когда требуется получить надёжный канал с заранее известными характеристиками, параметры радиоканала тщательно спроектированы, произведены все необходимые расчёты, тестирование и пуско-наладочные мероприятия.

Однако, в некоторых случаях для быстрого включения в работу каналов с некритичными свойствами, например для бытового применения, бывает удобнее пожертвовать тщательностью планирования и довериться автоматике. Режим автоматического управления скоростью передачи решает эту задачу. В этом режиме каждое устройство независимо контролирует параметры соединения (амплитуда принимаемого от него сигнала, количество переповторов при передаче, ошибок, отношение сигнал/шум (SNR) на противоположной стороне и т.д.) и выбирает скорость передачи при которой выполняются необходимые условия для надёжной работы при минимуме переповторов и потерь. Скорость в каждом направлении может быть различной, но всегда оптимальной.

Если этот режим включен на абонентском устройстве, то базовая станция также автоматически включит его для обмена с данным абонентским блоком. В этом случае начальный поиск базовой станции выполняется на минимальной для конкретного типа радиомодулей скорости и потом поднимается до максимально возможного значения (ограничением являются также верхний предел скорости указанный в команде rma bs и ограничения конкретной модели устройства). Если установить режим autobitrate на базовой станции, то все абонентские устройства, подключенные к ней, автоматически перейдут на работу в этом режиме.

*Внимание: в реальных условиях производителем настоятельно рекомендуется включать режим **autobitrate** отдельно на **каждом** клиенте.*

Следует помнить, что включение режима autobitrate может снизить общую пропускную способность сети, поскольку средняя скорость передачи при интенсивной загрузке базовой станции будет стремиться к скорости самого медленного абонента. Это связано с полудуплексным характером протокола 802.11a/b/g и необходимостью обязательного квитирования передачи каждого пакета.

Режим работы с централизованной раздачей полномочий

***rma** IFname **poll**[**2**|**3**] [options]*

rma** IFname **-poll

Включает/выключает режим опроса.

rma IFname poll stat [clear]

Показывает статистику работы протокола.

Режим централизованной раздачи маркера (опроса) позволяет увеличить устойчивость и пропускную способность базовых станций в условиях большой нагрузки и дисбаланса в уровнях сигналов от абонентских блоков. Особенно полезен в тех случаях, когда абонентские блоки находятся на значительном удалении или вне прямой видимости друг от друга и не могут координировать свои действия, прослушивая среду передачи. Режим опроса позволяет наладить устойчивую связь нескольких абонентов в условиях когда метод случайного доступа CSMA/CA вообще не работает.

Несмотря на некоторое снижение абсолютной пиковой скорости передачи, этот режим существенно повышает общую пропускную способность базовой станции и обеспечивает равномерность потоков данных. Алгоритм разработан таким образом, чтобы минимизировать количество служебной информации, пересылаемой в эфире при сохранении высокой производительности и устойчивости. Интервал рассылки маркеров вычисляется индивидуально для каждого абонентского блока и динамически изменяется в процессе работы.

К недостаткам маркерного доступа в радиосреде можно отнести относительно большую начальную задержку и разброс времени отклика на коротких пакетах, что сказывается на критических к этим параметрам сервисов, таких как online-игры и телеметрия. Кроме того, две базовые станции с маркерным доступом, находящиеся в непосредственной близости и работающие на одинаковых или близких частотах, практически полностью подавляют работу друг друга.

Режим опроса включается только на базовой станции. На абонентских блоках никаких изменений в конфигурации не требуется. При этом на всех устройствах должен быть включен протокол RMA. Базовая станция с включенным режимом опроса может одновременно обслуживать и устройства не поддерживающие поллинг, хотя наличие последних будет сильно снижать эффективность работы системы.

Существует 3 варианта запуска режима опроса.

Первый вариант - "**rma** IFname **poll"**, представляет собой классический вариант с постоянной рассылкой маркеров без учёта загрузки канала. Для устройств серии 5000 с радиомодулями стандарта 802.11a этот режим наиболее предпочтителен.

Второй вариант - "**rma** IFname **poll2"**, учитывает реальную загрузку канала и динамически исключает малоактивные станции из цикла опроса.

Третий вариант - "**rma** IFname **poll3"**, анализирует количество проходящих через радиоинтерфейс пакетов и при превышении порогового значения включает алгоритм поллинга для всех станций (в обычном режиме). Этот режим наиболее полезен для решения проблемы скрытого узла, которая особенно сильно проявляется при интенсивных восходящих потоках (от абонентских блоков к базовой станции). При малой загрузке базовой станции работает обычный метод случайного доступа к среде CSMA/CA. При низкой входящей нагрузке этот режим позволяет обеспечить наилучшее время отклика и, несмотря на некоторое увеличение числа переповторов из-за коллизий, не приводит к снижению скорости. С ростом количества пакетов увеличивается и число переповторов, что приводит к снижению пропускной способности станции и подавлению одних узлов другими. Именно в такой ситуации, в момент критического перелома, автоматически включается режим поллинга и ситуация стабилизируется вплоть до предельных нагрузок.

Для точной настройки механизма поллинга существует ряд дополнительных параметров, которые можно задавать в форме:

rma IFname poll MI MP MT

(устаревший вариант)

Copyright © 2004-2005 by InfiNet Wireless

либо

*rma IFname poll[X] [**it**=XX] [**mi**=XX] [**ub**=XX] [**mt**=XX]*

Описание параметров:

- **IT - Input Threshold**

  Количество пакетов в секунду при превышении которого будет включен алгоритм поллинга. Только для режима poll3

- **MI - Marker Interval**

  Базовый интервал, используемый при вычислении частоты рассылки маркерных пакетов.

  Примерно соответствует половине минимального времени отклика на данном интерфейсе.

  Задаётся в миллисекундах в интервале от 4 до 20.

- **UB - Upper Bound**

  Верхний предел интервала рассылки маркеров. Обеспечивает гарантированный интервал, реже которого маркеры посылаться не будут. Выбирается исходя из компромисса между общим количеством маркеров напрасно загружающих канал и времени отклика на первое нажатие клавиши в программе telnet. Задаётся в миллисекундах в диапазоне от 50 до 1000. Значение по умолчанию: 800 мс.

- **MP - Minimal Priority (устаревший параметр, в новой реализации не используется)**

  Минимальный приоритет неактивного узла.

  В процессе работы, частота рассылки маркеров для каждого узла постоянно варьируется, в зависимости от его активности. Для неактивых узлов она постепенно понижается и повышается вновь только при появлении полезных данных для передачи.

  Произведение MP*MI определяет максимальный интервал рассылки маркеров для неактивного узла в миллисекундах и может находиться в пределах от 200 до 900.

- **MT - Marker Timeout**

  Максимальное время ожидания ответа на маркер или пакет данных в миллисекундах.

  И выполнении команды без дополнительных аргументов используются предустановленные значения по умолчанию. Не рекомендуется изменять эти параметры без особой необходимости.

  Для анализа эффективности работы протокола можно воспользоваться режимом статистики.

  По команде "**rma** IFname poll **stat**" будет отображаться следующая информация:

  MI=9, MP=60, MT=120 md=143234, ma=123212, mt=36

  pd=312762, pa=22343, pt=11

  где:

- **md** - количество маркеров, в ответ на которые получен пакет данных.
- **ma** - количество маркеров, в ответ на которые получен ответ "нет данных".

Copyright © 2004-2005 by InfiNet Wireless

- **mt** - количество маркеров, в ответ на которые не получено ответа (таймаут).
- **pd** - количество пакетов, в ответ на которые получен пакет данных.
- **pa** - количество пакетов, в ответ на которые получен ответ "нет данных".
- **pt** - количество пакетов, в ответ на которые не получено ответа (таймаут).

При нормальной работе, количество таймаутов не должно превышать 5% от общей суммы пакетов. То есть должно соблюдаться соотношение:

mt+pt < (md+ma+pd+pa)/20

Если число таймаутов превышает эту величину, то можно попробовать увеличивать значение параметра MT шагами по 20 мс не превышая разумного предела.

Изменение параметров и режима работы возможно в любой момент времени:

rma rf0 poll2 mt=120 mp=40; rma rf0 poll3 it=200

При отключении режима опроса устройство переходит в стандартный режим CSMA/CA, при этом возможна перерегистрация абонентских блоков.

Внимание! Режим опроса не может быть включен на устройстве работающем в режиме абонентского блока.

3. Muffer

Модуль контроля радиосреды

Синтаксис:

muf**fer [**-t**XX] [**-l**XX] IFname **review** [F1..F7] |**sid**|**mac**|**mac2**|**mynet**|**scan

***muf**fer **stat** [**clear**]*

Описание:

Модуль контроля радиосреды позволяет быстро проанализировать частотную обстановку в эфире, визуально оценить эффективность использования радиосреды, а также выявить наличие и мощность посторонних помех.

Несколько режимов работы модуля позволяют получить различную степень детализации анализа.

*На 11Mb радиомодуле все режимы анализатора (кроме **muf stat** и **muf mynet**) прерывают нормальную работу интерфейса. Кроме того, в этом случае анализируются только принимаемые пакеты (соответствующие стандарту 802.11b). Анализ помех от посторонних шумоподобных источников пока невозможен.*

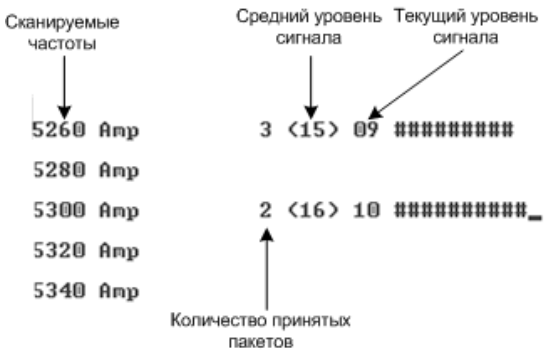
Режим общего обзора включается опцией review и позволяет оценить общую частотную и помеховую обстановку в эфире. В качестве дополнительных аргументов можно указать до 7 частот, подлежащих анализу (через пробел). По умолчанию будет отображаться информация по 7 стандартным частотным диапазонам: 2412, 2427, 2442, 2457, 2465, 2472, 2484

В данном режиме нормальная работа радиомодуля невозможна.

Например:

muffer rf4.0 review

Copyright © 2004-2005 by InfiNet Wireless



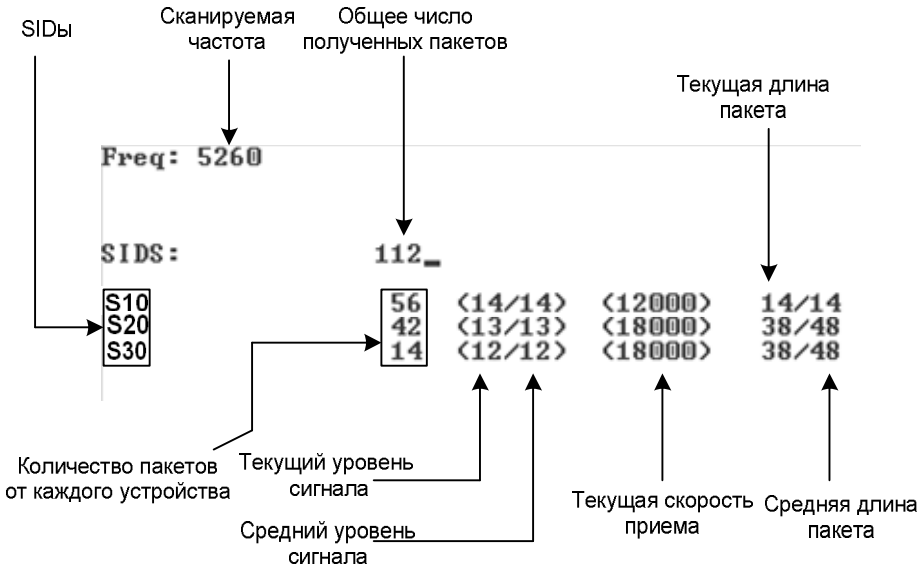
Режим анализа SID включается опцией **sid** и позволяет оценить количество и интенсивность работы в эфире групп абонентов с различными сетевыми идентификаторами (SID).

Анализ выполняется для всех сетевых идентификаторов на текущей частоте, установленной предварительно с помощью команды **rfconfig**.

Данный режим нарушает нормальную работу 11Mb радиомодуля.

Например:

muffer rf4.0 sid



Режим анализа MAC-адресов включается опцией "mac/mac2/mynet [mac]" и позволяет оценить количество и интенсивность работы в эфире абонентов с различными сетевыми адресами (MAC-адресами).

Анализ выполняется для всех сетевых идентификаторов на текущей частоте, установленной предварительно с помощью команды **rfconfig**. Режим mac отображает только пакеты данных, **mac2** учитывает все пакеты, включая протокольные квитанции MAC-уровня (если это возможно). Дополнительно можно указать конкретное значение MAC-адреса, по которому следует выполнять анализ.

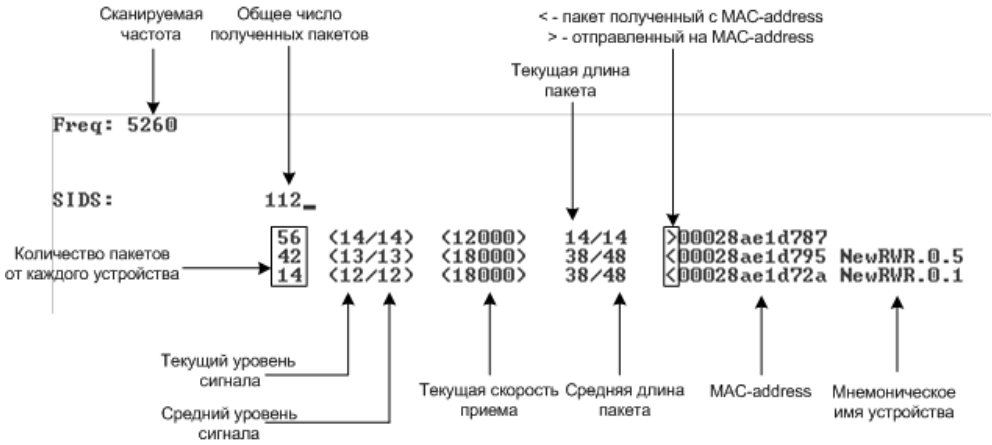
*Режимы **mac** и **mac2** нарушают нормальную работу 11Mb радиомодуля.*

Режим **mynet** позволяет выполнить анализ без нарушения работы радиомодуля, но только для пакетов принадлежащих данной сети.

Например:

Copyright © 2004-2005 by InfiNet Wireless

muffer rf4.0 mac2



Режим сканирования эфира включается командой **muf scan** и позволяет выполнить глубокий анализ сети на наличие источников радио сигналов. В этом режиме устройство сканирует эфир на всех частотах и режимах модуляции. Выводится информация обо всех неповторяющихся источниках сигналов.
Для получения возможно более полной информации этот процесс может потребовать значительного времени.

В данном режиме нормальная работа радиомодуля невозможна.

Например:

muffer rf4.0 scan

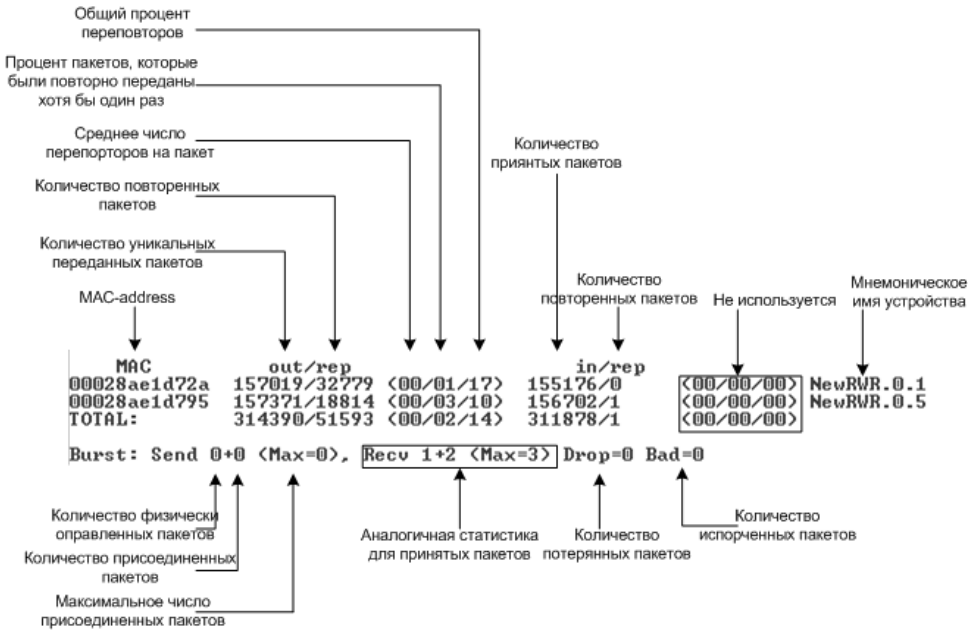
Дополнительные опции:

- **-t**XX **-** задаёт время работы в соответствующем режиме, в секундах. По умолчанию, все режимы включаются только на 2 минуты. Значение 0" полностью отменяет ограничение по времени.
- **-l**XX **-** Определяет размер экрана для отображения информации, в строках. По умолчанию - 24 строки.

Завершить работу анализатора можно нажатием клавиш ESC или Ctrl/C.

Режим статистики (отладочный) включается командой **muffer stat** и позволяет оценить интенсивность и эффективность загрузки канала с каждым абонентом. Для каждого MAC-адреса участвующего в обмене, показывается количество принятых и отправленных пакетов, а также количество повторных передач.

Copyright © 2004-2005 by InfiNet Wireless



Большое количество повторных передач, вызванное низким уровнем сигнала или большим количеством помех, приводит к бесполезной загрузке канала и снижению общей пропускной способности сети.

Команда **muffer stat clear** обнуляет значение всех счётчиков.

4. Macf

Фильтр соответствия MAC и IP-адресов

Синтаксис:

macf *MAC IP Comment*

macf del *N*

macf [-]dhcp *[-]**strict*** | *[-]**reverse*** | *[-]**simple*** *[-]**quiet***

macf show | ***clear***

Описание:

Команда **macf** позволяет задать жёсткое соответствие MAC и IP-адресов в ethernet сети.

Это может быть полезно сервис провайдерам, предоставляющим услуги подключения к сети независимой группе абонентов через один блок доступа, например частным лицам в жилом доме.

В этом случае, у абонентов часто появляется искушение изменить свой IP-адрес (на соседский) и, тем самым, обмануть учётную систему провайдера.

В общем случае, эта проблема почти неразрешима, однако, зафиксировав чёткое соответствие MAC адреса абонента и назначенного ему IP-адреса, можно существенно облегчить себе жизнь, поскольку процедура смены MAC-адреса намного сложнее.

Команда **macf MAC IP** добавляет в таблицу очередную пару соответствия.

macf 102030405060 1.1.1.1 Room123

macf 203040506070 2.2.2.2 Room125

Copyright © 2004-2005 by InfiNet Wireless

Посмотреть текущую таблицу соответствия можно командой **macf show** или **co show**:

macf show

macf	*1*	*0020af915099*	*192.78.64.99*	*Server*
macf	*2*	*0020af9150a3*	*192.78.64.194*	*Room94*
macf	*3*	*0020af9150a4*	*192.78.64.134*	*Room57*
macf	*4*	*0020af9150a7*	*192.78.64.174*	*Admin*

Внутренние номера присваиваются системой автоматически и могут быть использованы для удаления правил из таблицы, командой **macf del N**. Команда **macf clear** полностью очищает таблицу соответствия.

Запись сообщений в системный журнал можно отключить с помощью опции **quiet**.

Фильтр может работать в двух режимах.

В обычном режиме (по умолчанию) все станции, не описанные в таблице соответствия, будут обслуживаться без каких-либо ограничений. В строгом режиме (включается командой **macf strict**), пакеты от станций, не описанных в таблице соответствия, будут отбрасываться.

*Если вы конфигурируете маршрутизатор удалённо, через **telnet**, то, включая режим **strict**, убедитесь, что ваша рабочая станция уже внесена в таблицу соответствия. Иначе вы потеряете контроль над маршрутизатором и отменить этот режим можно будет только через диагностический порт.*

В любом случае, если пакет отбрасывается фильтром, то этот факт регистрируется сообщением системный журнал (**sys log**). Чтобы предотвратить лавину регистрационных пакетов, регистрируется только первая попытка нарушения из группы однотипных.

Алгоритм работы фильтра состоит из двух частей.

В нормальном режиме работы:

1. Фильтр выполняет поиск MAC-адреса в таблице

2. Если таковой обнаружен, то выполняется проверка соответствия IP-адресов.

Включение опции **reverse** изменяет порядок обработки на обратный:

1. Сначала ищется IP-адрес
2. Затем сравниваются MAC-адреса.

Параметр, определяемый в первом пункте, является ключевым и не может находится в таблице более одного раза

С включенной опцией **simple**, алгоритм выполняет только 1-ый пункт. Если поиск произведён удачно, то пакет будет принят. В противном случае пакет будет отброшен, независимо от значения опции **strict**. Значение второго параметра при этом не играет роли.

При включенной опции **dhcp**, фильтр macf автоматически дополняется адресами, выдаваемыми локальным DHCP сервером. Эти записи не сохраняются в постоянной конфигурации и работают только до тех пор, пока выданный адрес не будет удалён DHCP сервером.

Возможные сценарии использования фильтра:

1. Плоская модель. Все рабочие станции абонентской локальной сети подключены непосредственно к ethernet интерфейсу абонентского блока.

В этом случае можно использовать самый простой, классический вариант фильтра, возможно усиленный опцией **strict**:

Copyright © 2004-2005 by InfiNet Wireless

macf MAC IP [strict]

2. Если между абонентским блоком и локальной сетью стоит промежуточный маршрутизатор, то можно использовать вариант "reverse strict" или "reverse simple", перечислив все допустимые IP-адреса рабочих станций, а в качестве MAC поставить MAC-адрес промежуточного маршрутизатора.

3. Если к абонентскому блоку подключено несколько локальных сетей, отделённых промежуточными маршрутизаторами, то возможно будет полезным вариант "simple" или "reverse strict" с перечислением MAC адресов маршрутизаторов.

Примечание.
Во многих случаях, более удобным способом для решения этой задачи может быть использование команды **arp**.

5. Sppp

Конфигурация протокола PPP для синхронного канала

Команда предназначена для настройки протокола 'Point to Point Protocol' (PPP) для синхронного канала, **sppp** реализует внутреннюю логическую связь между сетевыми интерфейсами ppp0, ppp1 и портами котроллера E1 **0**(**main**), **1**(**sub**) соответственно.

Синтаксис:

sppp *[-v] ifname [{enable/disable}] [options]*

Описание

- **ifname** - имя сетевого интерфейса (ppp0 или ppp1).
- **enable/disable** - подключить/отключить PPP-интерфейс к порту контроллера E1.

Протокол SPPP может работать в следующих режимах:

- Cisco HDLC;
- PPP (Point to Point Protocol, RFC 1661);
- Frame Relay

Выбор и настройку режимов работы можно осуществить с помощью опций к команде **sppp**.

Пример:

sppp *ppp0 enable mode=cisco*

ifconfig *ppp0 10.0.0.1 10.0.0.2 up*

где адрес 10.0.0.1 - это адрес собственного PPP интерфейса, а 10.0.0.2 - адрес интерфейса удаленной стороны.

Опции:

- **mode**

С помощью опции **mode** указывается режим работы (протокол канального уровня) и может принимать следующие значения: **cisco**; **ppp**; **fr**.

- **cisco** - Cisco HDLC.
- **ppp** - Point to Point Protocol RFC 1661
- **fr** - Frame Relay

Пример:

sppp *ppp0 enable mode=cisco*

sppp *ppp0 enable mode=ppp*

sppp *ppp0 enable mode=fr*

- **keepalive**

Опция используется совместно с режимами **cisco** и **ppp**, может принимать значения: **on**; **off**.

Включает/выключает проверку состояния линии на удалённой стороне интерфейса при помощи рассылки управляющих пакетов с периодом в 10 секунд. Опция **keepalive** имеет смысл только в режимах **cisco** и **ppp**.

Пример:

sppp *ppp0 enable mode=cisco* ***keepalive****=on*

sppp *ppp0* ***keepalive****=off*

sppp *ppp0* ***keepalive****=on*

Опции авторизации режима ppp

Следующие опции используются только в режиме **ppp** для выбора протокола аутентификации и указания дополнительных параметров для выбранного протокола.

- **proto**

Устанавливает протокол аутентификации для обеих сторон линии (см. **myproto** или **hisproto**), может принимать значения: **none**; **pap**; **chap**.

  - **none -** отключить протокол аутентификации
  - **pap** - отключить протокол аутентификации. Password Authentication Protocol, протокол аутентификации пароля, осуществляется передача пары идентификатор/пароль (см. {**my**|**his**}**name** и {**my**|**his**}**secret**) пользователя между двумя устройствами в открытом виде.
  - **chap** - Challenge Handshake Authentication Protocol RFC 1994, протокол аутентификации с предварительным согласованием вызова; протокол аутентификации по методу "вызов-приветствие" (тип сигнальной аутентификации с помощью пары секретных кодов).

- **myproto**

Тоже самое что и **proto**, но только для своей стороны линии. То есть, для случая, когда удалённая сторона линии требует аутентификации.

- **hisproto**

То же, что и **proto**, но только для удалённой стороны линии.

- **myname**

Устанавливает имя пользователя для протокола аутентификации (**chap**; **pap**).

- **hisname**

Устанавливает имя удаленной системы (пользователя) для протокола аутентификации (**chap**; **pap**).

- **mysecret**

Устанавливает секрет (key, password) для использования во время фазы аутентификации. Для протокола **chap** используется для генерации хэш-

Copyright © 2004-2005 by InfiNet Wireless

значения в ответ на удалённый запрос. Для протокола **pap** передаётся в открытом виде совместно с именем системы (**myname**).

- **hissecret**

Тоже самое что и **mysecret**, но только для удаленной стороны.

Примеры:

sppp *ppp0 enable mode=**ppp** proto=**none***

Устанавливается режим **ppp** без аутентификации. В нижеследующем примере для интерфейса **ppp0** устанавливается режим аутентификации **chap** как для удалённой стороны, так и для своей, а для интерфейса ppp1 для удаленной стороны аутентификация не требуется .

sppp *ppp0 enable mode=**ppp** proto=**chap***

sppp *ppp0 myname=local mysecret=secret*

sppp *ppp0 hisname=remote hissecret=secret*

sppp *ppp1 enable mode=**ppp** myproto=chap hisproto=**none***

sppp *ppp1 myname=local mysecret=secret*

- **{no}rechallenge**

Включает/отключает режим повторной аутентификации уже в процессе работы. Отключение режима повторной аутентификации необходимо для некоторых неверных реализаций протокола **chap** (Conware CoNet, firmware 4.0.x).

Пример:

sppp ppp0 norechallenge

6. Arp

Управление таблицами ARP

Синтаксис:

arp view [IP]

arp add IP MAC|auto proxy

arp del IP|all [proxy]

arp [-]freeze

arp [-]proxyall [$ACL]

Описание:

Протокол **ARP** служит для взаимного преобразования адресов сетевого и MAC уровня.

Например, в сети Ethernet он позволяет преобразовать IP-адрес станции назначения в её 48-битный Ethernet адрес для передачи пакета по локальной сети.

В обычном случае протокол ARP работает автоматически, разрешая адреса по мере необходимости. Однако в некоторых ситуациях появляется необходимость ручной корректировки таблиц адресов в административных целях. Команда arp решает эту задачу.

Команда имеет несколько форм:

arp view [IP]

Copyright © 2004-2005 by InfiNet Wireless

Отображает ARP записи для адреса IP, либо все имеющиеся в системе, если адрес не указан.

arp add IP MAC [proxy]

arp add IP auto proxy

Добавляет запись в ARP таблицу. Адресу IP ставится в соответствие адрес MAC. Если указано ключевое слово **proxy**, то система будет анонсировать эту информацию в ответ на запросы других станций, выполняя роль ARP сервера (proxy), даже если этот адрес (IP) не является собственным адресом системы. В этом случае вместо MAC-адреса можно указать ключевое слово auto, при этом реально будет подставлен MAC-адрес интерфейса находящегося в той же подсети, что и адрес IP.

arp del IP|all [proxy]

Удаляет запись для IP-адреса, либо все записи в системе, если указано ключевое слово all. Если указан дополнительный параметр proxy, то будут удалены только те адреса, для которых система выполняла роль сервера (proxy).

arp [-]freeze

Позволяет "заморозить" таблицу ARP. Никакие другие записи более не будут автоматически вноситься в систему. Команда фиксирует только записи внесённые оператором и не действует на радио интерфейс с активным протоколом RMA или MINT. **Будьте осторожны при введении этой команды через telnet.**

arp [-]proxyall [$ACL]

В режиме **proxyall** устройство будет отвечать на все ARP запросы, если соответствующий IP-адрес назначения имеется в таблицах маршрутизации и доступен через интефейс отличный от источника запроса. То есть, если до пункта назначения есть маршрут, то устройство готово стать шлюзом для него.

В режиме **proxyall** можно указать список адресов/сетей **$ACL**, который ограничивает действие данной команды.

Пример:

arp add 10.10.10.10 00:11:22:33:44:55

arp add 192.168.5.1 5544332211 proxy

Copyright © 2004-2005 by InfiNet Wireless

IV.Команды управления сетевым звеном

1. Ifconfig (конфигурирование интерфейсов)

Конфигурация интерфейсов.

Синтаксис:

***ifc**onfig интерфейс [**inet** АДРЕС] [[-]**alias**]*

***ifc**onfig интерфейс [**up|down**] [**mtu** N] [**link0**|**link1**|**link2**]*

***ifc**onfig интерфейс **media** mediatype*

***ifc**onfig **vlan**X **vlan** TAG [-]**vlandev** IFPARENT*

***ifc**onfig -a*

Описание:

- **интерфейс** - имя интерфейса (имена всех интерфейсов можно увидеть, выполнив команду **ifconfig -a** или **netstat -i**).
- **АДРЕС** может задаваться в виде **адрес/количество бит в маске** или **адрес:маска** или **адрес**.

  Например:

  ifconfig eth0 inet 192.168.1.1/26

  ifconfig eth0 inet 192.168.1.1:255.255.255.192

  ifconfig eth0 inet 192.168.1.1

  Для активизации интерфейса после назначение адреса необходимо перевести интерфейс в состояние UP.
- Флаги **up**, **down** - включить, выключить интерфейс. Ограничения системы:
  - интерфейс **lo0** невозможно перевести в состояние **down**.
  - информация о состоянии радиоинтерфесов не сохраняется в конфигурации (после рестарта радиоинтерфейсы всегда находятся в состоянии **up**)

  Например:

  ifconfig eth0 up

  ifconfig eth0 1.1.1.1/24 up

  ifconfig rf0 down.
- Флаги **`link0`**, **`link1`**, **`link2`** (только для моделей 210, 5XX) - специальные флаги для интерфейсов с физической средой передачи Ethernet
  - **`link0`** - использовать физический интерфейс AUI
  - **`link1`** - интерфейс BNC
  - **`link2`** - интерфейс UTP

  Например:

  ifconfig eth0 inet 192.168.1.1 link0 up

  Если не указывать специальных флагов при конфигурации интерфейса типа eth, то физический интерфейс выбирается автоматически между BNC и UTP. Для того чтобы снять любой из специальных флагов необходимо указать его со знаком '**-**', например:

Copyright © 2004-2005 by InfiNet Wireless

ifconfig eth0 -link0.

- Параметр **media** позволяет указать свойства физического интерфейса eth0 10/100 (для серии 5000). Допустимые значения: **100BaseTX-fullduplex**, **100BaseTX-halfduplex**, **10BaseT-fullduplex**, **10BaseT-halfduplex**, **auto** по умолчанию**: auto**
- Для конфигурации интерфейса **vlanX** (VLAN IEEE 802.1q) используются опции **vlan** и **vlandev.**

  Параметр **vlan** задаёт VLAN тэг для данного интерфейса (1-4094). Параметр **vlandev** создаёт связь с физическим интерфейсом обслуживающим среду передачи, в данном случае **eth0**. **Например:**

  ifconfig vlan1 1.1.1.1/24 vlan 5 vlandev eth0 up

  или

  ifconfig vlan1 1.1.1.1/24 up

  ifconfig vlan1 vlan 5 vlandev eth0

  ifconfig -vlandev eth0

  (последняя строка в примере отменяет связь логического интерфейса **vlan1** с физическим устройством **eth0**)

  Оба дополнительных параметра интерфейса **vlanX** следует вводить вместе в одной строке, так как показано в примере, при необходимости объединяя вместе с назначением основного IP адреса. Для нормальной работы интерфейса **vlanX**, необходимо, чтобы физический интерфейс с которым он связан (eth0) был в активном состоянии (флаг **up**).

  По умолчанию, интерфейсы vlanX имеет MTU равное 1496, для того чтобы общая длина ethernet фрейма не превышала 1512 байт. Это сделано для совместимости с некоторыми ethernet картами, которые не поддерживают работу с нестандартными фреймами. Однако сами устройства RWR поддерживают расширенный фрейм ethernet и могут нормально работать с MTU 1500 (длина фрейма 1514). Для VLAN интерфейсов MTU по умолчанию равен 1500.

- Флаг **alias** необходимо ставить в том случае, если на один интерфейс нужно назначить несколько адресов. При этом первый адрес, назначенный на интерфейс, будет считаться основным (primary), а все остальные дополнительными (alias). Например, при выполнении команд:

  ifconfig eth0 inet 193.124.189.1/27 up

  ifconfig eth0 inet 10.0.0.1 alias

  на интерфейсе **eth0** будет одновременно два адреса из разных сетей.

  Удалить любой адрес с интерфейса можно с помощью опции **-alias**, указанной вслед за адресом подлежащим удалению, например:

  ifconfig eth0 inet 193.124.189.1/27 -alias

  Опцию [-]**alias** можно указывать в любой команде конфигурации интерфейса, т.е. все адреса, назначенные на интерфейс можно считать равноправными alias-ами.

Copyright © 2004-2005 by InfiNet Wireless

Если удаляется основной (primary) адрес, то в качестве основного будет подставлен следующий за ним alias.

- Параметр **mtu** N позволяет установить желаемый MTU (Maximum Transfer Unit) для интерфейса. Обычно нет необходимости менять значение этого параметра, но, в некоторых случаях, уменьшение значения MTU позволяет добиться устойчивой работы абонентов с **очень** низким уровнем сигнала. Кроме того, он может использоваться для изменения параметров туннельных интерфейсов.
- Если в качестве параметров задано только имя интерфейса, то команда покажет конфигурацию заданного интерфейса. Для просмотра конфигурации всех интерфейсов маршрутизатора необходимо выполнить команду **co show ifc**

2. Tun

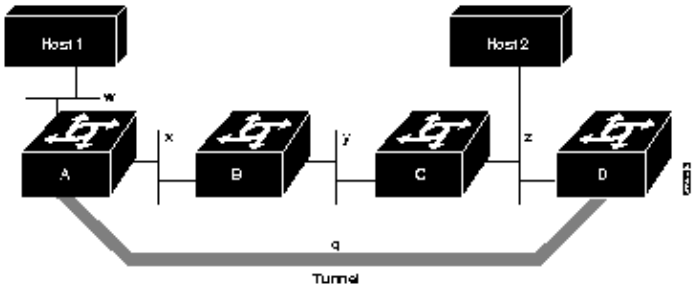
Конфигурация программного туннеля.

Синтаксис:

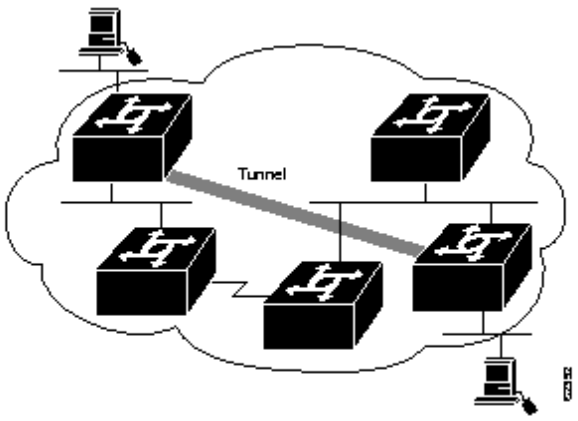
tun N src IP_ADDR dst IP_ADDR [mtu N] [clear].

Описание:

Туннель - это механизм позволяющий объединить две удалённые и не связанные физически сети в единую логическую структуру.



Туннели широко используются для создания корпоративных сетей или так называемых "виртуальных частных сетей" (VPN), когда несколько офисов, разнесённых территориально и, возможно, подключенных к сети через разных провайдеров, соединяются с центральным офисом или друг с другом туннелями, образуя, таким образом, единую корпоративную структуру. При этом, во всей корпоративной сети может использоваться собственное адресное пространство и учётная политика, не зависящая ни от места подключения ни от провайдера услуг.



Использование туннелей позволяет также решить проблему использования единой транспортной среды в сети коллективного доступа для предоставления услуг различным клиентам несколькими провайдерами. То есть независимо от места подключения клиента к единой транспортной сети, он может быть соединён туннелем и получать услуги от конкретного провайдера.

Существует несколько различных способов организации туннелей. В OS WANFleX реализован один из них - IP Encapsulation within IP, в соответствии с **RFC2003**.

Этот тип туннеля обеспечивается также маршрутизаторами Cisco и является подмножеством протокола IPSEC поддерживаемого многими операционными системами.

Практически, туннель реализуется в виде P2P линка между двумя маршрутизаторами.
Весь поток, проходящий через такой линк, инкапсулируется в IP пакеты и доставляется к конечной точке линка через уже существующую транспортную среду.

Для конфигурации туннеля необходимы 4 параметра:

- адрес локального конца P2P линка
- адрес удалённого конца P2P линка
- адрес источника, подставляемый в исходящих пакетах
- адрес получателя, подставляемый в исходящих пакетах

Адреса P2P линка устанавливаются с помощью команды **ifconfig** для интерфейса tunN:

ifconfig *tun0 1.1.1.1 1.1.1.2*

Все остальные параметры задаются с помощью команды **tun**:

tun *0* ***src*** *195.23.23.23* ***dst*** *194.34.34.34*

В данном примере конфигурируется туннель с адресами 1.1.1.1 -> 1.1.1.2 Исходящие пакеты будут упаковываться в IP-датаграммы и отправляться по адресу 194.34.34.34

При этом **важно**, чтобы адрес 194.34.34.34 был безусловно доступен через какой-либо другой интерфейс маршрутизатора, кроме самого туннеля. Возможно, для этого потребуется использование явной статической маршрутизации (route add) или запрет на импорт некоторых маршрутов поступающих через этот интерфейс по протоколу RIP.
Если это условие не выполняется, то произойдёт рекурсия, при которой уже упакованные пакеты будут снова попадать на вход туннеля и т.д, что может привести к перегрузке системы. Система отслеживает эту ситуацию и в случае возникновения такого цикла отбрасывает ошибочные пакеты и заносит сообщение в системный журнал:

tunX: looping...

В качестве адреса источника будет поставляться адрес 195.23.23.23 Адрес 195.23.23.23 **должен** быть действительным адресом одного из интерфейсов маршрутизатора и быть доступным со стороны удалённого конца P2P линка через имеющуюся сеть.

С обратной стороны P2P линка все настройки выглядят в точности наоборот:

ifconfig *tun0 1.1.1.2 1.1.1.1*

tun *0* ***src*** *194.34.34.34* ***dst*** *195.23.23.23*

Copyright © 2004-2005 by InfiNet Wireless

В случае использования на удалённом конце маршрутизатора Cisco, конфигурация может выглядеть следующим образом:

```
interface Tunnel0
 ip address 1.1.1.2 255.255.255.252
 tunnel source 195.34.34.34
 tunnel destination 195.23.23.23
 tunnel mode ipip
!
```

MTU (Maximum Transfer Unit) используемый по умолчанию - 1480, однако может быть изменён опцией **mtu N** команды **tun**.

3. Qm (конфигурация Quality-of-Service)

QoS менеджер.

Синтаксис:

```
qm option {[-]voice [-]dot1p [-]tos [-]strict}
qm classN if=ifname | max=N
qm chN [max=N] [pri=N] [pps=T] [to=ADDR] [vlan=N|-1|-2] [dscp=N|-1|-2]
       [dot1p=N|-1|-2] [classN] | clear qm stat [clear]
qm stat [clear]
qm del N
qm mov N M
qm add [ifname] chN rules...
```

Описание

QoS менеджер представляет собой удобный и гибкий механизм манипуляции потоками данных, проходящими через маршрутизатор. Концепция этого механизма заключается в следующем:
В системе существует несколько (200) программных каналов, каждый из которых может обладать некоторыми свойствами. С помощью специальных правил, можно заставить пакеты, проходящие через маршрутизатор, проходить через тот или иной канал и изменять, тем самым, свойства самого пакета или потока, которому он принадлежит. Количество свойств, которыми обладают каналы, будет расширяться по мере развития системы.

Команда "**qm class**N **if=**ifname | **max=**N" создаёт класс обслуживания и связывает его с интерфейсом ifname. Используется для динамического распределения пропускной способности интерфейса между различными каналами. Если используется опция **max=**N, то общая полоса пропускания класса будет ограничена заданной величиной (тысяч бит/сек), в противном случае для ограничения полосы будет использоваться текущая физическая скорость интерфейса. Отключить параметры можно так: "if=ifname max=0".

Команда "**qm chN [max=N] [pri=N] [pps=T] [to=ADDR] [vlan=N|-1|-2] [dscp=N|-1|-2] [dot1p=N|-1|-2] [classN] | clear**" описывает один программный канал.

Номер канала (1..200) задаётся параметром **ch**N.

Параметр **max=**N устанавливает максимальную скорость потока для данного канала (**тысяч бит/сек**). Может принимать значения от 10 до 10000. При установке в 0 (ноль) отменяет ограничение по скорости.

Copyright © 2004-2005 by InfiNet Wireless

Дополнительно можно указать параметр **class**N, привязав данный канал к классу обслуживания N.

В этом случае ограничение скорости будет нежёстким: если пропускная способность интерфейса, указанного для класса N, не используется полностью, то ограничение скорости канала может быть превышено, вплоть до полной загрузки интерфейса. Однако, при одновременной работе нескольких каналов, связанных с данным классом, они поделят всю полосу пропускания пропорционально их начальным ограничениям.

Параметр **pps=T** устанавливает максимальное количество пакетов в секунду для данного потока

Параметр **pri=**N устанавливает приоритет потока для данного канала (0..16). Меньшее числовое значение определяет больший приоритет.

Параметр **to=**ADDR позволяет перенаправить весь поток на другой адрес, вне зависимости от сложившихся условий маршрутизации. Адрес ADDR должен быть непосредственно достижим через один из интерфейсов маршрутизатора (без дополнительной маршрутизации). Эту опцию можно использовать, например, в случае использования маршрутизатора в качестве блока доступа к сети, когда два независимых клиента хотят пользоваться услугами разных провайдеров через один блок.

Параметры **vlan=N**, **dot1p=N**, **dscp=N** позволяют манипулировать метками DSCP и/или приоритета 802.1p. Значение "-2" удаляет параметр из командной строки, значение "-1" "сбрасывает" указанное поле:

- DSCP (допустимые значения 0-63) устанавливается в ноль.
- приоритет 802.1p (допустимые значения 0-7) обнуляется и, если VLAN ID не указан, удаляется вместе с заголовком VLAN.
- Vlan ID (допустимые значения 0-4095) удаляется вместе с заголовком VLAN независимо от наличия приоритета 802.1p.

Параметр **clear** снимает все установки и освобождает канал.

При одновременном указании нескольких параметров, порядок обработки следующий: сначала отрабатывается ограничение по скорости, затем переадресация, а потом приоритет. При одновременном указании параметров **vlan** и **dot1p**, сначала обрабатывается **vlan**, а затем **dot1p**.

Ключевые слова и значения параметров пишутся слитно, без обрамляющих пробелов вокруг знака "=".

Команда **qm stat** отображает статистику работы конкретного канала (только для каналов с ограничением по скорости):

```
qm ch1 max=128 cur=127 packets=12345 (1234) bytes=1234567 (12345)
```

числа в скобках отображают количество отброшенных единиц.

Команда "**qm add** [ifname] **ch**N rules..." позволяет подключить пакет, удовлетворяющий правилу "rule" к каналу N. Дополнительно может быть указано имя интерфейса, через который пакет попадает в систему (ifname). Параметр "**pass**" позволяет после выполнения команды продолжить просмотр других правил по списку. Синтаксис правил, описывающих пакет, полностью соответствует синтаксису команды **ipfw**.

Команда "**qm del N**" удаляет правило N из списка.

Команда "**qm mov N M**" перемещает правило N в позицию M.

Copyright © 2004-2005 by InfiNet Wireless

Номера правил можно посмотреть командой **config show**.

Следует иметь в виду, что во время работы каждый пакет, проходящий через систему, проверяется на соответствие правилам строго по порядку, от первого до последнего, пока не встретится правило удовлетворяющее свойствам пакета, либо до конца списка.

Порядок правил существенно важен для правильной работы, поэтому следует очень внимательно следить за этим.

На основе каналов qm менеджера осуществляется **сквозная приоритезация** в сети MINT.

Администратор имеет возможность направлять входящие потоки в подходящий канал на основе правил qm/ipfw, а также полей tos и dscp.

```
qm ch1 pri=6

qm add ch1 all from x/x to y/y

qm add ch1 dscp31 all from a to b

qm add ch1 dscp42
```

Каждому каналу может быть присвоен приоритет (0..16). Однажды назначенный приоритет будет автоматически обрабатываться каждым узлом при перемещении пакета внутри сети MINT. Схема распределения приоритетов выглядит так:

QM_PRIO_NETCRIT	0
QM_PRIO_VOICE	1
QM_PRIO_RT1	2
QM_PRIO_VIDEO	3
QM_PRIO_RT2	4
QM_PRIO_QOS1	5
QM_PRIO_QOS2	6
QM_PRIO_QOS3	7
QM_PRIO_QOS4	8
QM_PRIO_BUSINESS1	9
QM_PRIO_BUSINESS2	10
QM_PRIO_BUSINESS3	11
QM_PRIO_BUSINESS4	12
QM_PRIO_BUSINESS5	13
QM_PRIO_BUSINESS6	14
QM_PRIO_BUSINESS7	15
QM_PRIO_BUSINESS8	16

Приоритеты 1 и 2 подвергаются дополнительной обработке как голосовые. Пакеты, для которых приоритет явно не установлен, будут переданы через отдельную общую очередь в режиме best effort.

Команда «**qm option»** позволяет управлять автоматической приоритезацией потоков данных в устройстве. Опции команды **[-]voice [-]dot1p [-]tos** позволяют включить/выключить автоматическую приоритезацию голосовых пакетов, пакетов с меткой приоритета стандарта IEEE 802.1p (схема соответствия приоритетов MINT и стандарта IEEE 802.1p приведена ниже), пакетов с меткой TOS соответственно. Опция **[-]strict** означает, что для всех очередей будет применяться политика Strict Priority (

Copyright © 2004-2005 by InfiNet Wireless

т.е. пока более приоритетная очередь не пуста остальные очереди вообще не обрабатываются), в противном случае (по умолчанию) политика Weighted Fair Queuing (т.е. даже если более приоритетная очередь не пуста пакеты из других очередей будут обрабатываться в определенной последовательности относительно более приоритетной очереди. Например, 4 пакета из очереди с приоритетом 1 – 1 пакет из очереди с приоритетом 2, 8 пакетов из очереди с приоритетом 1 – 1 пакет из очереди с приоритетом 3).

Схема соответствия приоритетов MINT и стандарта IEEE 802.1p:

MINT	IEEE 802.1p
QM_PRIO_BUSINESS8	0 BE Best Effort
нет приоритета	1 BK Background
нет приоритета	2 Spare
QM_PRIO_BUSINESS1	3 EE Excellent Effort
QM_PRIO_QOS3	4 CL Controlled Load
QM_PRIO_VIDEO	5 VI Video
QM_PRIO_VOICE	6 VO Voice
QM_PRIO_NETCRIT	7 NC Network Control

Например, на узле настроена автоматическая приоритезация пакетов с меткой приоритета стандарта IEEE 802.1p. На узел приходит пакет с меткой приоритета IEEE 802.1p «6 VO Voice». Тогда узел присвоит ему приоритет «QM_PRIO_VOICE» и согласно схеме распределения приоритетов, этот пакет будет обработан раньше пакетов с другими приоритетами.

*Внимание: Реальная приоритезация внутри сети MINT осуществляется по приоритету, заданному параметром **pri=N**.*

Метка DSCP прозрачно передаётся через сеть MINT в любом режиме её работы.

Приоритет 802.1p прозрачно передаётся только при работе в режиме коммутатора.

*При необходимости, на выходе из сети MINT можно назначить требуемые параметры **dot1p** и **dscp**, по желанию оператора.*

QoS менеджер позволяет составлять достаточно гибкие правила по приоритезации и "перекраске" трафика (см. примеры ниже).

Примеры:

qm ch1 max=64

qm add eth0 ch1 all from 0/0 to 0/0

При выполнении на абонентском блоке позволяет ограничить весь исходящий трафик клиента величиной 64 Кбит/сек.

qm ch1 pri=10

qm add ch1 all from 1.1.1.0/24 to 0/0

qm add ch1 all from 0/0 to 1.1.1.0/24

Трафик сети 1.1.1.0/24 будет **более приоритетным** по отношению ко всем остальным потокам данных.

qm ch1 pri=10

qm ch2 pri=20

Copyright © 2004-2005 by InfiNet Wireless

qm add ch2 all from 1.1.1.0/24 to 0/0

qm add ch2 all from 0/0 to 1.1.1.0/24

qm add ch1 all from 0/0 to 0/0

Трафик сети 1.1.1.0/24 будет **менее приоритетным** по отношению к всем остальным потокам данных. *Обратите внимание на порядок правил. Последнее правило, которому удовлетворяет **каждый** пакет, должно стоять в конце списка.*

qm ch1 to=10.10.10.10

qm ch2 to=20.20.20.20

qm add ch1 all from 1.1.1.0/24 to 0/0

qm add ch2 all from 2.2.2.0/24 to 0/0

Абоненты сети 1.1.1.0/24 будут работать через провайдера 10.10.10.10, в то время как абоненты сети 2.2.2.0/24 будут работать через провайдера 20.20.20.20.
В более сложном случае, когда маршрутизаторы провайдеров не достижимы непосредственно с данного узла, следовало бы сначала организовать **туннели** до выбранных провайдеров, а уже потом выполнять переадресацию.

qm option –voice tos

Данная команда отключает автоматическую приоритезацию голоса и включает автоматическую приоритезацию tos.

Примеры приоритезации и "перекраски" трафика:

Канал 1 сбрасывает метки DSCP и приоритеты 802.1p

qm ch1 dscp=0 dot1p=-1

Канал 2 устанавливает приоритет потока QM_PRIO_BUSINESS1 и метку DSCP 31

qm ch2 pri=9 dscp=31

Канал 3 устанавливает приоритет потока QM_PRIO_VIDEO и метку DSCP 11

qm ch2 pri=3 dscp=11

Канал 4 устанавливает приоритет потока QM_PRIO_BUSINESS8 и метку DSCP 51

qm ch4 pri=16 dscp=51

Пропускаем весь трафик через канал 1, для обнуления всех приоритетов

qm add ch1 pass all from 0/0 to 0/0

Часть трафика помещаем в канал 2

qm add ch2 tcp from X.X.X.0/24 to 0/0

Другую часть трафика помещаем в канал 3

qm add ch3 udp from X.X.X.0/24 PORT to 0/0

Оставшийся трафик будет обрабатываться как не приоритетный или можно придать ему какой-нибудь приоритет по умолчанию например поместив его в канал 4

qm add ch4 all from 0/0 to 0/0

Copyright © 2004-2005 by InfiNet Wireless

Канал 25 устанавливает 802.1p приоритет пакета. Если заголовка VLAN еще не было, он будет добавлен автоматически

qm ch25 dot1p=5

Канал 26 устанавливает 802.1p приоритет и VLAN ID. Если заголовка VLAN еще не было, он будет добавлен автоматически

qm ch26 vlan=7 dot1p=4

Пакеты, выходящие из сети MINT через интерфейс eth0 и помеченные меткой DSCP 11, помещаем в канал 25

qm addout eth0 ch25 dscp11 from 0/0 to 0/0

Пакеты, выходящие из сети MINT через интерфейс eth0 и помеченные меткой DSCP 13, помещаем в канал 26

qm addout eth0 ch26 dscp13 from 0/0 to 0/0

4. Route (статические маршруты)

Статические таблицы маршрутов.

Синтаксис:

***rou**te **add** АДРЕС шлюз [**metric** XX] [-интерфейс}*

***rou**te **delete** АДРЕС [шлюз]*

Описание:

Команда **route** позволяет манипулировать содержимым системных таблиц маршрутизации.
В нормальном режиме, когда запущен **модуль динамической маршрутизации**, эта команда не нужна, однако в некоторых случаях позволяет добиться более точной, нестандартной настройки.

- **add** - добавление маршрута в таблицу
- **delete** - удаление маршрута из таблицы
- **адрес** - адрес сети назначения или адрес хоста
- **шлюз** - адрес маршрутизатора через который доступен **адрес**
- **metric** - метрика маршрута [0-15], для протокола RIP (по умолчанию - 0)
- **интерфейс** – маршрутизация через интерфейс. В качестве адреса шлюза должен быть использован любой адрес интерфейса, через который назначается маршрутизация

адрес может задаваться в виде **адрес_сети**/**количество_бит_в_маске** или **адрес_сети:маска** или **адрес_сети** или **default**.

Например:

route add default 195.38.44.129

route add 193.124.189.0/27 195.38.44.108

route add 193.124.189.0:255.255.255.224 195.38.44.108.

Все маршруты описанные командой **route add** являются "псевдостатическими". Это означает, что информация о маршруте будет немедленно помещена в конфигурацию и будет находиться там до тех пор

Copyright © 2004-2005 by InfiNet Wireless

пока её явно не удалят командой **route delete**, однако реально указанные маршруты будут устанавливаться в системные таблицы лишь тогда, когда появится интерфейс с адресом и сетевой маской, в пределах которой находится указанный адрес шлюза (**gateway**). При исчезновении интерфейса, указанные маршруты будут немедленно вычеркнуты из таблиц маршрутизации (но останутся в конфигурации).

5. Rip (модуль динамической маршрутизации)

Модуль маршрутизации RIP-1, RIP-2.

Модуль маршрутизации поддерживает две версии протокола RIP *(Routing Information Protocol)* - **RIP-1** и **RIP-2**

Настройка модуля осуществляется опциями команды **rip**.

Синтаксис языка управления приведён в таблице.

rip start|stop|restart|flush|[-]trace [LEVEL]|dump|[-]ridhosts|[-]keepstatic
rip IFNAME v1 v2 [-]in [-]out [-]v1in [-]v1out [-]v2in [-]v2out [-]ag [-]subag
rip IFNAME peer ADDR[/MASK | /MASKLEN] ... | del
rip [-]static NET[/MASK] GATEWAY

Экспортно/импортные фильтры:

rip [INTERFACE] [no]export |[no]import NET[/MASK | /MASKLEN] [exact] | all | default [[+|-]metric N] [pref N] ...
rip [INTERFACE] [no]export |[no]import NET[/MASK]|[/MASKLEN] del

rip start
rip stop
rip restart

Старт, остановка, перезапуск процесса. В случае сохранения конфигурации в энергонезависимую память (**co**nfig **save**), запоминается текущее состояние модуля маршрутизации.

- **rip flush**

  Сброс всех импортных и экспортных фильтров.

- **rip [-]trace [LEVEL]**

  Включение режима трассировки

  LEVEL задаёт уровень детализации отладочной информации

  1 - минимальный уровень трассировки событий

  2 - трассировка принимаемых/отправляемых пакетов

  3 - трассировка принимаемых/отправляемых пакетов и их содержимого

  4 - трассировка изменений в маршрутных таблицах ядра.

  По умолчанию, если уровень не задан явно, включается уровень 4.

- **rip dump**

  Показывает состояние внутренних таблиц маршрутов и интерфейсов модуля маршрутизации.

- **rip show**

Copyright © 2004-2005 by InfiNet Wireless

Отображает информацию о текущих параметрах настройки.

- **rip [-]ridhosts**

  Запрещает экспорт адресов локальных интерфейсов точка-точка, если есть более общий маршрут на ту же сеть через этот же интерфейс.

  Это частный случай агрегации и позволяет уменьшить число экспортирумых маршрутов.

- **rip [-]keepstatic**

  Опция сохранеиия статических маршрутов.

  Иногда бывает полезно задать некоторые маршруты статически с помощью команды **route add**. Это позволяет настроить маршрутизатор на режим "горячего старта", при котором он начнёт выполнять свои функции сразу после включения, когда ещё не полностью построены динамические таблицы маршрутизации.

  Однако, в этом случае, возникает вопрос - как быть с такими маршрутами, если после запуска модуля маршрутизации обнаруживается один или более реальных источников информации. Данная опция позволяет сохранять статические маршруты (по умолчанию) в таблицах, несмотря на наличие прочих источников.

  При этом экпортируемое значение метрики таких маршрутов будет равно 1.

  Если требуется иное значение, то нужно явно задать правило экспорта командой **rip export**.

  Если данный режим выключен (**rip -keepstatic**), то маршрут будет замещён реальным, полученным от динамического источника.

---

rip IFNAME v1 v2 [-]in [-]out [-]v1in [-]v1out [-]v2in [-]v2out

---

Группа опций управления версией протокола.

Позволяет явно задать версии используемых протоколов на вход и на выход, для каждого интерфейса в отдельности. По умолчанию включен RIP2 на вход и на выход и полностью выключен RIP1 (**rip IFNAME v2 -v1**)

---

rip IFNAME [-]ag [-]subag

Управление агрегатными маршрутами (по умолчанию выключено).

Включение агрегации (объединения) позволяет существенно уменьшить объём передаваемой по сети маршрутной информации.

При включенной опции **subag** модуль маршрутизации пытается объединить мелкие подсети, поступающие через один интерфейс от разных источников в более крупные блоки.

При включенной опции **ag,** то же самое делается по отношению к натуральным сетям класса "C".

Следует с осторожностью пользоваться этими опциями.

Например, лучше воздержаться от использования агрегации в кольцевых сетях, если не все узлы поддерживают этот режим или на каком-либо из них используется статическая маршрутизация, поскольку в этом случае может возникнуть ситуация когда одна и та же группа подсетей будет приходить по одному пути в агрегированном виде, а по другому в разобранном. Естественно, при выборе маршрута будет использован более частный путь, что не всегда правильно.

Напротив, использование возможностей агрегации крайне желательно на узлах включенных между двумя независимыми частями системы или стоящих на выходе во внешнюю сеть.

Copyright © 2004-2005 by InfiNet Wireless

Например, абонентский маршрутизатор, стоящий на границе локальной сети абонента и опорной сети провайдера, вполне может использовать агрегацию, если абоненту выделены все подсети одного блока.

rip IFNAME peer { ADDR[/MASK] ... | clear

С помощью этого фильтра можно ограничить количество узлов, с которыми происходит обмен маршрутной информацией через интерфейс **IFNAME**.

В качестве ограничивающего параметра задаётся диапазон адресов, в пределах которого могут находиться потенциальные партнёры.

Маршрутная информация будет отправляться только через те интерфейсы, адреса которых совпадают, включают или входят в заданный диапазон.

Принимаемая информация будет отбрасываться, если адрес отправителя не попадает в заданный диапазон.

Например:

rip rf0 peer 10.1.2.3 10.4.5.6 192.168.1.0/16

Ограничивает обмен маршрутной информацией узлами 10.1.2.3, 10.4.5.6 и всеми, попадающими в диапазон 192.168.1.0/16

rip [-]static NET[/MASK] GATEWAY

Эта команда упразднена начиная с версии 3.31. Вместо неё следует использовать команду **route add**. Все команды **rip static**, имеющиеся в конфигурации, будут автоматически преобразованы в **route add**.

Экспортные/импортные фильтры:

Экспортные и импортные фильтры позволяют ограничить поток маршрутной информации и/или несколько изменить свойства маршрутов.

Существует 3 группы фильтров:

- Фильтры для конкретных адресов интерфейсов.

  Параметр **INTERFACE** имеет вид "**int:ADDR**"

  Таким образом можно задать фильтр для любого конкретного адреса интерфейса, если на интерфейс назначено несколько адресов (алиасов). **Например:**

  rip int:10.2.3.4 export all

- Фильтры для всего интерфейса в целом.

  Параметр **INTERFACE** задаётся символьным именем интерфейса. Фильтр распространяется на весь интерфейс целиком, на зависимо от числа назначенных адресов.

  Например:

  rip eth0 export all

- Фильтры для всего протокола в целом.

  Параметр **INTERFACE** на указывается. Фильтр действует на весь протокол целиком, на всех интерфейсах.

  Например:

  rip export all

Таблицы **EXPORT** и **NOEXPORT** используются для явного указания сетей, которые должны или не должны экспортироваться от имени данного

Copyright © 2004-2005 by InfiNet Wireless

маршрутизатора.
Таблицы IMPORT и NOIMPORT соответственно используются для указания сетей, которые должны или не должны импортироваться во внутренние таблицы маршрутизатора.

При составлении фильтров следует помнить следующее:

- Фильтры просматриваются в порядке от более частного к более общему.
- Сначала фильтры, связанные с конкретными адресами, потом с интерфейсами, и наконец общие.
- Отдельные правила в таблицах упорядочиваются по такому же принципу, от мелких сетей к более крупным, от детальной информации к более общей.
- По умолчанию, когда не задано никаких фильтров, принимаются и отправляются все маршруты с их натуральными метриками.
- Если включен хотя бы один запрещающий фильтр, то считается, что всё остальное разрешено.
- Если включен хотя бы один разрешающий фильтр, то считается, что всё остальное запрещено.
- Если вы начали разрешать что-то, то нужно разрешать всё, что необходимо, до конца.
- С другой стороны, если вы запретили что-то, то только это и будет запрещено.
- Если для одной и той-же сети указаны одновременно и запрещающий и разрешающий фильтры, то запрещающий фильтр будет иметь приоритет.
- К каждой сети/подсети будут применены фильтры только одной группы, имеющей наивысший приоритет.

Синтаксис описания фильтров достаточно прост, поэтому лучше рассмотреть его на примерах.

rip noexport 192.168.9.0/24

rip noexport default

rip export all

Запрещает экспорт конкретной частной сети, а также маршрута по умолчанию (default), но разрешает экспорт всей остальной информации.

rip noimport 10.0.0.0/255.0.0.0

rip import all

Запрещает импорт частных сетей из блока 10.0.0.0/8, но разрешает всё остальное.

Можно указывать в одной строке сразу несколько сетей:

rip noexport 192.168.9.0/24 192.168.10.0/24 192.168.20.0/24

Можно задавать явное значение метрики, устанавливаемой для маршрута при импорте или экспорте:

***rip import** 192.168.9.0/24 **metric** 5*

***rip export** 192.168.9.0/24 **metric** 7*

Copyright © 2004-2005 by InfiNet Wireless

Более того, можно указывать относительное изменение метрики маршрута при прохождении данного узла.

rip export all +metric 2

Метрики всех маршрутов будут увеличены на 2 по отношению к их оригинальным значениям.

rip import default -metric 1

При относительном изменении метрики, значение её никогда не будет меньше 2 и больше 13.

Обычно сетевой адрес, задаваемый в фильтре, описывает сеть со всеми вложенными подсетями (с более длинной маской). Если требуется точное соответствие адреса в фильтре, то следует добавлять ключевое слово **exact**.

Например, фильтр:

rip noimport *10.0.0.0/255.0.0.0*

запрещает импорт сети 10.0.0.0, а также всех менее крупных сетей (10.XXX.XXX.XXX).
В то же время, фильтр:

rip noimport *10.0.0.0/255.0.0.0* ***exact***

запретит импорт только одной сети 10.0.0.0, но разрешит обработку вложенных менее крупных сетей.

Для удаления фильтра нужно добавить ключевое слово **del** после указания сетевой информации.

rip noexport *192.168.9.0/24* ***del***

6. OSPFv2 (модуль протокола динамической маршрутизации)

Введение

Протокол OSPF является стандартным протоколом маршрутизации для использования в сетях IP. Основные принципы организации современной версии протокола маршрутизации OSPF изложены в RFC 2328. Протокол OSPF представляет собой классический протокол маршрутизации класса Link–State, который обеспечивает:

- отсутствие ограничений на размер сети
- поддержку внеклассовых сетей
- передачу обновлений маршрутов с использованием адресов типа multicast
- достаточно высокую скорость установления маршрута

Copyright © 2004-2005 by InfiNet Wireless

- использование процедуры authentication при передаче и получении обновлений маршрутов

Командный язык. Общие принципы

OSPF имеет свою командную оболочку (КО), вход в которую осуществляется командой:

```
#1> ospf
OSPF>
```

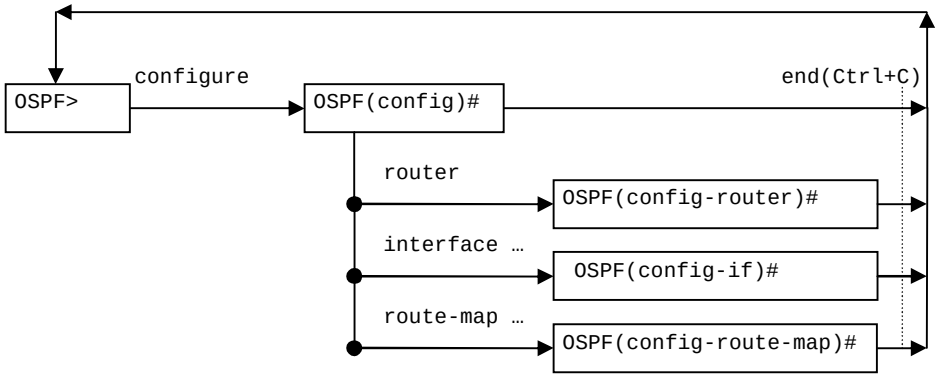
Команды, вводимые в КО, нечувствительны к регистру букв и допускают произвольное сокращение вплоть до одной буквы, если команда в целом не может быть воспринята КО неоднозначно. В любой момент можно получить справку о наборе команд. Для этого нажмите клавишу '?':

```
OSPF>?
  configure  Configuration from vty interface
  end        End current mode and change to root mode (CTRL+C).
  exit       Back to WANFlex command shell (CTRL+D).
  help       Print command list
  show       Show running system information
OSPF>
```

КО может находится в различных режимах. Режим отображается вместе с префиксом команды в виде OSPF(режим)#. Например, если ввести команду configure, то КО перейдет в режим *config*:

```
OSPF> configure
OSPF(config)#
```

На рисунке показана диаграмма переходов между режимами КО.



В каждом режиме существует собственный набор команд, однако в любом режиме существуют команды:

- Help - вывод полного списка команд данного режима
- End – переход из текущего режима в базовый
- Exit – команда выхода из КО в командную оболочку WANFlex

Изначально КО находится в базовом режиме, который содержит набор команд только для просмотра текущего состояния маршрутизатора. Переход в режим конфигурации разрешен только для пользователя WANFlex

Copyright © 2004-2005 by InfiNet Wireless

имеющего привилегии *superuser*. После перехода в режим конфигурации, конфигурация блокируется и вход в этот режим с другого терминала (например, другая сессия telnet) запрещен. Для исключения «мертвой» блокировки, КО автоматически выходит из режимов конфигурации, если в течение 5-ти минут не поступало никаких команд.

При вводе команд, в любой момент можно получить контекстную подсказку нажав клавишу "**?**". Например:

```
OSPF> config
OSPF(config)#?
  access-list  Add an access list entry
  clear        Reset functions
  end          End current mode and change to root mode (CTRL+C).
  exit         Back to WANFlex command shell (CTRL+D).
  help         Print command list
  interface    Select an interface to configure
  no           Negate a command or set its defaults
  prefix-list  Build a prefix list
  route-map    Create route-map or enter route-map command mode
  router       Enable a routing process
  show         Show running system information
  stop         stop
OSPF(config)# interface?
  IFNAME  Interface's name
OSPF(config)# interface eth0
OSPF(config-if)#?
  authentication       Enable authentication on this interface
  authentication-key   Authentication password (key)
  cost                 Interface cost
  dead-interval        Interval after which a neighbor is declared dead
  description          Interface specific description
  end                  End current mode and change to root mode (CTRL+C).
  exit                 Back to WANFlex command shell (CTRL+D).
  hello-interval       Time between HELLO packets
  help                 Print command list
  message-digest-key   Message digest authentication password (key)
  network              Network type
  no                   Negate a command or set its defaults
  priority             Router priority

  retransmit-interval  Time between retransmitting lost link state
  show                 Show running system information
  transmit-delay       Link state transmit delay
OSPF(config-if)#
```

После выхода из КО командой exit (или Ctrl+D) КО маршрутизатора остается в последнем режиме.

Команды могут иметь параметры. Параметры команд задаются в нескольких форматах. Формат параметра описывается в контекстной справке или в списке команд (по команде help) следующим образом:

Copyright © 2004-2005 by InfiNet Wireless

- A.B.C.D - параметр задается в формате IP адреса. Например: 192.168.0.15
- WORD – набор символов не содержащий пробелов
- <1-N> - параметр задается в виде целого десятичного числа в диапазоне от 1 до N
- A.B.C.D/M – параметр задается в форматке IP адрес/длина маски сети – целое число в диапазоне от 0 до 32. Например: 192.168.0.0/24
- IFNAME – имя физического сетевого интерфейса. Например: eth0

Если параметр допускает варианты, то в справке они указываются в круглых скобках, а возможные варианты разделены символом "|". Например: **(A.B.C.D|<0-4294967295>).**

При описании команд в случае, если какой-либо параметр или опция является необязательной, этот параметр или опция будет заключена в квадратные скобки "**[]**".

Любая команда конфигурации может содержат префикс "**no**". Наличие такого префикса приводит к отмене (удалению) соответствующего параметра конфигурации.

Запуск/останов OSPF маршрутизатора.

Запуск OSPF маршрутизатора осуществляется командой WANFlex:

ospf start

Останов маршрутизатора производится в командой режима *config*:

stop (daemon|clear)

Например:

>ospf

OSPF> configure

OSPF(config)# stop daemon

Если команду останова выполнить с параметром *clear*, то маршрутизатор перед остановом очистит свою часть системной конфигурации.

Идентификатор маршрутизатора

Любой маршрутизатор OSPF системы имеет уникальный идентификатор. Идентификатор – это целое 32-битовое число. Для того, чтобы присвоить идентификатор необходимо в режиме *config-router* выполнить команду:

router-id A.B.C.D

Например:

OSPF>configure

OSPF(config)# router

OSPF(config-router)# ospf router-id 195.38.45.107

OSPF(config-router)#

Copyright © 2004-2005 by InfiNet Wireless

Если идентификатор не был назначен администратором, то маршрутизатор автоматически будет назначать себе идентификатор равный максимальному (по значению) IP адресу из всех адресов маршрутизатора, участвующих в OSPF.

Отменить назначение идентификатора можно командой:

no router-id

Объекты фильтрации

Во многих настройках маршрутизатора применяются специальные объекты – фильтры. Фильтры представлены объектами двух классов

- Списки доступа (access-list)
- Списки префиксов (prefix-list)

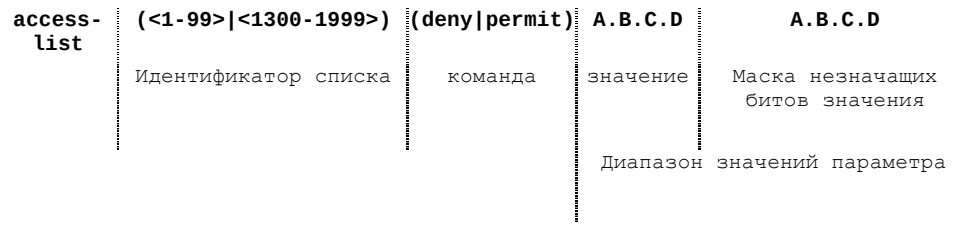
Списки доступа состоят из набора операторов. Каждый оператор состоит из диапазона IP адресов (условия соответствия) и команды *deny* (отвергнуть) или *permit* (принять). Диапазон адресов записывается в виде <значение> < маска незначащих битов значения>. Объект подлежащий фильтрации имеет основной параметр в таком же виде (IP адрес, подсеть и т.п.). Для принятия решения о том удовлетворяет ли объект данному списку или нет, к основному параметру объекта последовательно применяется каждый оператор списка доступа до тех пор, пока этот параметр не удовлетворит условию соответствия. При достижении соответствия выносится решение записанное в команде данного оператора (*deny* или *permit*).

В OSPF маршрутизаторе применяется три типа списков доступа:

- Стандартный. Идентифицируется номерами 1-99 или 1300-1999 и предназначен для оценки одного параметра объектов фильтрации.
- Расширенный. Идентифицируется номерами 100-199 или 2000-2699 и предназначен для оценки двух параметров объектов фильтрации (например, адрес источника и адрес назначения).
- Именованный. Аналогичен стандартному, но идентифицируется не номерами, а именем. Кроме того, диапазон (условие) операторов записывается в виде <значение>/<длина маски>

Для создания или изменения списков доступа в OSPF маршрутизаторе предназначены следующие команды режима *config:*

1. Для стандартных списков доступа



Данная команда создает оператор в стандартном списке доступа. Значение и маска незначащих битов определяют диапазон (условие) оператора. Маска незначащих битов определяет те биты значения, которые определяют диапазон. Например, чтобы задать диапазон IP адресов от 192.168.12.0 до 192.168.12.255, необходимо указать значение 192.168.12.0, маску – 0.0.0.255. Для диапазонов обозначающих любое значение параметра (0.0.0.0 255.255.255.255) предусмотрено ключевое слово *any*. Например, команда :

OSPF(config)# access-list 1 permit 0.0.0.0 255.255.255.255

Copyright © 2004-2005 by InfiNet Wireless

Тождественна команде:

OSPF(config)# access-list 1 permit any

Аналогично, для задания диапазона, состоящего из единственного значения, допускается применять ключевое слово *host*.

Например, команда :

OSPF(config)# access-list 1 permit 192.168.12.150 0.0.0.0

Тождественна команде:

OSPF(config)# access-list 1 permit host 192.168.12.150

2. Для расширенных списков доступа

access-list	**(<100-199>|<2000-2699>)**	**(deny|permit)**	**ip**	**A.B.C.D A.B.C.D**	**A.B.C.D A.B.C.D**
	Идентификатор списка	команда		Диапазон адресов источника	Диапазон адресов назначения

3. Для именованных списков

access-list	**WORD**	**(deny|permit)**	**A.B.C.D/M**	**[exact-match]**
	Идентификатор списка	Команда	Диапазон	Требование полного равенства параметра диапазону

В этом случае идентификатор списка представляет собой произвольное сочетание символов. Диапазон значений указывается в виде <значение>/<длина маски>. Например, чтобы задать диапазон IP адресов от 192.168.12.0 до 192.168.12.255, необходимо указать значение 192.168.12.0/24. Для диапазона любых значений параметра (0.0.0.0/0) так же допускается применять ключевое слово *any*. Например:

OSPF(config)# access-list TestList1 deny 192.168.1.0/24

OSPF(config)# access-list TestList1 permit any

При конфигурировании операторы списка безусловно добавляются в конец списка. Т.о. последовательность просмотра операторов списка при фильтрации определяется только последовательностью конфигурирования этих операторов.

Списки префиксов отличаются от списков доступа тем, что каждый оператор имеет кроме диапазона (условие) и команды еще и последовательный номер. Кроме того, при определении принадлежит ли параметр объекта фильтрации диапазону оператора, можно ввести дополнительные условия на длину маски параметра. Операторы списка префиксов сортируются по последовательным номерам операторов. Т.о. можно управлять последовательностью просмотра операторов при фильтрации объектов.

prefix-list	**WORD**	**[seq <1-4294967295>]**	**(deny|permit)**	**A.B.C.D/M**	**[ge <0-32>] [le <0-32>]**
	Идентификатор списка	Последовательный номер оператора	Команда	Диапазон	Диапазон допустимых длин маски параметра

Если не указан последовательный номер, то маршрутизатор назначает его автоматически путем прибавления 5 к последовательному номеру последнего оператора в списке. Т.о. новый оператор будет иметь самый большой номер и, соответственно, будет расположен в самом конце списка.

Copyright © 2004-2005 by InfiNet Wireless

Анонсирование каналов

Маршрутизатор может анонсировать в OSPF систему свои каналы двух видов:

1. *Внутренние каналы*. Это каналы, адресатами которых является одна из подсетей, к которой маршрутизатор подключен непосредственно (собственным сетевым интерфейсом) и описанная в конфигурации OSPF маршрутизатора.
2. *Внешние каналы*. Это каналы, адресатами которых являются адресаты маршрутов, прописанных в ядре ОС. Такими маршрутами могут быть статические маршруты, прописанные в системную таблицу маршрутов администратором ОС WANFlex командой **route add (kernel).** Либо маршруты, образующиеся в системной маршрутной таблице при присваивании физическому сетевому интерфейсу IP адреса (алиаса) (**connected**).

Для того, чтобы OSPF маршрутизатор анонсировал внутренний канал, необходимо указать подсеть, адресатом которой является такой канал. Это можно сделать в режиме *config-router* командой:

network A.B.C.D/M area (A.B.C.D|<0-4294967295>)

Номер сети задается IP адресом маршрутизатора в этой сети и длиной маски подсети. Номер области, к которой принадлежит данная сеть можно вводить как в формате IP адреса, так и целым десятичным числом. **Например:**

OSPF>configure

OSPF(config)# router

OSPF(config-router)# network 4.7.8.32/24 area 0.0.0.1

OSPF(config-router)# network 192.168.15.1/24 area 0

OSPF(config-router)#

Если ни у одного сетевого интерфейса маршрутизатора нет IP адреса принадлежащего данной подсети, то OSPF не будет заявлять о наличии канала к этой сети, хотя в конфигурации маршрутизатора эта сеть будет прописана (неактивный канал). Как только администратор подключит один из интерфейсов к этой сети командой WANFlex ifconfig <intf> <IP/MASK> alias up, данная сеть автоматически станет участником маршрутизации OSPF.

Таким образом у маршрутизатора появляется внутренний (по отношению к OSPF системе) канал, у которого адресатом является данная сеть. Если эта сеть есть на самом деле адрес физического интерфейса типа *point-to-point*, то у маршрутизатора появляется внутренний канал с адресатом равным идентификатору OSPF маршрутизатора подключенного к противоположному концу этой point-to-point линии связи.

Отменить анонсирование внутреннего канала можно командой:

no network A.B.C.D/M area (A.B.C.D|<0-4294967295>)

В некоторых случаях есть необходимость анонсировать внутренние каналы автоматически для выбранного сетевого интерфейса. Это бывает необходимо, когда IP адреса этого интерфейса (алиасы) образуются и исчезают автоматически. Например, при подключении абонентских блоков к базовой станции через радио интерфейс. Для этого в режиме *config-router* предназначена команда:

auto-interface IFNAME area (A.B.C.D|<0-4294967295>)

Copyright © 2004-2005 by InfiNet Wireless

В команде указывается имя сетевого интерфейса, IP сети (alias) которого буду автоматически анонсироваться OSPF маршрутизатором при их появлении. При этом указывается идентификатор области к которой будут отнесены сети (адресаты) этих каналов. Отменить автоматическое анонсирование каналов для данного интерфейса можно командой в режиме *config-router*:

no auto-interface IFNAME

Для анонсирования внешних каналов предназначена команда в режиме *config-router* :

redistribute (kernel|connected|static) [metric <0-16777214>] [metric-type (1|2)] [route-map WORD]

Определить критерий, по которому маршрутизатор будет анонсировать тот или иной канал из всех существующих можно командой режима config-router:

distribute-list WORD out (kernel|connected|static)

Если не определить этого фильтра, то маршрутизатор будет анонсировать все каналы заданного типа записей системной маршрутной таблицы, если они не будут впоследствии отвергнуты картой маршрутов, указанной в параметрах команды **redistribute**.

Все такие каналы OSPF маршрутизатор заявляет как внешние каналы с типом метрики либо 1, либо 2 (External Type1|2). Информация о наличии внешних каналов распространяется по всей системе OSPF, а не только в пределах области. Исключения составляют лишь тупиковые области OSPF системы, в которые информация о внешних каналах анонсируется как *default gateway* через граничный маршрутизатор этой области (полностью агрегируются). Отличие двух типов метрик заключается в том, что метрика типа 1 – это метрика «соизмеримая» с метриками внутренних для OSPF каналов. При расчете пути до внешнего адресата метрика полного пути до адресата такого канал считается как сумма метрики пути до заявившего этот канал маршрутизатора плюс метрика этого канала. Таким образом, предпочтение будет отдано пути, интегральная метрика которого меньше. Если же внешний канал объявлен с метрикой типа 2, то предпочтение всегда отдается пути, лежащему через маршрутизатор заявившем о таком канале с наименьшей метрикой, несмотря на то, что возможно внутренний путь до этого маршрутизатора и оказался длиннее (дороже). Однако, если два маршрутизатора заявили о внешнем канале с типом метрики 2 и одинаковой метрикой этого канала, то предпочтение опять же будет отдано пути через маршрутизатор, внутренний путь до которого короче. Если же два разных маршрутизатора заявили о канале к одному и тому же внешнему адресату, но с разным типом метрики, то предпочтение отдается пути через маршрутизатор с метрикой внешнего канала типа 1.

WORD – это идентификатор именованного списка доступа, которому должен отвечать адресат записи системной маршрутной таблицы.

Значение и тип метрики для внешних каналов можно задавать и с помощью карты маршрутов (*route-map)*. В этом случае тип и значение метрики можно поставить в зависимость от параметров маршрута (интерфейс, *gateway*, адресат и т.д.), импортированного из ядра ОС.

Если тип и/или значение метрики внешнего канала оказалось не заданным, то маршрутизатор будет считать эти внешние каналы с метрикой по умолчанию и типом 2. Метрика по умолчанию задается при помощи команды:

default-metric <0-16777214>

Если метрика по умолчанию не установлена, то ее значение равно 1.

Copyright © 2004-2005 by InfiNet Wireless

В режиме **redistribute kernel** маршрутизатор не будет заявлять в OSPF систему о наличии у него канала к *default gateway* (адресат = сеть 0.0.0.0/0), даже если он явно прописан в системной маршрутной таблице администратором. Для того, чтобы маршрутизатор заявлял о наличии у него канала к *default gateway* необходимо дать ему явное указание в режиме *config-router*:

default-information originate [always] [metric-type (1|2)] [metric <0-16777214>] [route-map WORD]

атрибуты metric-type (1|2) и metric <0-16777214> определяют аналогичные параметры внешнего канала команды **redistribute.** Указывать их так же необязательно. У этой команды есть еще один необязательный атрибут – **always.** Атрибут **always** указывает на необходимость заявлять о наличии у него канала к *default gateway* даже если такого маршрута нет в системной таблице.

Отмена заявления внешнего канала к *default gateway* производится командой:

no default-information originate

Карта маршрутов (route-map)

Как говорилось ранее, для более гибкого назначения типа и значения метрики внешних каналов, можно воспользоваться картой маршрутов (route-map). Карта маршрутов содержит последовательность записей-условий. Каждая запись описывается своим последовательным номером в карте, условием соответствия импортируемого маршрута данной записи, действиями, производимыми над результирующим объектом в случае соответствия (присваивания типа и/или значения метрики образуемого канала), результирующей реакцией (отвергнуть, принять) и сценарием дальнейшего просмотра записей (может быт не задан). Маршруты просматриваются по карте в порядке увеличения их последовательных номеров. Если маршрут соответствует условию записи, то:

- в случае результирующей реакции отвержения (**deny**), маршрут отбрасывается, просмотр записей карты прекращается, результирующий объект аннулируется (канал не анонсируется)
- в случае результирующей реакции принять (**permit**), с результирующим объектом производятся указанные в записи действия (установка типа и/или значения метрики). Далее, просмотр записей прекращается, либо, если задан сценарий, продолжается с указанного в сценарии номера записи в двух вариантах:
    1. **on-match next** – просмотр продолжается с записи, непосредственно следующей за текущей
    2. **on-match goto <N>** - просмотр записей продолжается с записи с номером больше или равным **N**, но не меньше, чем номер текущей.

Для формирования карты маршрутов применяется команда в режиме *config*:

route-map WORD (deny|permit) <1-65535>

где **WORD** – это идентификационное название (имя) карты маршрутов. Далее указывается порядковый номер и результирующая реакция записи данной карты маршрутов, параметры которой необходимо изменить/добавить. Если записи с таким порядковым номером не существует, она будет автоматически создана. После выполнения этой команды КО переходит в режим редактирования выбранной записи карты маршрутов. Например:

OSPF> configure

Copyright © 2004-2005 by InfiNet Wireless

```
OSPF(config)# route-map testmap permit 10
OSPF(config-route-map)#
```

Далее устанавливается условие соответствия импортируемого маршрута данной записи. Для этого в режиме *config-route-map* применяются команды:

```
match address (<1-199>|<1300-2699>|WORD)
match address prefix-list WORD
match interface WORD
match next-hop (<1-199>|<1300-2699>|WORD)
match next-hop prefix-list WORD
```

Эти команды устанавливают условия соответствия маршрута по трем возможным его параметрам: адресата (destination), gateway (nexthop) и интерфейса. Для одной записи допускается устанавливать несколько различных условий. Если задано несколько условий, то они объединяются по правилу «и». В командах **match next-hop и match address** указывается ссылка (номер или имя) на объект фильтрации: номер или имя **access-list** или имя **prefix-list.** В этом случае условие будет выполнено, если соответствующий параметр маршрута принадлежит указанному фильтрующему списку по соответствующему этому типу списка правилу. В команде **match interface** в качестве параметра указывается имя сетевого интерфейса, которому принадлежит маршрут (канал).

При выполнении маршрутом всех условий записи можно установить значения и/или тип метрики канала, образуемого из этого маршрута при помощи команд режима *config-route-map*:

```
set metric <0-4294967295>
set metric-type (type-1|type-2)
```

Сценарий дальнейшего поведения записи при выполнении маршрутом указанных условий и присвоению образующемуся в результате внешнему каналу параметров, можно создать одной из команд:

```
on-match goto <1-65535>
on-match next
```

Пример конфигурации:

```
OSPF> configure
OSPF(config)# access-list AnyNetwork permit any
OSPF(config)# access-list net200 permit 192.168.200.0/24
OSPF(config)# route-map mapForConnected permit 10
OSPF(config-route-map)# match address net200
OSPF(config-route-map)# set metric 7
OSPF(config-route-map)# route-map mapForConnected deny 11

OSPF(config-route-map)# match address AnyNetwork
OSPF(config-route-map)# router
OSPF(config-router)# redistribute connected route-map mapForConnected
OSPF(config-route-map)#
```

Copyright © 2004-2005 by InfiNet Wireless

В этой конфигурации маршрутизатору указано анонсировать внешние каналы, образованные из connected маршрутов системной таблицы маршрутов с метрикой типа 2. При этом, если адресат этого маршрута – подсеть 192.168.200.0/24, то образованный канал будет иметь метрику 7, любой другой адресат не приведет к анонсированию внешнего канала к нему.

Метрика канала

Метрика канала – это стоимость доставки информации через сетевой интерфейс этого канала. OSPF маршрутизатор автоматически рассчитывает стоимость внутренних каналов исходя из пропускной способности физического сетевого интерфейса, к которому принадлежит канал по формуле:

M = reference_bandwidth/bandwidth.

Где reference_bandwidth – это величина по умолчанию равная 100 Мбит/сек, bandwidth – пропускная способность физического сетевого интерфейса, к которому принадлежит данный канал. Величину reference_bandwidth можно изменять командой в режиме *config-router:*

auto-cost reference-bandwidth <1-4294967>

Параметр указывается в единицах Мбит/сек.

Указанный выше метод определения метрики канала применяется для всех каналов, у интерфейсов которых не задано специфической стоимости. Определить индивидуальную стоимость (метрику) каналов принадлежащих какому-либо интерфейсу можно командами в режиме *config-if*:

cost <1-65535> [A.B.C.D]

Для этого необходимо перейти в режим *config-if* с указанием имени физического сетевого интерфейса для которого будет производиться установка стоимости каналов*:*

interface IFNAME

Например:

OSPF> configure

OSPF(config)# interface eth0

OSPF(config-if)# cost 4 192.168.15.1

OSPF(config-if)#

В команде cost указывается IP адрес, который имеет данный интерфейс в подсети (адресат канала), подключенной к данному интерфейсу. Если этот параметр не указать, то для любого канала относящегося к данному физическому интерфейсу, вне зависимости от подсети-адресата, будет установлена указанная стоимость (метрика).

Деление OSPF системы на области

Протокол OSPF позволяет объединять смежные сети и хосты в группы. Такая группа вместе с маршрутизаторами, имеющими канал в одну (любую) из включенных в группу сетей, называется областью (area). В каждой области работает своя независимая копия алгоритма OSPF. Это означает, что каждая область имеет свою базу данных каналов и соответствующий граф.

Маршрутизатор, при конфигурации которого анонсировались внутренние каналы к сетям принадлежащим только одной области, называется

Copyright © 2004-2005 by InfiNet Wireless

внутренним маршрутизатором области. Маршрутизатор подключенный к сетям более чем одной области, называется *граничным маршрутизатором области* (ABR). Маршрутизатор, объявляющий о наличии у него каналов к внешним адресатам (команда redistribute), называется *граничным маршрутизатором системы* OSPF (ASBR).

Каждой области присваивается уникальный идентификатор *area-id*. Область с *area-id* равным нулю называется магистральной областью OSPF системы (backbone). Магистраль OSPF всегда включает все граничные маршрутизаторы областей. Магистраль отвечает за распространение маршрутной информации между остальными (non-backbone) областями. Магистраль должна обеспечивать смежность (contiguous), однако это не обязательно физическое соседство – магистральные соединения могут организовываться и поддерживаться с помощью настройки виртуальных соединений.

Модели ABR

OSPF маршрутизатор поддерживает четыре модели ABR:

1. **cisco** – маршрутизатор будет считать себя ABR, если у него сконфигурировано несколько каналов к сетям разных областей и хотя бы одна из этих областей - магистральная. При этом канал к магистральной области должен быть активным (реально работающим).
2. **ibm** – аналогична модели cisco, но канал к магистрали может быть и неактивным
3. **standard** – маршрутизатор имеет несколько активных каналов к разным областям
4. **shortcut** – аналогична standard, но этой модели маршрутизатору разрешается использовать топологию немагистральных подключенных к нему областей для построения межобластных маршрутов, пролегающих через такие области (замыкание маршрутов помимо магистрали)

Подробности отличий моделей *cisco* и *ibm* можно узнать в RFC3509. Модель *shortcut* позволяет ABR вносить маршруты между областями, основываясь на топологии подключенных к такому маршрутизатору областей, а не обязательно на информации магистральной, если такие маршруты оказываются «дешевле» магистральных («замыкание» маршрута через немагистральную область). Подробности *shortcut* режима можно узнать в документе ospf-shortcut-abr-02.txt.

Модель ABR переключается командой в режиме *config-router:*

abr-type (cisco|ibm|shortcut|standard)

Использовать ли конкретную (не магистральную) область для «замыкания» межобластных маршрутов можно указать командой режима *config-router*:

area (A.B.C.D|<0-4294967295>) shortcut (default|enable|disable)

Три режима определяют условие использования указанной области для замыкания через нее маршрутов ABR-ом в режиме shortcut следующим образом:

- **Default** – данная область будет использована для замыкания только в том случае, если ABR не имеет или потерял канал к магистральной области.
- **Enable** – область используется для замыкания в любом случае, если «замкнутый» через нее маршрут оказался более дешевым.

Copyright © 2004-2005 by InfiNet Wireless

§ **Disable** – область никогда не используется данным shortcut-ABR для замыкания маршрутов.

Тупиковые области

Некоторую область возможно определить как тупиковую. Это применяется для области, у которой имеется единственный ABR или когда ABR несколько, но для выбора не требуется принимать решение с учетом адреса внешнего (по отношению к области) адресата. Информация о внешних по отношению к OSPF системе каналах не распространяется ABR в тупиковую область. Вместо этого, ABR анонсирует в такую область default gateway, с маршрутом через себя.

Объявить область тупиковой можно в режиме config-router командой:

area (A.B.C.D|<0-4294967295>) stub [no-summary]

Опцию **no-summary** можно указать, если в такую область нет необходимости анонсировать суммарные анонсы каналов других областей, т.к. данный ABR – единственный у данной области. Это сэкономит ресурсы баз данных каналов внутренних маршрутизаторов этой области. Надо помнить, что все маршрутизаторы области должны иметь одинаковую установку данной опции, иначе они не смогут установить отношения смежности, о которых речь пойдет ниже.

Обеспечение связности магистрали. Виртуальные каналы

В общем случае, протокол OSPF требует, чтобы магистраль (area 0) была нераздельной или полностью связной. Т.е любой маршрутизатор магистрали должен иметь в топологии магистрали путь к любому другому маршрутизатору магистрали. Кроме того, каждый ABR должен иметь канал в магистральную область. Однако далеко не всегда есть возможность обеспечить физический канал в магистральную область. В таких случаях между двумя ABR области (один из которых так или иначе имеет канал в магистральную область) не являющейся тупиковой можно организовать виртуальный канал. Это осуществляется в режиме config-router командой:

area (A.B.C.D|<0-4294967295>) virtual-link A.B.C.D

где

- **(A.B.C.D|<0-4294967295>)** – идентификатор области, через которую будет пролегать виртуальный канал, второй параметр
- **A.B.C.D** – идентификатор ABR (router-id), с которым устанавливается этот виртуальный канал. Виртуальный канал должен быть настроен на обоих маршрутизаторах. Например:

Router 192.168.152.45:

OSPF> configure

OSPF(config)# router

OSPF(config-router)# area 0.0.0.1 virtual-link 192.168.78.12

Router 192.168.78.12:

OSPF> configure

OSPF(config)# router

OSPF(config-router)# area 0.0.0.1 virtual-link 192.168.152.45

Формально, организованный виртуальный канал выглядит как сеть типа point-to-point, связывающая два ABR одной области, один из которых имеет физический канал в магистральную область. Эта псевдо сеть считается принадлежащей магистральной области.

Copyright © 2004-2005 by InfiNet Wireless

Фильтрация информации о каналах для области

Суммарная информация о каналах в области, передаваемая ABR в магистраль для других областей (экспорт) может быть отфильтрована. Так же может быть отфильтрована и информация передаваема ABR в данную область из магистрали о суммарных каналах других областей (импорт).

Фильтры устанавливаются в режиме *config-router* командами:

area (A.B.C.D|<0-4294967295>) export-list NAME

area (A.B.C.D|<0-4294967295>) filter-list prefix WORD (in|out)

area (A.B.C.D|<0-4294967295>) import-list NAME

где

- **NAME** – имя фильтрующего списка (*access-list*),
- **WORD (in|out)** – имя фильтрующего префиксного списка (prefix-list) с указанием направления фильтрации (in – импорт, out – экспорт). Фильтры можно устанавливать для любых областей, к которым подключен ABR, кроме магистральной.

Агрегирование каналов. Подавление анонсов

Для каждой области, к которой подключен OSPF маршрутизатор, существует список адресных диапазонов для агрегации каналов перед передачей суммарной информации в магистральную область. Суммарные каналы просматриваются на предмет принадлежности их к одному из адресных диапазонов. Если несколько каналов принадлежат одному из диапазонов, то ABR анонсирует в магистраль (или в другие области магистральные каналы) только один канал с адресатом равным значению этого диапазона и метрикой равной максимальной метрике из всех каналов, попадающих в этот диапазон, либо установленной для этого диапазона при конфигурировании. Допускается так же объявлять, что тот или иной диапазон является блокирующим, и тогда анонсирование каналов попадающих в такой диапазон будет подавляться. При анонсировании суммарных магистральных каналов в другие (немагистральные) области, агрегирование (и подавление) не производится, если область, в которую требуется анонсировать магистральные каналы, является транзитной (через нее проходят виртуальные каналы).

Таким образом, список адресных диапазонов области состоит из записей, каждая из которых содержит:

§ Значение диапазона адресов (R)

§ Признак подавления (not-advertise)

§ Стоимость (метрика) агрегированного канала (C)

§ Анонсируемый канал (Rs)

Если признак подавления не задан, то могут быть сконфигурированы параметры C и Rs. Если адресат одного или более каналов принадлежит R, то маршрутизатор будет анонсировать один канал с адресатом R (или Rs, если задан) и с метрикой равной максимальной метрике каналов (или C, если задана).

Для конфигурирования адресных диапазонов области предназначены несколько команд режима *config-router.*

Команда создает адресный диапазон R и возможно устанавливает у него признак подавления not-advertise:

area (A.B.C.D|<0-4294967295>) range A.B.C.D/M [not-advertise]

Copyright © 2004-2005 by InfiNet Wireless

Команда создает адресный диапазон R и возможно устанавливает для него метрику агрегированного канала C:

area (A.B.C.D|<0-4294967295>) range A.B.C.D/M [cost <0-16777215>]

Команда создает адресный диапазон и возможно устанавливает для него адресат анонсируемого канала Rs вместо R:

area (A.B.C.D|<0-4294967295>) range A.B.C.D/M substitute A.B.C.D/M

Отношения смежности. Соседи

В случае, когда к одной и той же сети имеют каналы более одного маршрутизатора, между ними устанавливаются отношения смежности в целях синхронизации баз данных анонсов каналов области, к которой принадлежит эта сеть. Кроме того, сеть к которой подключены более одного маршрутизатора, является транзитной сетью и, если она не типа точка-точка, является активным объектом OSPF (может анонсировать свои каналы к маршрутизаторам). Функции по анонсированию каналов сети выполняет так называет ведущий маршрутизатор. Ведущий маршрутизатор выбирается из числа подключенных (и работающих) к данной сети OSPF маршрутизаторов на основании их приоритетов, идентификаторов и IP адресов сетевых интерфейсов, посредством которых они к данной сети подключены. Для поддержания отношений смежности и синхронизации маршрутизатор использует специальные протоколы, параметры работы которых должны быть идентичны у смежных маршрутизаторов (соседей). К таким параметрам относятся:

- hello-interval
- dead-interval

По умолчанию, hello-interval равен 10 секундам, а dead-interval равен 40 секундам. Указать другое значение этих параметров для какого-либо сетевого интерфейса можно в режиме *config-if* командами:

dead-interval <1-65535> [A.B.C.D]

hello-interval <1-65535> [A.B.C.D]

Значение параметра задается в секундах. Необязательный параметр определяет IP адрес конкретного канала, если требуется индивидуальная настройка этого параметра. Если этот параметр не указан, то выбранный параметр будет установлен конкретному сетевому интерфейсу в целом, и любой канал, принадлежащий интерфейсу будет иметь такое значение данного параметра, если оно не установлено для этого канала индивидуально. Следует помнить, что для установления отношений смежности необходимо, чтобы эти параметры имели одинаковые значения у маршрутизаторов данной сети.

Как говорилось выше, один из маршрутизаторов, подключенных к одной сети, автоматически выбирается ведущим (DR) руководствуясь тремя параметрами. Если известен приоритет маршрутизатора для канала, то именно значение приоритета имеет решающий вес при выборах ведущего маршрутизатора сети. Если приоритет маршрутизатора не установлен, то на результаты выборов оказывают влияние только значения идентификатора и IP адреса маршрутизатора в рассматриваемой сети.

Установить приоритет маршрутизатора на данном интерфейсе можно в режиме config-if командой:

priority <0-255> [A.B.C.D]

Аналогично предыдущим параметрам, приоритет можно установить как индивидуально каждому каналу на интерфейсе, так и для всего интерфейса

Copyright © 2004-2005 by InfiNet Wireless

в целом. Чем больше значение приоритета, тем больше шансов у данного маршрутизатора стать ведущим для данной сети. Если же установить приоритет равным нулю, то маршрутизатор не будет выбран ведущим ни при каких обстоятельствах.

Протокол OSPF требует, чтобы базы данных каналов всех маршрутизаторов одной области были идентичны. Для этого осуществляется обмен информацией о состоянии каналов между маршрутизаторами данной области. В частности, для этого используются транзитные сети. Для уменьшения сетевого трафика, маршрутизаторы обмениваются изменениями в состоянии каналов не каждый с каждым, а каждый с ведущим и с резервным (Backup). Резервный маршрутизатор сети предназначен для «подстраховки» ведущего, в случае выхода из строя последнего. Выборы резервного маршрутизатора осуществляются по тем же принципам, что и выборы ведущего. При синхронизации баз данных маршрутизаторы обмениваются описаниями баз данных по протоколу master-slave, используя широковещательные IP пакеты. Прием каждого пакета при этом должен быть подтвержден противоположной стороной. В случае неприема подтверждения, передающая сторона производит периодические передачи неподтвержденного пакета (*retransmit*). Администратор OSPF маршрутизатора имеет возможность корректировать периодичность этох попыток для конкретного интерфейса и/или канала этого интерфейса. В режиме *config-if*

retransmit-interval <3-65535> [A.B.C.D]

Аналогично другим параметрам, интервал повторных попыток передачи указывается в секундах.

- Обмен информацией о существующих каналах происходит в случаях:
- запуск маршрутизатора или подключение его к данной сети (образование канала) после проведения выборов ведущего маршрутизатора сети
- при получении информации об изменении состояния каналов какого-либо маршрутизатора области.
- Периодически по мере старения этой информации в базе данных

При получении информации об изменениях состояний каналов, данный маршрутизатор начинает процесс синхронизации своей базы данных со всеми своими соседями, если он является ведущим, или только с ведущим и резервным. Но процесс этот он начинает не сразу предполагая возможность поступления других сведений об изменениях каналов. Это делается для предотвращения сетевых «штормов». Величину этой задержки можно установить и регулировать для каждого интерфейса/канала командой режима config-if:

transmit-delay <1-65535> [A.B.C.D]

Кроме того, маршрутизатор производит периодические обновления информации о каналах со своими соседями При этом передается только устаревшая информация, т.е. возраст которой (время нахождения в базе данных) превысило некоторый порог. По умолчанию порог это равен 1800 секунд (полчаса), но его можно изменить командой режима *config-router:*

refresh timer <10-1800>

Следует заметить, что этот параметр устанавливается для всего OSPF маршрутизатора в целом.

Как было ранее сказано, виртуальный канал представляет собой так же транзитную сеть типа точка-точка. Поэтому, на ней устанавливаются такие же отношения смежности между двумя маршрутизаторами. Соответственно, для виртуальных каналов существуют аналогичные параметры протоколов

Copyright © 2004-2005 by InfiNet Wireless

установления отношений смежности. Эти параметры для виртуальных каналов устанавливаются (регулируются) в режиме *config-router:*

area (A.B.C.D|<0-4294967295>) virtual-link A.B.C.D (hello-interval| <1-65535> retransmit-interval| transmit-delay| dead-interval)

Аутентификация. Проверка подлинности

Для предотвращения несанкционированного подключения маршрутизаторов к OSPF системе в протоколе предусмотрена проверка подлинности пакетов протокола. В настоящий момент маршрутизатор располагает двумя различными способами проверки подлинности (аутентификации):

- **Парольная аутентификация**. Все пакеты, передаваемые в сеть, должны иметь соответствующее значение в 64-битовом поле данных *аутентификации заголовка OSPF*. Значение этого поля представляет собой 64-битовый пароль в незашифрованном виде. Простая парольная аутентификация уязвима для пассивных атак, часто встречающихся в Internet (sniffing), т.к. для передачи пакетов используется широковещание, а пакет содержит пароль в незашифрованном виде.
- **Криптографическая аутентификация**. Для каждого пакета OSPF используется ключ при генерации и проверке цифровых подписей (**message-digest**), добавляемых в конце пакета OSPF. Для построения цифровой подписи используется алгоритм MD5. Цифровая подпись создается с помощью необратимой (one-way) функции на основе пакета OSPF и секретного ключа. Поскольку секретный ключ никогда не передается через сеть в открытом виде, это обеспечивает защиту против пассивных атак

По умолчанию у маршрутизатора не установлено ни какой аутентификация (null- аутентификация).

Аутентификацию можно установить индивидуально для каналов интерфейса (или на весь интерфейс, включая виртуальный канал) и/или индивидульно на какую-либо область, к которой подключен маршрутизатор.

1. Для интерфейсов параметры аутентификации устанавливаются командами режима *config-if*

   Для простой парольной:

   authentication-key AUTH_KEY [A.B.C.D]

   где

   AUTH_KEY – это пароль, а необязательный параметр – **IP адрес** канала в случае индивидуальной установки

2. Для криптографической:

   message-digest-key <1-255> md5 KEY [A.B.C.D]

   где **KEY** – это секретный ключ MD5, а необязательный параметр **– IP адрес** канала в случае индивидуальной установки. <1-255> - это порядковый номер секретного ключа. Т.о. для данного канала или интерфейса можно установить до 255 секретных ключей. Для передачи пакетов маршрутизатор будет использовать последний из установленных ключей. При приеме пакетов маршрутизатор будет использовать ключ с тем же порядковым номером, который использовал передающий.

Copyright © 2004-2005 by InfiNet Wireless

Т.о. маршрутизатор будет принимать через данный канал только пакеты, содержащие совпадающий пароль в случае простой парольной аутентификации или только имеющие цифровую подпись полученную алгоритмом MD5 с использованием одного из сконфигурированных (один из 255) секретных ключа, в случае криптографической аутентификации.

Установив параметры аутентификации, можно включить (резрешить) ее тот или иной режим командой в режиме *config-if*:

authentication	**[(null|message-digest)]**	**[A.B.C.D]**
	Вид аутентификации. null – эквивалентно отсутствию (принудительному отключению) аутентификации. Отсутствие же этого параметра включает простую парольною аутентификацию	IP адрес канала данного интерфейса.

Аналогично конфигурируется аутентификация для виртуальных каналов в режиме *config-router*:

Параметры:

area (A.B.C.D|<0-4294967295>) virtual-link A.B.C.D authentication-key AUTH_KEY

area (A.B.C.D|<0-4294967295>) virtual-link A.B.C.D message-digest-key <1-255> md5 KEY

Включение того или иного типа:

area (A.B.C.D|<0-4294967295>) virtual-link A.B.C.D (authentication|) (message-digest|null)

Тип аутентификации может быть задан и для всей области, к которой принадлежит сеть и канал, по которому принимаются пакеты OSPF. Если аутентификация включена и у интерфейса и у области, то будет применяться тот вид аутентификации, который установлен у интерфейса. Т.о., например, если у интерфейса был установлен вид аутентификации **null**, то аутентификация проводиться не будет вовсе. Для того, чтобы определить вид аутентификации, в случае, если она была полностью отключена у интерфейса(канала), можно установить вид аутентификации для области командой в режиме *config-router:*

area (A.B.C.D|<0-4294967295>) authentication [message-digest]

Если опция **message-digest** не будет указана, то для области включится простая парольная аутентификация.

Как говорилось выше, тип аутентификации области применяется только в том случае, если аутентификация была полностью отключена у интерфейса, через который был принят (отправлен) пакет. Однако, при этом параметры аутентификации используются те, что были установлены у этого интерфейса.

Отключение аутентификации для области производится в режиме *config-router:*

no area (A.B.C.D|<0-4294967295>) authentication

Copyright © 2004-2005 by InfiNet Wireless

Команды просмотра состояния маршрутизатора

Для просмотра текущего состояния маршрутизатора предназначено несколько команд *базового режима* КО. Кроме того, в любом режиме КО существует команда

show running-config

Эта команда выводит на экран текущую конфигурацию маршрутизатора.

Конфигурация показывается в виде набора команд КО маршрутизатора, необходимого для приведения его настроек в текущее состояние.

Например:

OSPF> show running-config

Current configuration:
interface eth0
interface eth1
interface lo0
interface null0
interface tun0
network point-to-point
router
router-id 195.38.45.107
network 1.1.1.1/32 area 0.0.0.0
network 4.7.8.0/24 area 0.0.0.1
network 192.168.15.1/24 area 0.0.0.1
network 195.38.45.107/26 area 0.0.0.0
area 0.0.0.1 virtual-link 192.168.151.10
end
OSPF>

Состояние соседей

show neighbor [A.B.C.D] [detail]

В качестве параметров допускается указывается IP адрес сетевого интерфейса (канала), состояние и набор соседей которого необходимо распечатать. При отсутствии этого параметра, команда выводит на экран суммарную информацию по всем интерфейсам.

Например:

```
OSPF> show neighbor

Neighbor ID     Pri   State           Dead Time   Address         Interface
9.1.1.8           1   Full/DROther    00:00:32    1.1.1.2         tun0:1.1.1.1
192.168.151.1     1   Full/DR         00:00:32    192.168.15.10   eth1:192.168.15.1
192.168.45.116    1   Full/DR         00:00:32    192.168.45.116  eth0:192.168.45.107
192.168.151.10    1   Full/DROther    00:00:39    192.168.151.10  VLINK0
OSPF>
```

Copyright © 2004-2005 by InfiNet Wireless

Колонки таблицы:

- *Neighbor ID* – идентификатор маршрутизатора соседа
- *Pri* – приоритет
- *State* – текущее состояние/статус. Текущее состояние может принимать следующие значения:
  - **Init.** Это состояние говорит о недавнем приеме пакета *Hello* от соседа, с которым еще не организована двухсторонняя связь (т. е., данный маршрутизатор еще не включен в пакет Hello от соседа).
  - **2-Way.** Между соседями организована двухсторонняя связь, обеспечивающая работу протокола *Hello*. С этого состояния начинается организация отношений смежности.
  - **ExStart.** Первый этап организации отношений смежности, обеспечивающий согласование режимов ведущий/ведомый.
  - **Exchange.** В этом состоянии маршрутизатор полностью описывает свою базу данных каналов, посылая пакеты описания своему соседу.
  - **Loading.** В этом состоянии соседу передаются пакеты запросов для получения свежих описаний каналов, которые были обнаружены (но еще не получены) в состоянии Exchange. Т.о. происходит синхронизация баз данных.
  - **Full.** Это состояние говорит об установлении полной смежности с соседом и о завершении текущего процесса синхронизации баз данных анонсов каналов.
  - Текущий статус может принимать следующие значения:
    - **DR** – маршрутизатор был выбран ведущим маршрутизатором сети.
    - **Backup** – маршрутизатор был выбран резервным.
    - **DROther** – маршрутизатор является обычным (не ведущим и не резервным)
- *Dead Time* – оставшееся время ожидания пакета подтверждения присутствия соседа.
- *Address* – IP адрес соседа
- *Interface* – интерфейс (канал) маршрутизатора через который производится обмен служебной информацией с соседом.

Если в команде указана опция **detail,** то информация о соседях выводится в подробном виде, а не в виде сводной таблицы.

База данных

show database

Команда выводит сводную таблицу содержимого базы данных анонсов каналов (LSA).

show database (asbr-summary|external|network|router|summary)	**[A.B.C.D]**	**[adv-router A.B.C.D]**
Тип анонсов канала для просмотра	Адресат канала, анонсы которых требуется распечатать	Идентификатор маршрутизатора, анонсы каналов которого требуется распечатать.

Copyright © 2004-2005 by InfiNet Wireless

Эта команда распечатывает подробную информацию об анонсах каналов.

Например, требуется распечатать базу данных каналов, которые были заявлены транзитными сетями, функцию анонсирования у которых выполняет маршрутизатор с идентификатором 192.168.45.107:

```
OSPF> show database network adv-router 192.168.45.107
       OSPF Router with ID (192.168.151.10)
                Net Link States (Area 0.0.0.0)
                Net Link States (Area 0.0.0.1)
  LS age: 473
  Options: 0x2  : *|-|-|-|-|-|E|*
  LS Flags: 0x6
  LS Type: network-LSA
  Link State ID: 192.168.15.1 (address of Designated Router)
  Advertising Router: 192.168.45.107
  LS Seq Number: 80000001
  Checksum: 0x9148
  Length: 32
  Network Mask: /24
        Attached Router: 192.168.45.107
        Attached Router: 192.168.151.1
                Net Link States (Area 0.0.0.2)
OSPF>
```

Объекты фильтрации

show access-list [(<1-99>|<100-199>|<1300-1999>|<2000-2699>|WORD)]

Команда применяется для просмотра того или иного списка доступа. Если идентификатор списка не задан, выводится содержимое всех списков доступа. Например:

OSPF> show access-list

IP access list any_network

permit any

IP access list net200

permit 192.168.200.0/24

Аналогичная команда применяется для просмотра списков префиксов:

show prefix-list

show prefix-list WORD

Таблица маршрутов

show route

Эта команда распечатывает таблицу маршрутов, полученных в результате вычислений. Например:

```
OSPF> show route
============ OSPF network routing table ============
N IA 1.1.1.1/32            [3] area: 0.0.0.1
                           via 192.168.15.1, eth0
N IA 1.1.1.2/32            [2] area: 0.0.0.1
```

Copyright © 2004-2005 by InfiNet Wireless

```
                            via 192.168.15.1, eth0
N    4.7.8.0/24             [2] area: 0.0.0.1
                            via 192.168.15.1, eth0
N IA 9.1.1.0/24             [12] area: 0.0.0.1
                            via 192.168.15.1, eth0
N IA 192.168.0.0/24         [3] area: 0.0.0.1
                            via 192.168.15.1, eth0
N    192.168.15.0/24        [1] area: 0.0.0.1
                            directly attached to eth0
N IA 192.168.80.0/24        [12] area: 0.0.0.1
                            via 192.168.15.1, eth0
N    192.168.151.0/24       [1] area: 0.0.0.1
                            directly attached to eth0
N IA 192.168.152.0/24       [2] area: 0.0.0.1
                            via 192.168.151.10, eth0
N IA 195.38.45.64/26        [2] area: 0.0.0.1
                            via 192.168.15.1, eth0
============ OSPF router routing table =============
R    192.168.151.10         [1] area: 0.0.0.1, ABR, ASBR
                            via 192.168.151.10, eth0
R    195.38.45.107          [1] area: 0.0.0.1, ABR
                            via 192.168.15.1, eth0

============ OSPF external routing table ===========
N E2 192.168.200.0/24       [1/7] tag: 0
                            via 192.168.151.10, eth0

OSPF>
```

Эта таблица состоит из трех частей:

1. Таблица сетей (**OSPF network routing table**). Этот раздел содержит список полученных маршрутов до всех достижимых сетей (или агрегированных диапазонов областей) OSPF системы. Признак IA в строке означает, что адресат маршрута находится в области, к которой данный маршрутизатор не подключен, т.е. этот путь межобластной или междоменный (inter-area path). В квадратных скобках указывается суммарная метрика всех каналов, через которые пролегает путь к этой сети. Префиксом *via* обозначается маршрутизатор-шлюз, т.е. первый маршрутизатор на пути к адресату (next hop).
2. Таблица путей до граничных маршрутизаторов области (**OSPF router routing table**).
3. Таблица внешних маршрутов, т.е. путей до адресатов внешних каналов OSPF системы (**OSPF external routing table**). Признак E2 указывает на тип метрики внешнего канала (E1 – для метрики типа 1, E2 – для метрики типа 2. Метрика маршрута для внешних путей указывается в формате <метрика пути до маршрутизатора, заявившего этот канал>/<метрика самого канала>.

Состояние интерфейсов

show interface [INTERFACE]

Copyright © 2004-2005 by InfiNet Wireless

Данная команда предназначена для вывода информации о сетевых интерфейсах, включая состояние виртуальных каналов. Информация будет выведена для какого-либо интерфейса отдельно, если в команде задано его имя или для всех интерфейсов в противном случае. Пример такой информации:

```
OSPF> show interface
VLINK0 is up
  Internet Address 192.168.151.10/24, Area 0.0.0.0
  Router ID 192.168.151.10, Network Type VIRTUALLINK, Cost: 2
  Transmit Delay is 1 sec, State Point-To-Point, Priority 1
  No designated router on this network
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:08
  Neighbor Count is 1, Adjacent neighbor count is 1
eth0 is up
  Internet Address 192.168.151.10/24, Area 0.0.0.1
  Router ID 192.168.151.10, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.151.10, Interface Address 192.168.151.10
  Backup Designated Router (ID) 192.168.151.1, Interface Address
         192.168.151.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:05
  Neighbor Count is 1, Adjacent neighbor count is 1
  Internet Address 192.168.152.1/24, Area 0.0.0.2
  Router ID 192.168.151.10, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 192.168.151.10, Interface Address 192.168.152.1
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    Hello due in 00:00:03
  Neighbor Count is 0, Adjacent neighbor count is 0
lo0 is up
  OSPF not enabled on this interface
null0 is down
  OSPF not enabled on this interface
rf4.0 is up
  OSPF not enabled on this interface
rf4.1 is up
  OSPF not enabled on this interface
OSPF>
```

7. Netstat

Таблицы интерфейсов и маршрутов

Синтаксис:

net*stat* ***-r***

net*stat* ***-i***

Описание:

Команда **netstat** отображает состояние различных структур данных, связанных с сетевыми параметрами системы.

"-r" - показывает содержимое системых таблиц маршрутизации.

```
Routing tables

Internet:
Destination        Gateway            Flags     Refs      Use    Mtu  Interface
default            10.1.0.33          UG           2   744837      -  eth0
10.0.0.12/30       10.1.0.61          UG           0        0      -  eth0
10.0.1/30          10.3.1.26          UG           0      607      -  rf0
10.3.1.22          0:40:96:11:6a:43   UHL          1      182      -  rf0
10.3.1.24/30       link#3             UC           0        0      -  rf0
10.3.1.26          0:40:96:10:ef:be   UHL          2      271      -  rf0
127.0.0.1          127.0.0.1          UH           1        0      -  lo0
195.38.43          10.1.0.61          UG           0     1410      -  eth0
195.38.44.192/28   10.3.1.18          UG           0   258947      -  rf0
195.38.46.64/27    10.1.0.61          UG           0        0      -  eth0
195.38.46.128/26   10.1.0.61          UG           0        0      -  eth0
195.38.52          10.3.1.2           UG           0    19367      -  rf0
```

Флаги конкретных маршрутов могут иметь следующие значения:

- **U** - элемент таблицы маршрутизации активен
- **H** - маршрут ведет к хосту. Если этот флаг не установлен, то маршрут указывает на сеть.
- **D** - маршрут был создан посредством протокола перенаправления (icmp redirect)
- **M** - маршрут был модифицирован посредством протокола перенаправления (icmp redirect)
- **G** - маршрут подключен к шлюзу. Если этот флаг не установлен, считается, что пункт назначения подключен непосредственно.
- **S** - статический маршрут, установленный оператором командой **route add**
- **1** - псевдостатический маршрут, установленный в результате действия команды **rip static**
- **L** - маршрут указывает на непосредственно подключенный хост (для данного маршрута возможно выполнение ARP запроса)
- **C** - при использовании данного маршрута возможно создание более специфических маршрутов (например с флагом **L**).

-i - отображает информацию о каждом сетевом интерфейсе включённом в систему.

```
Name  Mtu   Network          Address              Ipkts Ierrs    Opkts Oerrs
lo0   32768 Link:                                     0     0        0     0
lo0   32768 127              127.0.0.1                0     0        0     0
eth0  1500  Link:           0.5a.35.88.c3.54    192477     0   180180     0
eth0  1500  10.1.0.32/27     10.1.0.60           192477     0   180180     0
rf0   1500  Link:           0.5a.35.88.c3.55    324247  9085   348138     0
rf0   1500  10.3.1.8/30      10.3.1.9            324249  9085   348140     0
rf0   1500  10.3.1/30        10.3.1.1            324249  9085   348140     0
rf0   1500  10.3.1.24/30     10.3.1.25           324249  9085   348140     0
```

8. Ipfw (IP Firewall)

Пакетный фильтр.

Copyright © 2004-2005 by InfiNet Wireless

Общее описание

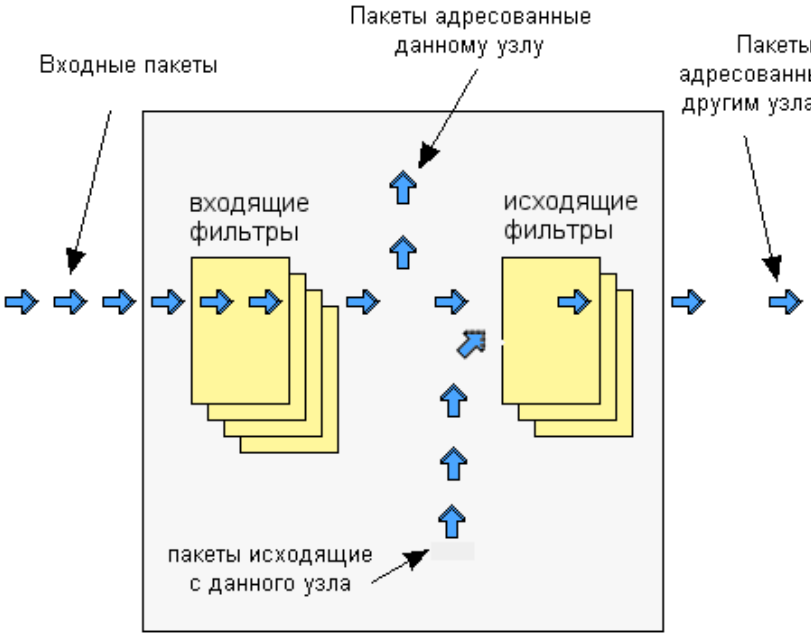
ipfirewall - это механизм позволяющий фильтровать проходящие через узел IP пакеты по различным критериям. Системный администратор может определить набор входящих (**addincoming**) и исходящих (**addoutgoing**) фильтров. Входящие фильтры описывают пакеты, которые могут приниматься **данным** узлом. Исходящие фильтры описывают пакеты, которые могут выходить **из** данного узла после маршрутизации.

Каждый фильтр описывает класс пакетов и определяет способ их обработки (отбросить и зарегистрировать, пропустить, пропустить и зарегистрировать).

Пакеты могут фильтроваться на основании следующих характеристик:

- тип пакета (все IP, TCP, UDP, ICMP, ARP)
- адрес источника и/или получателя (и номера портов для TCP и UDP)
- интерфейс, через который поступил пакет
- пакет установления TCP-соединения.
- пакет является первым, не первым или последним фрагментом фрагментированного пакета
- включены или нет различные опции IP протокола.
- MAC-адрес станции назначения или источника.

Рисунок ниже иллюстрирует прохождение пакетов через систему фильтрации маршрутизатора:



Фильтры могут быть двух классов (наборов) - запрещающие (**reject**) и разрешающие (**accept**).

Кроме того, фильтр может быть применим ко всем принимаемым пакетам или только к пакетам, поступающим через определённый интерфейс.

Каждый принимаемый пакет сравнивается со всеми правилами в наборе по очереди. Первый же фильтр, который полностью соответствует принятому пакету, определяет его дальнейшую судьбу. Если фильтр является разрешающим (**accept**), то пакет принимается, если фильтр запрещающий, то пакет отбрасывается. Если нет ни одного фильтра в наборе, или ни один из них не подходит к данному пакету, то пакет принимается.

Copyright © 2004-2005 by InfiNet Wireless

Внимание! Отбрасываемый пакет просто уничтожается без уведомления источника.

Фильтры определяются с помощью команды **ipfw**. Например, команда

ipfw add reject all from 192.168.5.3 to 192.168.11.7

добавит в набор входящих фильтров запрещающий фильтр, который отбросит все пакеты, приходящие с адресом источника 192.168.5.3 и адресом назначения 192.168.11.7.

Для более полного понимания работы фильтра необходимо ознакомиться с описанием того, **как строятся фильтры** и механизмом **применения фильтров**.

Интерфейс

ipfw - командный интерфейс к пакетному фильтру **ipfirewall**

Синтаксис:

*ipfw **list***

*ipfw **flush***

*ipfw **add**[incoming] [num] . . .*

*ipfw **addout**[going] [num] . . .*

*ipfw **del** num*

*ipfw **mov** num1 num2*

*ipfw **quiet***

*ipfw **show***

*ipfw **reset***

Описание

ipfw list

Текущий набор фильтров выводится на консоль.

ipfw flush

Все определённые в данный момент фильтры в обоих (маршрутизирующем и блокирующем) наборах удаляются. Механизм фильтрации отключается.

***ipfw add**[**incoming**] [num] . . .*

***ipfw addout**[**going**] [num] . . .*

Эти две команды используются для добавления фильтра в наборы `addincoming` и `addoutgoing` соответственно. За ключевым словом `add*` следует **описание фильтра**.

Необязательный параметр **num** используется для явного указания порядкового номера правила в списке. Все ранее написанные правила, начиная с этого номера и выше, будут перенумерованы автоматически.

***ipfw del** num*

Используются для удаления фильтра из списка.

Удаляемый фильтр указывается своим порядковым номером **num**, который можно посмотреть командой **ipfw list**.

Copyright © 2004-2005 by InfiNet Wireless

Оставшиеся в списке правила будут автоматически перенумерованы.

ipfw mov *num1 num2*

Эти две команды используются для перемещения фильтра с номером **num1** в позицию **num2**

ipfw [-]quiet

Отключение регистрации отброшенных пакетов. По умолчанию регистрация включена, для отключения её следует выполнить команду **ipfw quiet**

ipfw ***show***

Данная команда показывает счётчики срабатывания для каждого фильтра

ipfw ***reset***

Команда сбрасывает счетчики срабатывания фильтров.

Фильтры

Синтаксис:

[interface] disp

[vlan=N] [dot1p=N] [ether=X] [dscp=N|tos=N]

proto [modifiers] ***from*** *[****not****] endpoint* ***to*** *[****not****] endpoint*

Описание:

Здесь приводится полное описание синтаксических правил для составления пакетных фильтров. В основном рассматривается именно синтаксис (т.е. строка описания фильтра), но и вопросам семантики (т.е. назначению фильтра) тоже уделяется некоторое место.

Общая форма описания фильтра приведена выше в подразделе **Синтаксис**. Необязательное поле **interface** определяет имя сетевого интерфейса, к которому применим данный фильтр. Имя интерфейса зависит от конкретной модели маршрутизатора и может, например, иметь вид **eth0** или **rf4.0** для обозначения ethernet или радио-интерфейса соответственно. Если поле **interface** задано, то фильтр будет применяться только к тем пакетам, которые поступают или выходят через этот интерфейс.

Поле **disp** (сокращение от disposition) определяет действие, которое должно быть выполнено в случае срабатывания данного фильтра. Возможные значения: **accept** или **reject**. Если указано **accept**, то пакет свободно пройдёт сквозь фильтр. Использование **reject** приводит к тому, что пакет будет отброшен фильтром. Вслед за действием можно указать необязательный параметр **log** (**accept log** или **reject log**), в этом случае при срабатывании фильтра в системный журнал будет занесена соответствующая информация.

Модуль **"ipfw"** дополнен фильтром **"rpfilter"** (reverse path filter). Этот фильтр позволяет убедиться, что отправитель пакета действительно доступен через тот интерфейс, через который пакет от него поступил в систему. Если фильтр не срабатывает, то обработка пакета модулем ipfw продолжается, иначе пакет уничтожается. Данный фильтр можно вставлять в список правил первым:

ipfw add rpfilter all from 0/0 to 0/0

Copyright © 2004-2005 by InfiNet Wireless

Поля **disp** может принимать значение "**pass**", позволяющее "пропустить" правило, выполнив связанные с ним действия, и продолжить просмотр других правил по списку.

Например:

ipfw add pass log tcp from 0/0 to 0/0

При срабатывании данного правила, работа фильтра продолжится дальше, а информация о пришедшем пакете запишется в журнал.

Параметры **[vlan=N] [dot1p=N] [ether=X] [dscp=N|tos=N]** – представляют собой классификаторы, позволяющие анализировать VLAN ID, приоритет 802.1p, тип пакета (EtherType), а также поле ip_tos на наличие конкретного значения метки DSCP или IP precedence.

Поле **proto** определяет конкретный протокол семейства IP, используемый фильтром. Возможные значения: **tcp**, **udp**, **icmp**, **arp**, **all** либо числовое значение номера протокола.

Необязательное поле **modifiers** может быть использовано для указания некоторых дополнительных характеристик пакета. Они описаны позже.

Обязательное ключевое слово **from** разделяет поля **proto** и **modifiers** от адреса источника (**endpoint**). Ключевое слово **to** отделяет адрес источника от адреса получателя (**endpoint**).

endpoint определяет адреса источника или получателя. Точный синтаксис полей **endpoint** зависит от значения поля **proto**. Если **proto** указан как **all** или **icmp**, то **endpoint** содержит адресную информацию. Если **proto** задано как **udp** или **tcp** , то **endpoint** содержит адресную информацию и необязательный список портов.

Адресная информация задаётся как IP-адрес и необязательная маска (mask). IP-адрес должен задаваться в традиционном числовом формате (nn.nn.nn.nn). Необязательная **маска** может быть задана как длиной маски в битах, так и числовым значением в формате (nnn.nnn.nnn.nnn). Возможные форматы задания адресной информации:

nn.nn.nn.nn

nn.nn.nn.nn:xxx.xxx.xxx.xxx

nn.nn.nn.nn/NN

Двоеточие указывает на то, что маска задана в числовом адресном формате.
Символ "/" указывает на использование длины маски (число начальных бит, установленных в единицу, остальные будут установлены в ноль).

Например: 192.168.9.0/24 описывает адрес сети 192.168.9.0 с длиной маской в 24 бита.

Второй вариант: 192.168.9.0:255.255.255.0.

Запись 0/0 описывает все возможные IP-адреса.

Если необходимо составить фильтр, который применяется к нескольких сетевым адресам или группам, то вместо повторения однотипных правил удобнее и эффективнее сгруппировать все фигурирующие адреса в соответствующий **список доступа** и указать имя списка в качестве IP-адреса (**$ACLRULE**)

Существует несколько предопределённых динамических ACL списков, которые невозможно построить другим способом.

Copyright © 2004-2005 by InfiNet Wireless

Список **$LOCAL** включает в себя все локальные адреса, принадлежащие данному маршрутизатору. Его можно использовать для удобной записи фильтров ограничивающих (или разрешающих) доступ к самому устройству.

ipfw add accept all from 0/0 to $LOCAL

Список **$ROUTE** содержит текущую системную таблицу маршрутов, за исключением **default route**. Совпадение адреса с этим списком означает, что для данного адреса существует точный маршрут и не будет использован маршрут по умолчанию.

ipfw add reject all from 0/0 to not $ROUTE

Для интерфейсов, имеющих физические MAC-адреса в стандарте Ethernet, допускается использовать непосредственно **цифровое значение MAC адреса** с предшествующим ключевым словом "**mac**". При этом для входящих фильтров можно указать только MAC-адрес источника, а для исходящих только адрес получателя. Кроме того, вместо числового значения, можно указывать ключевое слово "**$BS**". В этом случае в качестве числового значения будет использоваться реальный MAC-адрес базовой станции к которой подключен абонентский блок (если таковая имеется).

Следует иметь ввиду, что правила использующие MAC-адреса для входящих пакетов будут обработаны раньше всех остальных правил, а правила для исходящих пакетов, напротив, будут обработаны в самую последнюю очередь. Будьте внимательны.

После ключевых слов "**from**" и "**to**" возможно использование отрицающей приставки "**not**", её действие будет распространяться только на соответствующий адрес(а), но не на порты, если они используются в команде.

Например:

ipfw add reject all from mac 0012345678 to 0/0

ipfw addout reject all from 0/0 to mac 0012345678

ipfw add rf1 reject all from mac $BS to 0/0

ipfw add reject all from 0/0 to not 1.1.1.0/24

Список портов задаётся в виде простого перечисления номеров портов разделённых пробелами.

Первый элемент в списке может быть **парой** портов, разделённых двоеточием. Эти порты будут определять диапазон номеров (от меньшего к большему, включая указанные).

В текущей реализации может быть указано до 10 номеров портов в списке.

Заметьте, что пакеты, которые не являются **первым фрагментом** фрагментированного IP-пакета, не проверяются на номер порта (поскольку номер порта фигурирует только в первом фрагменте). Это не является серьёзным ограничением, поскольку если первый фрагмент будет отвергнут фильтром, то остальные фрагменты будут отвергнуты IP-протоколом на целевой машине.

Модуль **"ipfw"** дополнен фильтром **"rpfilter"** (reverse path filter). Этот фильтр позволяет убедиться, что отправитель пакета действительно доступен через тот интерфейс, через который пакет от него поступил в систему. Если фильтр не срабатывает, то обработка пакета модулем ipfw продолжается, иначе пакет уничтожается. Данный фильтр можно вставлять в список правил первым:

ipfw add rpfilter all from 0/0 to 0/0

Copyright © 2004-2005 by InfiNet Wireless

Поле **modifiers** используется для указания дополнительных характеристик пакета, которые должен учитывать фильтр.

Возможные значения:

- **tcp_connection**

  определяет, что фильтр относится только к пакетам установления TCP соединения.

- **connection**

  является синонимом **tcp_connection**.

  Технически, пакет запроса соединения имеет TCP заголовок с установленным флагом SYN и сброшенным флагом ACK.

- **ip_fragment**

  указывает, что фильтр относится только к фрагментированным пакетам.
  Технически, либо поле **offset** в заголовке пакета имеет ненулевое значение, либо установлен бит **more fragments**.

- **ip_head_fragment**

  указывает, что фильтр применим только к первому фрагменту фрагментированного пакета.

  Технически, поле **offset** равно нулю и бит **more fragments** установлен.

- **ip_tail_fragment**

  указывает, что фильтр применим ко всем фрагментам фрагментированного пакета, **кроме первого**.

  Технически, поле **offset** ненулевое (значение бита **more fragments** не имеет значения).

- **ip_option**

  фильтр применим к IP-пакетам, которые имеют любые установленные IP опции (кроме опции **NO-OP**).

- **ip_recroute_option**

  фильтр применим только к IP-пакетам, которые имеют включенные опции **record-route** или **timestamp IP** и не имеют никаких других опций.
  Эти опции могут использоваться злоумышленниками для построения карты вашей сети. Они не несут никакой другой угрозы.

- **ip_misc_option**

  фильтр применим только к пакетам, имеющим одну или более IP-опций кроме **record-route**, **timestampIP** или **NO-OP**. Многие из IP опций группы misc могут быть использованы и используются злоумышленниками для обхода фильтров и проникновения в чужую сеть.

Есть несколько дополнительных правил для поля **modifiers**, которые нужно отметить:

- **tcp_connection** может быть использовано только если поле **proto** имеет значение **tcp**.

Copyright © 2004-2005 by InfiNet Wireless

- Если использована более чем одна из опций **ip_fragment**, **ip_head_fragment** или **ip_tail_fragment**, то последние будут отменять действие первых.
- Если использована более чем одна из опций **ip_option**, **ip_recroute_option** или **ip_misc_option**, то последние будут отменять действие первых.

Пакет должен удовлетворять всем установленным опциям, в противном случае он будет пропущен фильтром.

Правила обработки пакетов пакетным фильтром

Здесь приводится детальное описание того, как пакеты обрабатываются пакетным фильтром. Каждый пакет, который принимается маршрутизатором, проходит через **блокирующий набор** фильтров. Пакеты, которые принимаются блокирующими фильтрами, поступают на обработку в IP-компонент ядра маршрутизатора. Если IP стек решает, что пакет не относится к данной машине и должен быть отправлен дальше, то он проходит через **маршрутизирующий набор** фильтров. Информация о пакетах, которые отбрасываются фильтрами, выводится на системную консоль, а сами пакеты уничтожаются без уведомления источника.

Пакет, "проходящий сквозь" набор фильтров, сравнивается с каждым фильтром в наборе, начиная с первого и заканчивая либо концом набора, либо конкретным подходящим фильтром.

Логика поведения фильтра следующая:

- Если набор фильтров пуст, то пакет принимается.
- Иначе, первый же подходящий фильтр определит судьбу пакета. Если фильтр разрешающий, то пакет принимается. Если фильтр запрещающий, то пакет отбрасывается.

Если не нашлось ни одного фильтра, подходящего к пакету, то пакет принимается.

Логика работы конкретного фильтра представлена ниже:

- Если поле **proto** фильтра не **all** и это не тот же протокол, что и в пакете, то фильтр пропускается.
- Если адрес источника в пакете и в фильтре не совпадают, то фильтр пропускается (если адрес был указан с **маской**, то, перед сравнением, маска накладывается на оба адреса).
- Если адреса получателя в пакете и в фильтре не совпадают, то фильтр пропускается. (опять-таки, **маска** обрабатывается так, как было описано ранее).
- Если задан параметр **ip_fragment**, но пакет не является фрагментом, то фильтр пропускается.
- Если задан параметр **ip_tail_fragment**, но пакет является либо первым, либо единственным фрагментом, то фильтр пропускается.
- Если задан параметр **ip_head_fragment**, но пакет не является первым фрагментом фрагментированного пакета, то фильтр пропускается.
- Если задан параметр **tcp_connection**, но пакет не является первым или единственным фрагментом TCP/IP пакета, который пытается установить соединение, то фильтр пропускается.

Copyright © 2004-2005 by InfiNet Wireless

- Если задан параметр **ip_option**, но пакет не имеет никаких IP опций (кроме возможно **NO-OP** или **EOL**), то фильтр пропускается.
- Если задан параметр **ip_recroute_option**, но пакет не имеет указанных опций, то фильтр пропускается.
- Если задан параметр **ip_misc_option**, но пакет не имеет никаких IP опций кроме, возможно, **record-route**, **timestamp**, **EOL** или **NO-OP**, то фильтр пропускается.
- Если фильтр имеет параметр **proto** установленный в **udp** или **tcp** и если **адрес источника** в фильтре имеет **список портов**, то если пакет не является первым (или единственным) фрагментом или порт источника в пакете не совпадает со списком портов в фильтре, то фильтр пропускается.
- Если фильтр имеет параметр **proto** установленный в **udp** или **tcp** и если **адрес получателя** в фильтре имеет **список портов**, то если пакет не является первым (или единственным) фрагментом или порт получателя в пакете не совпадает со списком портов в фильтре, то фильтр пропускается.
- Пакет обрабатывается фильтром в соответствии с указанным полем **disp**.

Особенности работы фильтра с ARP пакетами:

- ARP пакеты будут всегда пропускаться для тех IP-адресов и диапазонов IP-адресов, которые указаны в разрешающих (**accept**) фильтрах, даже если эти фильтры созданы для других типов пакетов.

Примеры

Здесь описывается, как использовать команду **ipfw** для создания фильтров различного назначения.

Простые примеры

В качестве первого примера можно рассмотреть фильтр, который запрещает прохождение всех пакетов с некоторого неблагонадёжного адреса 1.1.1.1 на адрес 2.2.2.2:

```
ipfw add reject all from 1.1.1.1 to 2.2.2.2
```

Поскольку враги имеют тенденцию группироваться, можно закрыть прохождение пакетов со всей вражеской сети:

```
ipfw add reject all from 1.1.1.0/24 to 2.2.2.2
```

Число 24 после символа "/" (слэш) означает длину маски в битах. Длина маски 24 как раз соответствует сети класса C, состоящей из 256 узлов.

Используя символ ":" можно переписать предыдущую команду следующим образом:

```
ipfw add reject all from 1.1.1.1:255.255.255.0 to 2.2.2.2
```

Следующая команда завершает этот абзац:

```
ipfw add reject all from 1.1.1.0/24 to 0/0
```

Она полностью блокирует все пакеты из вражеской сети 1.1.1.0 класса C посланные на любой адрес (если, конечно, они будут проходить через данный маршрутизатор).

Фильтрация по номерам портов

Copyright © 2004-2005 by InfiNet Wireless

Теперь предположим, что мы хотим написать фильтр, который позволит всем обращаться к сервису **smtp** (почтовый агент) на машине с IP-адресом 192.5.42.1. Это можно сделать следующей командой:

ipfw add accept tcp from 0/0 to 192.5.42.1 25

Ключевое слово **tcp** определяет, что фильтр будет применяться только к TCP пакетам. Сразу за адресом хоста следует номер порта 25, который и определяет сервис **smtp**.

Можно использовать **список портов** для указания сразу нескольких портов в одной команде. Первый элемент списка портов может задавать диапазон номеров от меньшего к большему, разделённый двоеточием. Например, команда

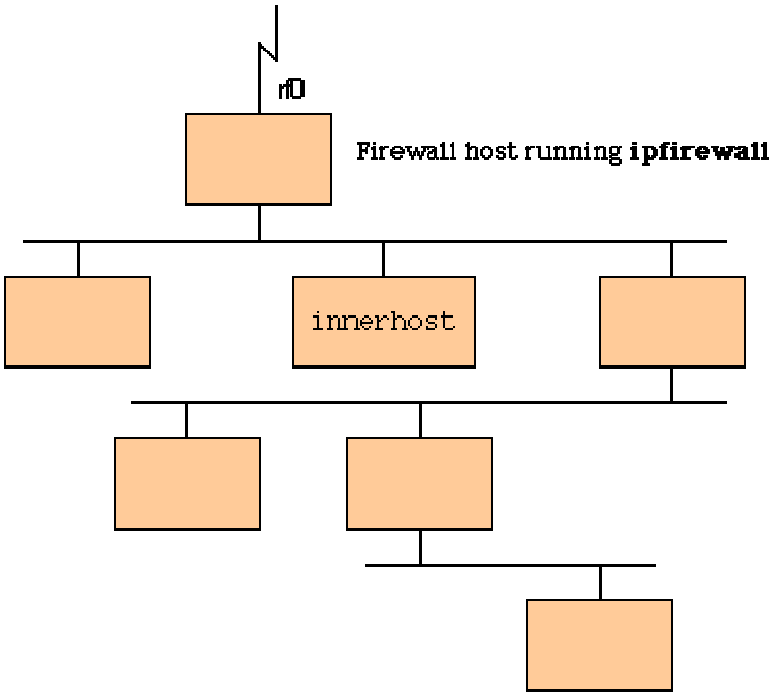
ipfw add accept tcp from 0/0 to 1.1.1.1 900:5000 25 113

разрешит прохождение TCP пакетов на адрес 1.1.1.1, если целевой порт при этом попадает в диапазон от 900 до 5000 или равен 25 (**smtp**) или 113 (**ident**).

IP spoofing

В предыдущих примерах адрес источника использовался как главный и единственный критерий проверки благонадёжности источника. К сожалению, имеется возможность посылать пакеты с неблагонадёжного адреса подставляя в качестве обратного адреса тот, которому вы доверяете (этот вид атаки называется **IP spoofing**). Понятно, что проверять один адрес источника недостаточно. Необходимо также проверить каким путём шёл пакет или, что более практично, через какой интерфейс он был принят.

Пример сети показан ниже:



Все подсети внутренней сети, включая и адрес хоста **innerhost**, принадлежат одной сети (или группе сетей). Предположим, во внешней сети нет и не может быть ни одного хоста, принадлежащего диапазону, выделенному для внутренней сети. Следовательно, любые пакеты, которые **принимаются через интерфейс rf4.0 маршрутизатора, на котором запущен firewall**, и имеют адрес источника, принадлежащий диапазону адресов внутренней сети, должны блокироваться. Следующая команда позволяет сделать это:

Copyright © 2004-2005 by InfiNet Wireless

ipfw add rf4.0 reject all from innerhost/16 to 0/0

В отличие от предыдущих примеров, этот фильтр будет применяться только к пакетам приходящим через интерфейс rf4.0. Пакеты, поступающие через любой другой интерфейс, блокироваться не будут (в примере предполагалось, что внутренняя сеть имеет адреса сети класса B). В качестве дополнительной меры, возможно, имеет смысл заблокировать также прохождение пакетов с адресом источника из сети **loopback** (127.0.0.0):

ipfw add rf0 reject all from 127.0.0.0/8 to 0/0

IP spoofing как метод атаки широко использовался в сети Интернет (для дальнейшей информации смотри **CERT summary CS-95:01**, а также **summaries** на сервере **CERT WWW site**).

Важно понимать, что злоумышленник может использовать метод **IP spoofing** для взлома вашей системы даже несмотря на то, что обратные пакеты никогда не вернутся к отправителю. Смотри, например **CERT advisory CA-95:01** .

Фильтрация TCP соединений

TCP/IP клиенты обычно используют порты в диапазоне от 900 до 5000, а сервера обычно обслуживают порты с номерами ниже 900 или выше 5000. Следовательно, следующая пара фильтров позволит запретить любым внешним клиентам работу с нашими серверами (предполагается, что rf4.0 это интерфейс соединяющий нас с внешним миром):

ipfw add rf4.0 accept tcp from 0/0 to 0/0 900:5000

ipfw add rf4.0 reject tcp from 0/0 to 0/0

Первый фильтр пропускает все входящие извне пакеты, посылаемые на сервисы с портами в диапазоне от 900 до 5000 (обычно используемые нашими клиентами), а второй фильтр отбрасывает всё остальное.

К сожалению, этого недостаточно. Некоторые внутренние сервера могут использовать порты в диапазоне от 900 до 5000. Предыдущая пара фильтров позволит внешним станциям работать с нашими серверами. Проблема в том, как закрыть внутренние сервера использующие диапазон портов от 900 до 5000 и в то же время позволить работать нашим клиентам в этом же диапазоне. Одно из решений - блокировать попытки установления соединения между внешним клиентом и внутренним сервером.
Модификатор **tcp_connection** даёт такую возможность:

ipfw add rf0 reject tcp connection from 0/0 to 0/0 900:5000

ipfw add rf0 accept tcp from 0/0 to 0/0 900:5000

ipfw add rf0 reject tcp from 0/0 to 0/0

Первый фильтр предотвращает попытки установить соединение извне с нашими внутренними серверами в диапазоне портов от 900 до 5000. Второй фильтр разрешает другие входящие TCP пакеты в этом диапазоне, и последний фильтр отбрасывает все прочие TCP пакеты.

Ненадёжный UDP

В отличие от протокола TCP, который ориентирован на установление соединений, UDP протокол использует отдельные пакеты (датаграммы). В этом протоколе, каждый пакет передаётся независимо от других и логические сеансы, которые могли бы устанавливаться между UDP/IP клиентом и сервером существует только на уровне приложений и не видны на уровне UDP.

Copyright © 2004-2005 by InfiNet Wireless

Поскольку все пакеты независимы, по заголовку пакета невозможно определить, посылается пакет от сервера клиенту или наоборот (фактически, в протоколе UDP, участники действуют как равноправные партнёры и термины **сервер** и **клиент** явно не определены. Поэтому лучшее, что мы можем сделать, это возможно более точно определить диапазон портов UDP, который может использоваться для связи с внешним миром.

Сервер имён (DNS) является примером сервера, использующего протокол UDP.
Предполагая, что rf4.0 - это наше соединение с внешним миром, следующая пара фильтров обеспечит взаимодействие между нашим и внешними DNS серверами, заблокировав весь остальной UDP трафик:

ipfw add accept udp from 0/0 53 to 0/0 53

ipfw add rf0 reject udp from 0/0 to 0/0

Хотя это может показаться простым, на самом деле очень трудно реализовать более открытую политику в области UDP обмена, не создав при этом больших дыр в безопасности. Если вы решите разрешить локальным клиентам общаться с внешними UDP серверами, то примите во внимание следующие соображения (это ДАЛЕКО не полный перечень):

Если вы имеете NFS сервера, имейте в виду, что традиционно они используют UDP порт 2049 (TCP/IP реализации NFS используют TCP порт 2049, который, вероятно, может быть защищён с помощью модификатора **tcp_connection** - смотри выше).

Некоторые реализации RPC portmapper имеют серьёзные проблемы с безопасностью. Будьте очень осторожны, разрешая внешний доступ к внутреннему ресурсу **portmapper** (TCP и UDP порт 111).

Будьте очень осторожны в выборе того, какие порты источника и назначения вы разрешаете. Вы можете поддаться искушению разрешить приём входящих пакетов, которые приходят с определённых UDP портов. Если вы сделаете это, не забудьте, что злоумышленник может легко посылать любые TCP/IP и UDP/IP пакеты с **любыми** номерами портов и комбинациями адресов.

Некоторые сервисы сети Microsoft LAN manager используют UDP. Зная патологическую неприязнь фирмы Microsoft к открытым и безопасным протоколам и учитывая беспрецедентное число ошибок в реализации, лучше исключить всякую возможность потенциальным злоумышленникам воспользоваться этой дырой в безопасности:

ipfirewall add rf0 reject tcp from 0/0 to 0/0 135:139

ipfirewall add rf0 reject udp from 0/0 to 0/0 135:139

Данный фрагмент достаточно надёжно закрывает внутреннюю сеть от большинства возможных посягательств несанкционированного доступа к вашей сети (если в ней имеются серверы или станции на базе Windows NT/95/98).

IP фрагменты

Модификаторы **ip_fragment**, **ip_head_fragment** и **ip_tail_fragment** предназначены для управления потоком фрагментированных IP пакетов. Чтобы понять, как использовать их в своих фильтрах, вы должны обратить внимание на следующие моменты:

- Фильтр, который проверяет номера портов TCP или UDP, никогда не проверяет IP фрагменты, если только это не первый фрагмент в последовательности.

Copyright © 2004-2005 by InfiNet Wireless

- если ваш фильтр позволяет приём IP фрагментов, то злоумышленник может применить атаку типа "denial-of-service" (отказ в обслуживании), посылая огромное число фрагментов с разными адресами источника, что может привести к нехватке памяти на вашем маршрутизаторе.

Если вы хотите предотвратить потенциальную атаку типа denial-of-service, то, вероятно, единственное решение заключается в запрете на приём всех фрагментированных пакетов:

ipfw add reject all ip_fragment from 0/0 to 0/0

Это, безусловно, повышает общий уровень защищённости, хотя следует иметь в виду, что злоумышленник может использовать другие пути для атаки (например, просто забрасывая вас пакетами или просто посылая кучу бесполезной почты)

Кроме того, блокирование всех фрагментированных пакетов может помешать нормальной работе.

К примеру, в каждом сегменте сети есть понятие максимально возможного размера пакета (обычно около 1500 байт). Но даже если отправитель посылает не фрагментированный пакет, нет никакой гарантии, что он дойдёт таковым до получателя. Вполне вероятно, что где-то по пути встретится участок с меньшей максимально допустимой длиной пакета, и он будет разбит на фрагменты. Даже новейшие протоколы определения минимальной длины пакета на всём пути от источника до получателя не гарантируют успеха, поскольку каждый IP-пакет доставляется независимо от других и может идти совершенно различными путями. То есть, блокирование фрагментированных пакетов может приводить к нарушениям в работе сетевых приложений (временным, либо постоянным) с определёнными хостами.

Если вы решите разрешить прохождение фрагментированных пакетов, то, вероятно, одним из первых фильтров можно смело поставить:

ipfw add accept all ip_tail_fragment from 0/0 to 0/0

Разрешение прохождения фрагментов пакета кроме первого не создаёт проблем с безопасностью (за исключением случаев атаки "отказ в обслуживании") поскольку первый фрагмент пакета, который несёт основную информацию о его назначении, будет контролироваться другими фильтрами. Если первый фрагмент пакета будет отвергнут каким-либо фильтром, то оставшиеся фрагменты будут доставлены получателю и уничтожены его программным обеспечением по истечении некоторого интервала времени (обычно 60 секунд).

9. Loadm (монитор загрузки каналов)

Монитор загрузки каналов

Синтаксис:

loadm [-B] [-l] [-w XX] интерфейс

Описание:

Команда **loadm** позволяет быстро оценить загрузку любого из интерфейсов системы.
По умолчанию вывод выполняется в одну строку с интервалом в одну секунду и все значения представлены в тысячах **бит** в секунду (Кбод) Дополнительные ключи позволяют изменить используемые умолчания:

- **-B** - выводить значения в тысячах **байт** в секунду (Кбайт)
- **-l** - выводить информацию построчно (строка за строкой)

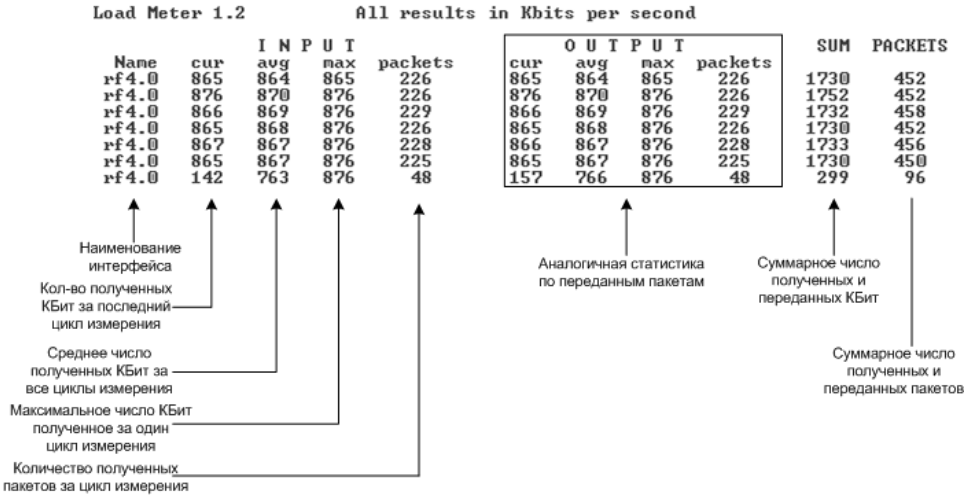
Copyright © 2004-2005 by InfiNet Wireless

- **-w XX** - задаёт интервал отображения изменений.

Внимание: для радиоинтерфейса rf4.0, loadm показывает общую загрузку канала, а не только ту, которая направлена или исходит из данного устройства. Например, даже если абонентский блок сам не учавствует в обмене, он может слышать пакеты других абонентских блоков и/или базовой станции. То есть под каналом в данном случае понимается весь частотный диапазон, в котором работают устройства данного типа и могут влиять друг на друга. Эта особенность присуща тоько устройствам с радиомодулем R200, установленным в модели RWR-110/120/210/510/520 Поэтому реальную загрузку устройства лучше смотреть на интерфейсе eth0.

Пример:

loadm -l rf4.0



Прервать вывод на экран можно нажатием славиш Ctrl/C или ESC

10.Bpf (Berkeley Packet Filter)

Модуль отбора пакетов

Синтаксис:

bpf *ifname ADDR PORT*

bpf *ifname* ***-***

Описание:

Модуль отбора пакетов, реализуемый командой **bpf**, позволяет перехватить весь поток информации, проходящий через любой интерфейс системы, и направить его на удалённую рабочую станцию для дальнейшего анализа и контроля. Отбор пакетов производится путём дублирования проходящих пакетов и не мешает обычной работе маршрутизатора.

В силу ограниченного объёма памяти и вычислительных ресурсов, программное обеспечение маршрутизатора не содержит средств разбора и анализа потоков данных, однако с помощью модуля отбора пакетов можно выполнять подобный анализ на любой рабочей станции сети (даже в реальном времени) с помощью более мощных средств анализа (подобных утилите **tcpdump**).

Практически, каждый пакет (вместе с заголовком MAC уровня), проходящий сквозь интерфейс, пересылается на удалённую рабочую станцию по протоколу UDP на заданный адрес и порт.

Copyright © 2004-2005 by InfiNet Wireless

Не составляет большого труда написать небольшую программу, которая будет принимать такие пакеты и сохранять их в файл или подавать на вход анализатора пакетов.

В качестве прототипа, предлагается исходный текст программы **bpfshow** для системы FreeBSD/OpenBSD. В данном случае, программа запускается с одним аргументом - номером порта и записывает все принимаемые пакеты в файл pcap.raw.

После чего можно проанализировать содержимое потока утилитой **tcpdump**:

bpfshow 8000

^c

tcpdump -r pcap.raw

Примеры:

bpf rf4.0 10.11.12.13 8000

Включает режим отбора пакетов с пересылкой всех пакетов, проходящих через интерфейс rf4.0 на станцию с адресом 10.11.12.13 порт 8000.

bpf rf4.0 –

Отключает режим отбора пакетов на интерфейсе rf4.0

11.Snmpd (SNMP daemon)

Модуль поддержки SNMP.

Синтаксис:

snmpd comm[unity] ключ_доступа

snmpd user имя_пользователя add|set [pass пароль] [sec[urity] noAuthNoPriv|authNoPriv]

snmpd user имя_пользователя del[ete]

snmpd v1disable|v1enable

snmpd start | stop

Описание:

Модуль сетевого протокола управления (Simple Network Management Protocol, SNMP) версии 1 и версии 3.

Поддержка протокола SNMP является важной составляющей любого коммуникационного устройства, поскольку позволяет администратору сети использовать единую систему и механизмы контроля всей сети в целом и каждой её компоненты в отдельности.

Хотя в первой версии SNMP-V1 уделено недостаточно внимания вопросам безопасности самого протокола, что делает весьма проблематичным его использование для управления сетью, однако он широко используется для контроля и анализа функционирования сети. Изменение переменных MIB для первой версии отключено, она по прежнему работает только в режеме read-only. Опция **v1disable** позволяет совсем отключить поддержку первой версии, а также немного ускоряет обработку входящих SNMP-запросов.

Теперь поддерживается и современная версия SNMP-V3 с моделью доступа USM (User-based Security Model) с MD5 аутентификацией, за исключением шифрования. Для организации доступа создаются пользователи с указанием

Copyright © 2004-2005 by InfiNet Wireless

имени, пароля и минимального уровня доступа (с аутентификацией или без нее).

Данная реализация поддерживает MIB-II, а также частные MIB.

Пример:

snmpd comm secret

snmpd user john add pass mypassword security authNoPriv

snmpd on

12.Td (Telnet daemon)

Telnet daemon

Синтаксис:

td enable | disable RemoteHOST

td start | stop | flush

Описание

Telnet сервер позволяет производить удалённую конфигурацию и диагностику работы маршрутизатора и выполнять любые команды операционной системы, также как это делается через консольный порт. Telnet сервер стартует автоматически при включении устройства.

Остановка сервера выполняется командой **td stop**. После этой команды все последующие запросы на установление соединения будут отвергаться. Повторный запуск после остановки выполняется командой **td start**.

По умолчанию, сервер принимает запросы на установление соединения с любых узлов сети (хостов).

С помощью команды "**td enable** RemoteHOST", можно ограничить возможность доступа к системе только с указанных адресов. **RemoteHOST** - это IP-адрес станции с которой разрешён вход в систему. Может быть указано до 10-ти независимых входов.

Для отмены разрешения выполняется команда **td disable** с теми же параметрами.

Команда **td flush** полностью очищает конфигурацию **telnet** сервера.

Пример:

td enable 195.38.44.1

td enable 195.38.44.11

td start

13.Nat (Network Address Translation)

nat - модуль преобразования сетевых адресов (Network Address Translation RFC1631)

Синтаксис:

nat command [arguments]

Описание:

Модуль преобразования сетевых адресов (Network Address Translation) позволяет в какой-то мере решить проблему исчерпания пространства адресов в IPv4 и даёт возможность подключить несколько компьютеров через единственное соединение с официальным IP адресом. Модуль NAT

Copyright © 2004-2005 by InfiNet Wireless

принимает исходящие из локальной сети IP-пакеты, изменяет адрес отправителя на официальный адрес, выделенный провайдером, и повторно отправляет эти пакеты в потоке исходящих данных. NAT делает это меняя IP-адрес отправителя и порт таким образом, что когда данные принимаются обратно, он может определить расположение источника, который запрашивал данные, и переслать их ему. Модуль NAT аналогичен по возможностям (за некоторыми исключениями) модулю natd и библиотеки libalias из FreeBSD. Чтение оригинальной документации может лучше прояснить некоторые вопросы.

Как известно (RFC1918), часть адресного пространства IPv4 зарезервирована для применения в так называемых частных IP сетях (private internets):

10.0.0.0 - 10.255.255.255 (10/8 prefix)

172.16.0.0 - 172.31.255.255 (172.16/12 prefix)

192.168.0.0 - 192.168.255.255 (192.168/16 prefix)

Протоколы маршрутизации глобальной сети Internet не передают информацию об этих адресах, что позволяет использовать одни и те же адреса в разных местах глобальной сети. Частные сети широко используются провайдерами или компаниями для построения внутренней транспортной среды, а также для подключения небольших групп абонентов.

Вполне возможно, что при подключении Вашей локальной сети к Интернет провайдер предложит Вам минимизировать требуемое количество реальных IP адресов с целью экономии своего адресного пространства. Ничего страшного. Реальные (public) IP адреса нужны далеко не всегда. Обычному пользователю Интернет как правило нужен доступ к вполне определённому набору распространённых сервисов, таких как WWW, FTP, ICQ, IRC, Telnet, SMTP, On-line Games и т.д. Всё это будет доступно и при использовании частных IP адресов с помощью NAT. Кроме того, существуют специализированные прокси сервера для конкретных типов сервисов которые гораздо лучше подходят для этой задачи. Например, для HTTP и FTP лучше использовать кэширующий прокси сервер Squid. Впрочем, одно другому не мешает.

Если Вы решите пользоваться системой IP-телефонии на базе страндарта H.323, то Вы также сможете использовать частные IP адреса. В нашу версию NAT включен модуль поддержки стандарта H.323, позволяющий это сделать.

Итак, Вы решили использовать в своей локальной сети частные IP адреса и провайдер выделил Вам ограниченное количество реальных IP адресов. Приступим к настройке модуля NAT.

Команды

Команда local_acl

Синтаксис:

local_acl $NAME [public_addr|dhcp IFNAME] [enable|disable|delete]

Эта команда задаёт реальный (public) IP-адрес, который будет использоваться для трансляции адресов. Для работы протоколов маршрутизации этот адрес должен быть назначен какому-нибудь физическому интерфейсу маршрутизатора. Маршрутизатор Revolution имеет как минимум два физических интерфейса: Ethernet (**eth**) и radio (**rf**). Обычно через радиоинтерфейс блок доступа подключен к опорной сети провайдера, которая, скорее всего, также построена на частных IP адресах. Ethernet используется для подключения вашей локальной сети и, следовательно будет иметь один из частных адресов вашей локальной сети.

Copyright © 2004-2005 by InfiNet Wireless

Куда же назначить реальный адрес ?

Его можно назначить alias-ом на любой из этих интерфейсов или на виртуальный интерфейс null0.

```
ifconfig null0 123.1.1.1/32 up
```

Более того, иногда можно и вовсе не назначать реальный адрес на интерфейсы маршрутизатора. Цель этой операции - обеспечить доступность данного адреса со стороны глобальной сети. Но это можно сделать и с помощью статической маршрутизации. Провайдер может просто описать маршрут на этот адрес так, чтобы пакеты направляющиеся на него попадали на ваш блок доступа. Этого будет достаточно. Наличие связи с интерфейсом необязательно. Модуль NAT выполняет преобразования ещё до того, как система разберётся ей ли адресован пакет, главное, чтобы он попал внутрь маршрутизатора.

Если провайдер выделил Вам небольшой блок адресов, например 123.1.1.0/30, то можно назначить на null0 весь блок, например:

```
ifconfig null0 123.1.1.0/30
```

и использовать все адреса блока по своему усмотрению. Например, в данном случае использовать первый адрес 123.1.1.0 как alias_address, а остальные (включая последний 123.1.1.3) использовать для перенаправления пакетов на локальные машины с помощью команд nat redirect_xxx (см. ниже) или в качестве публичных адресов для других частных сетей.

По отношению к другим компонентам системы (ipfw, ipstat, qm) модуль NAT встроен таким образом, что всегда используются истинные адреса источника и приёмника. То есть, например, при составлении правил ipfw следует оперировать локальными адресами внутренней сети. Они же будут показаны при сборе статистики модулем ipstat.

Также эта команда задаёт имя списка (ACL) ваших локальных сетей, которые требуют трансляции адресов.

Все пакеты с адресами источников попадающими в список local_acl считаются исходящими и подлежат преобразованию. Исключение составляют пакеты проходящие из local_acl в ту же local_acl, а также из local_acl на собственные адреса маршрутизатора. Эти и все остальные пакеты считаются входящими и, если они не являются обратными к уже преобразованным соединениям, пропускаются насквозь без изменения.

```
acl add $NAT net 192.168.1.0/24
nat local_acl $NAT 123.1.1.1
```

В данном случае мы создали список с одной единственной сетью 192.168.1.0/24 (это ваша частная сеть), сослались на него в команде local_acl и назначили сетевой адрес 123.1.1.1 в качестве публичного адреса этой сети.

Можно создать несколько частных сетей и каждой назначить свой собственный публичный адрес, преобразование адресов для разных сетей выполняется независимо.

Также, в качестве публичного адреса можно использовать адрес, полученный по протоколу DHCP. Для этого вместо явного указания публичного IP адреса необходимо указать опцию **dhcp** и интерфейс через который DHCP выдал адрес.

Например:

```
nat local_acl $NAT dhcp eth0
```

Copyright © 2004-2005 by InfiNet Wireless

Команды **enable**, **disable**, **delete** делают рабочей, нерабочей данную запись и удаляют ее соответственно.

Команда alias_address X.X.X.X

В настоящее время эта команда устарела. Пользуйтесь командой local_acl.

Команда maxlinks NUM

Эта команда задаёт максимальное количество поддерживаемых соединений, по умолчанию 1000.

Система сама следит за состоянием соединений и динамически особождает ненужные в зависимости от их типа, времени действия и активности. Однако, при работе различных сканеров сети может возникнуть ситуация, что количество соединений будет расти до бесконечности или пока не кончится оперативная память. С помощью этой команды можно предотвратить бесконтрольное и неограниченное распределение оперативной памяти маршрутизатора. В случае когда количество одновременных соединений превысит установленный предел, система выведет предупреждение в системный журнал и запретит создание новых соединений, пока ситуация не нормализуется. Когда количество соединений вернётся в норму, в системный журнал будет сделана соответствующая запись и работа возобновится.

В принципе, этого достаточно, чтобы запустить NAT.

Команда enable

Эта команда разрешает модулю NAT выполнять трансляцию адресов в соответствии с установленными правилами.

Итак, полный пример:

```
ifconfig null0 123.1.1.1/32 up
rip start  #чтобы работала динамическая маршрутизация для реального
           # адреса
acl add $NAT net 192.168.1.0/24
nat local_acl $NAT 123.1.1.1
nat enable
```

Всё. Можно проверять доступ к интернет из локальной сети.

Команда disable

Эта команда прекращает трансляцию адресов, но сохраняет все ранее введённые параметры NAT.

```
nat disable
```

Команда same_ports yes|no

Эта команда заставляет модуль NAT стремиться оставлять номера портов в преобразуемых пакетах без изменений. Если это становится невозможным, то будут использоваться произвольные номера из числа доступных. Опция включена по умолчанию.

```
nat same_ports no
```

Команда verbose yes|no

Включает режим диагностики и печатает содержимое всех обрабатываемых пакетов и их изменения в системный журнал.

```
nat verbose
```

Copyright © 2004-2005 by InfiNet Wireless

Команда proxy_only yes|no

Если включена, то модуль NAT выполняет только функции перенаправления пакетов заданные командами proxy_rule (см. ниже). Обычная трансляция адресов не выполняется.

Команда stat

Показывает статистику текущей работы со всеми активными каналами преобразования.

nat stat

Перенаправление пакетов

Минусом использования NAT является то, что машины в локальной сети недоступны из Интернет. Клиенты в ЛВС могут выполнять исходящие соединения во внешний мир, но не могут обслуживать входящие. Это является проблемой при запуске служб Интернет на клиентских машинах в локальной сети. Простым решением является перенаправление некоторых портов машины с NAT на клиента локальной сети.

Ниже описаны команды для создания правил перенаправления пакетов (redirect_xxx и proxy_rule). Допускается многократное выполнение команд с различными аргументами, каждая команда добавляет правило в общий список. При просмотре конфигурации config show команды перенаправления пакетов будут пронумерованы. Это позволяет удалять ненужные правила из списка командами nat del XX, где XX порядковый номер правила по конфигурации.

Команда redirect_port arguments

Команда имеет две формы:

redirect_port proto localIP:localPORT[-localPORT]
[aliasIP:]aliasPORT[-aliasPORT]
[remoteIP[:remotePORT[-remotePORT]]]

Перенаправляет входящие соединения поступающие на заданный порт(-ы) на другой адрес и другой порт(-ы).

Аргумент **proto** может быть **tcp**, **udp**, **ras** или **cs**.

В случае **ras** или **cs** осуществляется изменение адресов в соответствии со стеком H.323.

targetPORT - заданный порт или диапазон портов на данной машине.

aliasPORT - порт назначения (или номера портов).

Диапазоны портов **aliasPORT** и **targetPORT** могут не совпадать в числах, но обязаны быть одинакового размера.

В случае одновременного использования нескольких пар публичный адрес - частная сеть рекомендуется указывать конкретный публиный адрес.

Параметры **remoteIP** и **remotePORT** могут задаваться для более точного определения входящих соединений (будут обрабатываться только пакеты приходящие с указанного адреса и порта). Если **remotePORT** не указан, то подразумеваются любые порты. Если **remotePORT** указан, то его размер должен совпадать с диапазоном **targetPORT**.

nat redirect_port tcp 192.168.1.5:23 7777

Copyright © 2004-2005 by InfiNet Wireless

В этом примере все входящие **tcp** соединения на порт 7777 данного маршрутизатора будут перенаправляться на машину 192.168.1.5 порт 23 (telnet).

nat redirect_port tcp 192.168.1.4:2300-2399 123.1.1.2:3300-3399

Все входящие **tcp** пакеты с портами назначения в диапазоне 3300-3399 и адресом назначения 123.1.1.2 будут перенаправляться на машину 192.168.1.4

Преобразование портов выполняется 1:1, то есть 3300->2300, 3301->2301 и т.д.

Пусть, к примеру, сервер IRC запущен на клиенте A, а веб-сервер работает на клиенте B. Чтобы это работало, соединения, принимаемые на портах 6667 (irc) и 80 (веб), должны перенаправляться на соответствующие машины:

nat redirect_port tcp 192.168.0.2:6667 6667

nat redirect_port tcp 192.168.0.3:80 80

Вторая форма команды:

redirect_port proto localIP:localPORT[,localIP:localPORT[,...]]
[aliasIP:]aliasPORT
[remoteIP[:remotePORT]]

Эта форма команды, также как и redirect_address выполняет циклическую пересылку входящих пакетов по нескольких адресам (LSNAT) для распределения нагрузки между ними, например:

redirect_port tcp 192.168.1.2:80, 192.168.1.3:80, 192.168.1.4:80 123.1.1.2:80

В данном случае все запросы к WEB серверу 123.1.1.2 будут перенаправляться на сервера локальной сети.

Команда redirect_address localIP [,LocalIP, ...] publicIP

Перенаправляет весь входящий трафик поступающий на адрес **publicIP** на адрес **localIP**.

Если указано несколько адресов localIP, то перенаправление будет выполняться поочерёдно на каждый адрес по кругу.

nat redirect_address 192.168.1.2 192.1.1.1

nat redirect_address 192.168.1.3 192.1.1.2

В данном случае весь трафик приходящий на адрес 192.1.1.1 будет перенаправляться в локальную сеть на адрес 192.168.1.2, а трафик приходящий на адрес 192.1.1.2 пересылается на машину 192.168.1.3

Перенаправление адреса полезно, если имеется несколько адресов IP, и они должны быть на одной машине. В этой ситуации NAT может назначить каждому клиенту ЛВС свой собственный внешний IP-адрес. Затем NAT преобразует исходящие от клиентов локальной сети пакеты, заменяя IP-адреса на соответствующие внешние, и перенаправляет весь трафик, входящий на некоторый IP-адрес, обратно конкретному клиенту локальной сети. Это также называют статическим NAT. К примеру, пусть IP-адреса 128.1.1.1, 128.1.1.2 и 128.1.1.3 принадлежат шлюзовой машине. 128.1.1.1 может использоваться в качестве внешнего IP-адреса шлюзовой машины, когда как 128.1.1.2 и 128.1.1.3 будут перенаправляться обратно к клиентам ЛВС A и B.

Copyright © 2004-2005 by InfiNet Wireless

nat redirect_address 192.168.1.2 128.1.1.2

nat redirect_address 192.168.1.3 128.1.1.3

Внешние IP-адреса машины с NAT должны быть активизированы и являться алиасами для внешнего интерфейса, либо описаны иным способом.

Команда redirect_proto proto localIP [publicIP [remoteIP]]

Перенаправляет все входящие пакеты с типом протокола proto на машину с адресом localIP.

nat redirect_proto 47 192.168.1.2

Если publicIP не указан, используется значение параметра alias_address для соотвествующей localIP сети.

Если указан remoteIP, то обрабатываются только пакеты с этого адреса.

Команда default_h323 [yes|no]

Включает изменение адресов в соответствии со стеком H.323 для исходящих соединений. Обрабатываются все UDP-пакеты, исходящие на порт 1719, и TCP-потоки, исходящие на порт 1720.

По умолчанию выключена.

***Внимание**: не стоит без необходимости включать эту опцию, так как с этой опцией NAT будет обрабатывать все UDP-пакеты, исходящие на порт 1719, и TCP-потоки, исходящие на порт 1720, что вызовет бесполезное замедление работы NAT при использовании этих портов и протоколов не для H.323 телефонии.*

Команда h323_destination ras|cs remote_addr[:remote_port] [local_addr[:local_port]]

Позволяет более детально описать использование H.323 элементов во внешней сети, осуществляя обработку потоков H.323.

- **ras|cs** - слой стека H.323, для которого будет выполняться обработка
- **remote_addr** - адрес внешней сети, исходящие соединения на который будут обрабатываться
- **remote_port** - порт, исходящие соединения на который будут обрабатываться. Если порт не указан, то используется порт 1719 для ras и 1720 для cs
- **local_addr** - адрес внутренней сети, исходящие соединения с которого будут обрабатываться. Если адрес не указан, то обрабатываются соединения, исходящие с любого адреса
- **local_port** - порт, исходящие соединения с которого будут обрабатываться. Если порт не указан, то обрабатываются соединения, исходящие с любого порта

Команда proxy_rule

Синтаксис:

proxy_rule параметр значение [параметр значение] ...

Перенаправление исходящих пакетов. TCP пакеты исходящие из локальной сети на любой адрес с указанным портом перенаправляются на заданный сервер и порт. Опционально первоначальный адрес назначения может быть включён в пакет несколькими способами. Командная строка состоит из

одной или более пар слов, разделённых пробелами. Первое слово является ключевым параметром, второе его значением.

Доспустимые параметры и их описание:

type encode_ip_hdr | **encode_tcp_stream** | **no_encode**

Если для организации прозрачного шлюза необходимо передавать информацию о первоначальном адресе и порте доступа на новый сервер, то это может быть сделано двумя путями:

- Если указана опция **encode_ip_hdr**, то оригинальный адрес и порт передаются в расширенных полях IP заголовка (IP option)
- Если указана опция **encode_tcp_stream**, то оригинальный порт и адрес передаются в пакете перед началом данных в формате "DEST IP port".

port portnum

Обрабатываются только пакеты отправленные на указанный порт.

server host[:portnum]

Обязательный параметр. Задаёт адрес сервера и порт на который будут перенаправляться пакеты. Если порт не указан, то будет использоваться оригинальный порт назначения.

proto tcp | udp

Если указано, то будут обрабатываться только пакеты с заданным протоколом.

src IP[/bits]

dst IP[/bits]

Необязательное. Задаёт адрес (подсеть) источника или назначения для которых следует выполнять перенаправление пакетов.

Пример:

nat proxy_rule proto tcp port 80 server 123.1.1.1:3128

В данном примере все TCP пакеты исходящие из локальной сети на порт 80 будут перенаправляться на прокси сервер провайдера.

Команда del rule_number

Удаляет правило с номером rule_number.

NAT и H.323 телефония

Абоненты и гейткиперы используют несколько протоколов стека H.323. Нас интересуют два из них - RAS (registration, admission, status), используемый для регистрации абонентов на гейткипере и мониторинга статуса абонентов, и CS (call signalling), используемый абонентами для сигнализации в пределах одного звонка. Оба эти протокола описаны в стандарте H.225.0. Распространенные конфигурации систем включают в себя следующие примеры (везде под словом "абонент" понимается шлюз или собственно абонент):

1. Абонент находится в частной сети, шлюз находится по реальному адресу, звонки абонента - только исходящие. Для организации доступа абонента из частной сети к шлюзу можно воспользоваться правилом h323_destination с протоколом cs. Если шлюз принимает звонки на стандартный порт 1720, достаточно будет включить режим default_h323.

Copyright © 2004-2005 by InfiNet Wireless

Пример: Абонент находится в частной сети по адресу 10.0.0.99, шлюз - во внешней сети по адресу 123.45.67.89. Необходимо, чтобы абонент мог совершать исходящие звонки на шлюз. Решение - задать правило:

nat h323_destination cs 123.45.67.89 10.0.0.99

Пример 2: Абонент находится в частной сети по адресу 10.0.0.99, шлюз или несколько шлюзов - во внешней сети по неизвестным заранее адресам. Необходимо, чтобы абонент мог совершать исходящие звонки на шлюзы. Решение - включить опцию default_h323:

nat default_h323

2. Несколько абонентов находятся в частной сети, шлюз находится по реальному адресу, звонки - исходящие и входящие. Для доступа шлюза к абонентам нужно использовать настройку redirect_port с указанным протоколом cs, разными для разных абонентов alias адресами или портами (и прописать alias адреса и порты в конфигурации шлюза), а также явно указать адрес и порт шлюза и адреса абонентов (можно указать и порты абонентов).

Пример: Абоненты находятся в частной сети по адресам 10.0.0.98 и 10.0.0.99, шлюз - во внешней сети по адресу 123.45.67.89. Alias_address NAT пусть будет 123.45.67.65. Необходимо, чтобы абоненты могли совершать исходящие звонки на шлюз и принимать звонки, входящие со шлюза. Решение - задать правила:

nat redirect_port cs 10.0.0.98:1720 1720 123.45.67.89

nat redirect_port cs 10.0.0.99:1720 1721 123.45.67.89

В конфигурации шлюза надо будет указать адреса абонентов как 123.45.67.65:1720 и 123.45.67.65:1721 соответственно.

3. Абонент из частной сети регистрируется на гейткипере с реальным адресом и работает через гейткипер. В этом случае достаточно задать правило вида h323_destination ras адрес_гейткипера. Если абоненты регистрируются по стандартному для регистрации порту 1719, тогда можно просто включить режим default_h323.

Пример: Абонент находится в частной сети по адресу 10.0.0.99, а гейткипер во внешней сети по адресу 123.45.67.89. Необходимо, чтобы абонент мог зарегистрироваться на этом гейткипере и совершать и принимать через него звонки. Решение - задать правило:

nat h323_destination ras 123.45.67.89 10.0.0.99

Пример 2: Некоторое число абонентов находится в частной сети, а гейткипер во внешней сети по адресу 123.45.67.89 и нестандартному для RAS порту 1024. Необходимо, чтобы любой абонент мог зарегистрироваться на этом гейткипере и совершать и принимать через него звонки. Решение - задать правило:

nat h323_destination ras 123.45.67.89:1024

Пример 3: Абонент находится в частной сети по адресу 10.0.0.99, гейткипер или несколько гейткиперов - во внешней сети по неизвестным заранее адресам. Необходимо, чтобы абонент мог зарегистрироваться на любом гейткипере и совершать и принимать через него звонки. Решение - включить опцию default_h323:

nat default_h323

4. Абонент с реальным адресом регистрируется на гейткипере из частной сети. Здесь нужно задать правило redirect_port с указанным протоколом ras и указать в нем частный адрес и RAS порт гейткипера, чтобы абоненты из

Copyright © 2004-2005 by InfiNet Wireless

внешней сети смогли зарегистрироваться на этом гейткипере. Для того, чтобы static абоненты тоже смогли работать с гейткипером, нужно будет дополнительно задать правило redirect_port с указанным протоколом cs и указать в нем частный адрес и CS порт гейткипера.

Пример: Абонент находится во внешней сети по адресу 123.45.67.89, а гейткипер в частной сети по адресу 10.0.0.99. Необходимо, чтобы абонент мог зарегистрироваться на этом гейткипере и совершать и принимать через него звонки. Alias_address NAT пусть будет 123.45.67.65. Решение - задать правило:

nat redirect_port ras 10.0.0.99:1719 1719 123.45.67.89

В конфигурации абонента RAS адрес гейткипера должен будет выглядеть как 123.45.67.65:1719.

Пример 2: Static абонент находится во внешней сети по адресу 123.45.67.89, а гейткипер в частной сети по адресу 10.0.0.99. Необходимо, чтобы абонент мог совершать и принимать через него звонки. Alias_address NAT пусть будет 123.45.67.65. Решение - задать правило:

nat redirect_port cs 10.0.0.99:1720 1720 123.45.67.89

В конфигурации абонента CS адрес гейткипера должен будет выглядеть как 123.45.67.65:1720.

14.Trapd (SNMP traps)

Модуль поддержки SNMP traps

Синтаксис:

trapd agent X.X.X.X dstaddr X.X.X.X

trapd start | stop

Описание:

Протокол SNMP предоставляет возможность сетевому агенту посылать асинхронные уведомления (**traps**) о возникновении того или иного события на контролируемом устройстве (объекте).

Модуль **trapd** осуществляет централизованную рассылку информации поступающей от внутренних подсистем маршрутизатора на указанный SNMP сервер.

Адрес SNMP сервера указывается командой "**trapd dstadd X.X.X.X**", (UDP порт 162).

Собственный адрес агента, передаваемый в пакете SNMP-trap, задаётся командой "**trapd agent X.X.X.X**", по умолчанию 127.0.0.1

Пример:

trapd dstaddr 192.168.1.1

trapd start

Copyright © 2004-2005 by InfiNet Wireless

15.DHCP Server

Командный язык DHCP сервера

Команды конфигурации или просмотра текущего состояния сервера вводятся с консоли или TELNET клиента WANFlex. Префикс команд управления DHCP сервером в командном интерпретаторе WANFlex – **dhcpd**. Полный список команд (без префикса):

Синтаксис:

add scope <SCOPE_NAME> <INTERFACE|> <START_IP> <END_IP>*
add virtual interface <GATEWAY>
clear
delete option <OPTION_NAME>
delete scope <SCOPE_NAME>
delete virtual interface <GATEWAY>
interface <INTERFACE> delete option <OPTION_NAME>
interface <INTERFACE> option <OPTION_NAME> <OPTION_VALUE>
interface <INTERFACE> reservation
<CLIENT_ID> delete option <OPTION_NAME>
interface <INTERFACE> reservation
<CLIENT_ID> option <OPTION_NAME> <OPTION_VALUE>
interface <INTERFACE|> show boundhistory*
interface <INTERFACE|> show client <CLIENT_ID|*>*
lock interface <INTERFACE>
option <OPTION_NAME> <OPTION_VALUE>
scope <SCOPE_NAME> add classid <CLIENT_CLASS_ID>
scope <SCOPE_NAME> add exclude <START_IP> <END_IP>
scope <SCOPE_NAME> add reservation <CLIENT_ID> <CLIENT_IP>
scope <SCOPE_NAME> delete classid <CLIENT_CLASS_ID>
scope <SCOPE_NAME> delete exclude <START_IP>
scope <SCOPE_NAME> delete option <OPTION_NAME>
scope <SCOPE_NAME> delete reservation <CLIENT_ID>
scope <SCOPE_NAME> interface <INTERFACE|>*
scope <SCOPE_NAME> option <OPTION_NAME> <OPTION_VALUE>
scope <SCOPE_NAME> reservation
<CLIENT_ID> delete option <OPTION_NAME>

scope <SCOPE_NAME> reservation
<CLIENT_ID> option <OPTION_NAME> <OPTION_VALUE>
scope <SCOPE_NAME> set range <START_IP> <END_IP>
scope <SCOPE_NAME|> show declinehistory*
show config
show interface <INTERFACE|>*
show options
show scope <NAME|>*
show unleases <SUBSTR|>*
show version
start
stop
unlock interface <INTERFACE>
virtual interface <GATEWAY> add
subnet <IP_ADDRESS> <SUBNET_MASK>
virtual interface <GATEWAY> delete
subnet <IP_ADDRESS> <SUBNET_MASK>

Команды нечувствительны к регистру букв и допускают произвольное сокращение вплоть до одной буквы, если команда в целом не может быть воспринята DHCP сервером неоднозначно.

Copyright © 2004-2005 by InfiNet Wireless

Например, команда **dhcpd show scope *** может быть сокращена до **dhcpd s s** *, а команда **dhcpd show config** до **dhcpd sh c**. Команды позволяющие изменять конфигурацию DHCP сервера, а также его старта или остановки, требуют, чтобы администратор имел в данной системе WANFlex права *super-user*. Другие команды DHCP сервер будет выполнять для любого пользователя.

В приведенном списке команд их параметры условно изображены в скобках <>. Если значение какого-либо параметра содержит пробелы, то это значение должно указываться в кавычках.

Например:

*#2>**dhcpd scope** MSOFT **add classid** "MSFT 5.0"*

или

*#2>**dhcpd add scope** "Micro Soft" eth0 9.1.1.201 9.1.1.250*

Внимание! DHCP исполняет команды только после своего запуска:

dhcpd start

Клиенты DHCP сервера (CLIENT)

Протокол DHCP используется для динамической настройки основных параметров подключения хостов (рабочих станций и серверов) к сети TCP/IP. Протокол DHCP использует в качестве транспортного протокол UDP/IP. Хост, запрашивающий данные о конфигурации своего подключения к сети (ip адрес, маску подсети, основной шлюз и др.) называется клиентом DHCP. Основным параметром конфигурации для клиента является IP адрес. После старта, клиент рассылает по сети специальные DHCP запросы о предоставлении ему в аренду IP адреса и получения прочих параметров сетевой конфигурации. Для своей идентификации клиент в этих запросах может указывать *идентификатор клиента*. В общем случае, идентификатор клиента – это двоичный набор байтов, уникальный в пределах физического сегмента сети, к которой этот клиент подключен. Если клиент не сообщает свой идентификатор, то за его идентификатор сервер примет физический адрес (MAC) сетевого интерфейса клиента. Т.о. в DHCP сервере клиент идентифицируется своим идентификатором клиента и сетевым интерфейсом с которого сервер принимает запросы от данного клиента (*интерфейс клиента*). Идентификатор клиента (параметр <CLIENT_ID> в командах) представляется в виде ID:<идентификатор>|01:<MAC адрес сетевого адаптера клиента>.

Например:

`ID:01:00:04:35:22:88:1D.`

В своих запросах серверу, клиент может сообщать свой класс (идентификатор класса). Идентификатор класса – это строка символов, которая определяет какое-либо свойство клиента, общее для некоторого множества однотипных. Это может быть, например, название операционной системы клиента или краткое название продукта. Например, DHCP клиенты, работающие под управлением ОС Windows XP, передают в качестве класса клиента “MSFT 5.0”, а IP телефоны Infinet Wireless – “IW_IP_PHONE”. Класс клиента может быть использован администратором сервера для автоматической группировки клиентов в необходимые пулы адресов для удобства назначения специфических параметров (опций) конфигурации.

Пулы адресов (SCOPE)

Пул (scope) - это диапазон IP адресов в пределах которого сервер имеет право выделять IP адреса клиентам. Пулы хранятся в конфигурационной базе данных сервера и идентифицируются в ней своим именем,

Copyright © 2004-2005 by InfiNet Wireless

определенным администратором сервера при создании этого пула. Пул создается командой сервера:

Синтаксис:

dhcpd add scope <SCOPE_NAME> <INTERFACE|>*
<START_IP> <END_IP>

где

- *SCOPE_NAME* – это имя пула. Имя пула безразлично к регистру и должно быть выбрано уникальным. Если имя пула содержит пробелы, сервер автоматически заменит их на знак «подчеркивание» (_)
- *INTERFACE* – имя сетевого интерфейса, к которому этот пул может подключаться (разрешенный интерфейс). Если вместо имени интерфейса указать *, то это значит, что этому пулу разрешено подключаться к любому подходящему сетевому интерфейсу. Подходящий сетевой интерфейс – это интерфейс, который содержит такую подсеть IP адресов (alias), к которой принадлежат и стартовый и конечный IP адрес этого пула. Если же при создании пула указано имя конкретного интерфейса, то при попытке подключиться, пул будет определять подходящую подсеть только среди подсетей этого интерфейса. Иными словами: пул подключается только к разрешенному интерфейсу, содержащему подсеть к которой пул принадлежит полностью.
- *START_IP* и *END_IP* – начальный и конечный IP адрес пула соответственно. При подключении к сетевому интерфейсу, проверяется что диапазон подключающегося пула не пересекается (и не принадлежит к) с диапазоном уже подключенного ранее к этому интерфейсу пула. При выделении IP адресов клиентам, рассматриваются только те пулы, которые подключены к сетевому интерфейсу, через который принимаются запросы от данного клиента. Т.е. только пулы подключенные к интерфейсу клиента.

В любом случае, если даже пул по каким либо причинам не может быть подключен, он не уничтожается.

Например:

*#2> **dhcpd add scope** MSOFT eth0 192.168.177.20 192.168.177.22*

[eth0] <192.168.177.12> (MSOFT):
192.168.177.20-192.168.177.22 Scope attached

OK

Здесь был успешно создан пул с именем MSOFT с разрешенным интерфейсом eth0. После создание пул автоматически подключился к интерфейсу eth0 к подсети 192.168.177.12

*#2> **dhcpd add scope** new * 10.12.12.30 10.12.12.50*

WRN: Scope created, but not attached.

Здесь пул под названием new был создан с разрешением подключаться к любому интерфейсу. Пул был успешно создан, но не смог подобрать интерфейса для подключения, т.к. ни один из интерфейсов не имеет подсети, в которую бы диапазон адресов этого пула помещался бы полностью.

Copyright © 2004-2005 by InfiNet Wireless

Изменить диапазон адресов ранее созданного пула можно командой:

Синтаксис:

dhcpd scope *<SCOPE_NAME>* ***set range*** *<START_IP> <END_IP>*

где

- *SCOPE_NAME* – это имя пула, диапазон которого требуется изменить
- *START_IP* и *END_IP* – новый начальный и конечный IP адрес пула соответственно.

Изменить разрешенный для подключения пула интерфейс можно командой:

Синтаксис:

scope *<SCOPE_NAME>* ***interface*** *<INTERFACE|*>*

где

- *SCOPE_NAME* – это имя пула, для которого требуется изменить разрешенный интерфейс
- *INTERFACE* – имя сетевого интерфейса разрешенного для данного пула. Если в системе не существует интерфейса с таким именем или у этого интерфейса не будет условий для подключения данного пула, то пул немедленно отключится. Это свойство можно использовать для временного отключения какого либо пула, если изменить имя разрешенного ему интерфейса на заведомо несуществующее.

Например:

#2> dhcpd scope OTHER interface -eth0

[eth0] <192.168.177.12> (OTHER):
192.168.177.10-192.168.177.19 Scope detached
OK

Таким образом мы отключили пул OTHER. Когда понадобится снова его подключить:

#2> ***dhcpd scope*** *OTHER* ***interface eth0*** *(или *)*
[eth0] <192.168.177.12> (OTHER):
192.168.177.10-192.168.177.19 ***Scope attached***
OK

В пул адресов могут быт внесены исключения (*exclude*). Это диапазоны адресов, принадлежащие пулу, но запрещенные для выдачи в аренду клиентам. Исключения вносятся командой сервера:

Синтаксис:

dhcpd scope *<SCOPE_NAME>* ***add exclude*** *<START_IP> <END_IP>*

где

- *SCOPE_NAME* – это имя пула в который вносится диапазон исключения
- *START_IP* и *END_IP* – стартовый и конечный адреса диапазона исключения соответственно. При внесении исключения, его диапазон не должен пересекаться (или принадлежать к) ни с одним из диапазонов исключения, ранее внесенным в данный пул. Вносимый диапазон исключения должен принадлежать пулу. Удаление ранее внесенного исключения производится командой сервера:

Синтаксис:

dhcpd scope *<SCOPE_NAME>* ***delete exclude*** *<START_IP>*

Copyright © 2004-2005 by InfiNet Wireless

Параметры этой команды аналогичны параметрам команды внесения диапазона исключения, за исключением того, что достаточно указать начальный адрес ранее внесенного диапазона исключения, который необходимо удалить.

Внимание! При выполнении команды dhcpd scope <SCOPE_NAME> set range <START_IP> <END_IP>, те ранее созданные в пуле исключения, которые перестают удовлетворять устанавливаемому диапазону, будут удалены из пула автоматически!

Фильтр классов клиентов (CLASSID)

Пулы адресов имеют фильтр классов клиентов. Если клиент в запросе сообщает свой класс, то сервер может выделить IP адрес для этого клиента только из таких пулов, подключенных к интерфейсу клиента, у которых в фильтре классов имеется сообщаемый класс. Фильтр представляет собой список идентификаторов классов клиентов, для которых разрешено выделять из этого пула адреса в аренду. Чтобы сформировать у пула фильтр классов, необходимо добавить в список классов пула один или несколько идентификаторов классов клиентов (client vendor class id), которым он будет вправе выдавать IP адреса из своего диапазона в аренду. Для добавления каждого класса существует команда:

Синтаксис:

scope *<SCOPE_NAME>* ***add classid*** *<CLIENT_CLASS_ID>*

где

- *SCOPE_NAME* – это имя пула в который добавляется идентификатор класса клиента со значением *CLIENT_CLASS_ID*
- *CLIENT_CLASS_ID* – это набор символов произвольной длины (не более 255). Если этот параметр должен содержать пробелы, то необходимо указывать его в командной строке в кавычках. Класс клиента сопоставляется с тем, который сообщает клиент в пакетах запроса аренды IP адреса. В случае, если клиент сообщил такой свой класс, которого нет ни у одного пула в своем фильтре, или клиент вовсе не сообщил свой класс, то для выделения ему IP адреса сервер в праве использовать только такие пулы, у которых фильтр классов пуст.

Для удаления класса из фильтра пула существует команда:

Синтаксис:

scope *<SCOPE_NAME>* ***delete classid*** *<CLIENT_CLASS_ID>*

Сетевые интерфейсы (INTERFACE)

Сетевой интерфейс – физический или VLAN сетевой адаптер, зарегистрированный в ядре ОС WANFlex. Сервер при старте самостоятельно определяет набор сетевых интерфейсов, пригодных для обслуживания DHCP клиентов. Пригодным считается интерфейс, подключенный к сети множественного доступа, поддерживающие широковещательную передачу broadcast (BMA), в том числе и виртуальный (VLAN). В базе данных сервера интерфейс идентифицируется своим именем, которое было присвоено интерфейсу при регистрации в ядре ОС WANFlex. Для просмотра информации о интерфейсах существует команда:

Синтаксис:

show interface *<INTERFACE|*>*

Copyright © 2004-2005 by InfiNet Wireless

где

- *INTERFACE* – это имя сетевого интерфейса, информацию о котором требуется распечатать. Если вместо имени интерфейса указать *, то будет выведена информация о всех интерфейсах. Информация выводится в виде структурированного списка, например:

Пример:

```
#2> dhcpd show interface *
        >INTERFACES
        [eth0] UP
         <SUBNET> 9.1.1.100/255.255.255.0
              <SCOPE> (PHONES) 9.1.1.151 - 9.1.1.200
         <SUBNET> 192.168.177.12/255.255.255.0
              <SCOPE> (OTHER) 192.168.177.10 - 192.168.177.19
              <SCOPE> (MSOFT) 192.168.177.20 - 192.168.177.22
        [vlan0] DOWN
         <SUBNET> 192.168.178.1/255.255.255.0
        OK
```

Здесь видно, что сервер обслуживает два сетевых интерфейса с именами ***eth0*** и ***vlan0. eth0*** включен (UP) и у него две IP подсети. К одной из этих подсетей в данный момент подключился пул с именем PHONES, а к другой подключились два пула: OTHER и MSOFT**.** К интерфейсу **vlan0** не может подключится ни один из пулов, т.к. интерфейс переведен системным администратором в положение ВЫКЛЮЧЕН (DOWN).

При необходимости можно запретить DHCP серверу использовать тот или иной интерфейс. Для этого можно подать команду

Синтаксис:

```
lock interface <INTERFACE>
```

где

- *<INTERFACE>* - имя интерфейса. При этом сервер отключит те пулы, которые были подключены на момент подачи этой команды и не будет допускать подключение других.

Например:

```
#2> dhcpd show interface *
        >INTERFACES
        [eth0] UP
         <SUBNET> 9.1.1.100/255.255.255.0
              <SCOPE> (PHONES) 9.1.1.151 - 9.1.1.200
         <SUBNET> 192.168.177.12/255.255.255.0
              <SCOPE> (OTHER) 192.168.177.10 - 192.168.177.19
              <SCOPE> (MSOFT) 192.168.177.20 - 192.168.177.22
        [vlan0] DOWN
         <SUBNET> 192.168.178.1/255.255.255.0
        OK
```

В этом примере у DHCP сервера есть два интерфейса: eth0 и vlan0. При этом vlan0 был отключен командой WANFlex ifconfig vlan0 down. Eth0 был включен и к нему подключились три пула: phones, other и msoft. При этом phones подключился к подсети 9.1.1.100/255.255.255.0, а остальные к 192.168.177.12/255.255.255.0. Допустим, мы хотим заблокировать использование интерфейса eth0 DHCP сервером:

Например:

```
#2> dhcpd lock interface eth0
        [eth0] <9.1.1.100> (PHONES):
          9.1.1.151-9.1.1.200   Scope detached
```

Copyright © 2004-2005 by InfiNet Wireless

```
[eth0] <192.168.177.12> (OTHER):
  192.168.177.10-192.168.177.19   Scope detached
[eth0] <192.168.177.12> (MSOFT):
  192.168.177.20-192.168.177.22   Scope detached
OK
```

После блокировки вновь посмотрим состояние интерфейсов DHCP сервера

```
#2> dhcpd show interface *
        >INTERFACES
        [eth0] UP LOCKED
         <SUBNET> 9.1.1.100/255.255.255.0
         <SUBNET> 192.168.177.12/255.255.255.0
        [vlan0] DOWN
         <SUBNET> 192.168.178.1/255.255.255.0
        OK
```

Видно, что теперь интерфейс eth0 заблокирован. При этом от него отключились все пулы.

Отменить запрет можно командой:

Синтаксис:

```
dhcpd unlock interface <INTERFACE>
```

Например:

```
#2> dhcpd unlock interface eth0
        [eth0] <192.168.177.12> (MSOFT):
          192.168.177.20-192.168.177.22   Scope attached
        [eth0] <192.168.177.12> (OTHER):
          192.168.177.10-192.168.177.19   Scope attached
        [eth0] <9.1.1.100> (PHONES):
          9.1.1.151-9.1.1.200   Scope attached
        OK

#2> dhcpd show interface *
        >INTERFACES
        [eth0] UP
         <SUBNET> 9.1.1.100/255.255.255.0
              <SCOPE> (PHONES) 9.1.1.151 - 9.1.1.200
         <SUBNET> 192.168.177.12/255.255.255.0
              <SCOPE> (OTHER) 192.168.177.10 - 192.168.177.19
              <SCOPE> (MSOFT) 192.168.177.20 - 192.168.177.22
        [vlan0] DOWN
         <SUBNET> 192.168.178.1/255.255.255.0
        OK
```

Резервации пулов (scope reservation)

Главная цель резервации пула – зафиксировать IP адрес выделяемый для определенного клиента из данного пула. Для создания резервации пула существует команда:

Синтаксис:

```
dhcpd scope <SCOPE_NAME> add
        reservation <CLIENT_ID> <CLIENT_IP>
```

где

- *SCOPE_NAME* – это имя пула в который добавляется резервация.
- ***CLIENT_ID*** – идентификатор клиента,

Copyright © 2004-2005 by InfiNet Wireless

- ***CLIENT_IP*** – IP адрес, который будет выдаваться этому клиенту. Резервации пулов хранятся в конфигурационной базе данных сервера и идентифицируются в ней названием пула плюс идентификатор клиента.

Например:

*#2>**dhcpd scope** PHONES **add reservation***
ID:01:00:04:35:00:22:23 9.1.1.170
OK

Т.о., если через интерфейс, к которому подключен пул PHONES поступит DHCP запрос от клиента ID:01:00:04:35:00:22:23, то сервер обязательно выдаст ему IP адрес 9.1.1.170. При этом IP адрес резервации обязан быть из диапазона пула, но на нее не действует правило исключений пула. Т.е. даже если бы у пула было задано исключение (например, dhcpd scope PHONES add exclude 9.1.1.165 9.1.1.175) к которому принадлежит адрес резервации, сервер все равно создаст такую резервацию. Если при добавлении резервации обнаружится, что уже существует резервация для такого клиента в каком-либо другом пуле, подключенном к данному интерфейсу, то сервер не создаст такой резервации и сообщит об ошибке.

#1> dhcpd scope other add reservation
ID:01:00:04:35:00:22:23 192.168.177.10

[eth0] <192.168.177.12> (OTHER):
192.168.177.10-192.168.177.19 Reservation for "ID:01:00:04:35:00:22:23" already exists in scope PHONES with IP=9.1.1.170

ERR: Reservation's IP is out of scope's range

Кроме того, что резервация может занимать IP адреса из диапазонов исключений пулов, она не подчиняется правилу фильтра классов клиентов.

Рассмотрим ситуацию:

```
#2> dhcpd show scope *
          >SCOPES:
          (MSOFT)         192.168.177.20  - 192.168.177.22  [eth0]
          ATTACHED [eth0] <192.168.177.12>/255.255.255.0
           <CLIENT CLASS IDs>: "IW_BRI_GATEWAY" "MSFT 5.0"
           <CLIENT> ID:01:00:C0:DF:10:AF:69 "MSFT 5.0"      'wad     '
          192.168.177.20  <BOUND>    since 01/01/2003 05:01:08
           <FREE RANGE>   192.168.177.21  - 192.168.177.22   =2


          (NEW)           10.12.12.30     - 10.12.12.50     [*]

          (OTHER)         192.168.177.10  - 192.168.177.19  [eth0]
          ATTACHED [eth0] <192.168.177.12>/255.255.255.0
           <CLIENT> ID:01:00:05:90:02:1F:C8 ""              '        '
          192.168.177.10  <BOUND>    since 01/01/2003 05:34:24
           <FREE RANGE>   192.168.177.11  - 192.168.177.11   =1
           <FREE RANGE>   192.168.177.13  - 192.168.177.19   =7

          (PHONES)        9.1.1.151       - 9.1.1.200       [*] ATTACHED
          [eth0] <9.1.1.100>/255.255.255.0
           <CLIENT CLASS IDs>: "IW_IP_PHONE"
           <CLIENT> ID:01:00:04:35:00:22:24 "IW_IP_PHONE"   'Stas    '
          9.1.1.151       <BOUND>    since 01/01/2003 05:00:34
           <FREE RANGE>   9.1.1.152       - 9.1.1.169        =18
           <RESERV> ID:01:00:04:35:00:22:23 "IW_IP_PHONE"   'Andrew  '
          9.1.1.170       <BOUND>    since 01/01/2003 05:49:35
           <FREE RANGE>   9.1.1.171       - 9.1.1.200        =30
           <OPTION>       Router  9.1.1.3
           <OPTION>       H323_GK_ADDRESS 195.38.45.84

          OK
```

Здесь видно, что клиент **ID:01:00:05:90:02:1F:C8** в DHCP запросе не указал свой класс (“”), и поэтому, сервер отдал право выделять ему IP адрес пулу OTHER, подключенному к подсети 192.168.177.12/255.255.255.0

Copyright © 2004-2005 by InfiNet Wireless

интерфейса eth0, т.к. у него отсутствует фильтр классов клиентов. С другой стороны, администратору необходимо, чтобы клиент **ID:01:00:05:90:02:1F:C8** получал IP адрес и дополнительные параметры конфигурации из пула PHONES. Для этого в пуле PHONES мы создадим резервацию для этого клиента с IP адресом, например, 9.1.1.200:

```
#2> dhcpd scope PHONES add reservation
                ID:01:00:05:90:02:1F:C8 9.1.1.200
          OK

#2> dhcpd show scope *
          >SCOPES:
          (MSOFT)         192.168.177.20  - 192.168.177.22  [eth0]
          ATTACHED [eth0] <192.168.177.12>/255.255.255.0
           <CLIENT CLASS IDs>: "IW_BRI_GATEWAY" "MSFT 5.0"
           <CLIENT> ID:01:00:C0:DF:10:AF:69 "MSFT 5.0"      'wad     '
          192.168.177.20  <BOUND>    since 01/01/2003 05:01:08
           <FREE RANGE>   192.168.177.21  - 192.168.177.22   =2

          (NEW)           10.12.12.30     - 10.12.12.50     [*]

          (OTHER)         192.168.177.10  - 192.168.177.19  [eth0]
          ATTACHED [eth0] <192.168.177.12>/255.255.255.0
           <FREE RANGE>   192.168.177.10  - 192.168.177.11   =2
           <FREE RANGE>   192.168.177.13  - 192.168.177.19   =7

          (PHONES)        9.1.1.151       - 9.1.1.200       [*] ATTACHED
          [eth0] <9.1.1.100>/255.255.255.0
           <CLIENT CLASS IDs>: "IW_IP_PHONE"
           <CLIENT> ID:01:00:04:35:00:22:24 "IW_IP_PHONE"   'Stas    '
          9.1.1.151       <BOUND>    since 01/01/2003 05:00:34
           <FREE RANGE>   9.1.1.152       - 9.1.1.169        =18
           <RESERV> ID:01:00:04:35:00:22:23 "IW_IP_PHONE"   'Andrew  '
          9.1.1.170       <BOUND>    since 01/01/2003 05:49:35
           <FREE RANGE>   9.1.1.171       - 9.1.1.199        =29
           <RESERV> ID:01:00:05:90:02:1F:C8 ""              '        '
          9.1.1.200       <BOUND>    since 01/01/2003 06:22:30
           <OPTION>       Router  9.1.1.3
           <OPTION>       H323_GK_ADDRESS 195.38.45.84

          OK
```

Недостатком такого метода «классификации» клиента является то, что нам пришлось зафиксировать выделяемый этому клиенту IP адрес из пула PHONES. Этот недостаток можно устранить образованием *резервации интерфейса*, о которых будет сказано позже.

Если пул больше не нуждается в резервации, то ее можно удалить командой:

Синтаксис:

dhcpd scope <SCOPE_NAME> delete reservation <CLIENT_ID>

Если к моменту удаления резервации клиент уже получил IP адрес этой резервации, то, после удаления, сервер оставит этот адрес в аренде у данного клиента, если тот не нарушает других правил пула (исключений и фильтра классов). В противном случае сервер аннулирует аренду.

Например:

```
#1> dhcpd scope phones delete
              reservation ID:01:00:05:90:02:1F:C8
          OK
          #1> dhcpd show scope *
          >SCOPES:
          (MSOFT)         192.168.177.20  - 192.168.177.22  [eth0]
          ATTACHED [eth0] <192.168.177.12>/255.255.255.0
           <CLIENT CLASS IDs>: "IW_BRI_GATEWAY" "MSFT 5.0"
           <CLIENT> ID:01:00:C0:DF:10:AF:69 "MSFT 5.0"      'wad     '
          192.168.177.20  <BOUND>    since 01/01/2003 01:01:08
           <FREE RANGE>   192.168.177.21  - 192.168.177.22   =2

          (NEW)           10.12.12.30     - 10.12.12.50     [*]

          (OTHER)         192.168.177.10  - 192.168.177.19  [eth0]
          ATTACHED [eth0] <192.168.177.12>/255.255.255.0
```

```
          <CLIENT> ID:01:00:05:90:02:1F:C8 ""              '        '
         192.168.177.10  <BOUND>    since 01/01/2003 01:16:36
          <FREE RANGE>   192.168.177.11  - 192.168.177.11   =1
          <FREE RANGE>   192.168.177.13  - 192.168.177.19   =7

         (PHONES)        9.1.1.151       - 9.1.1.200       [*] ATTACHED
         [eth0] <9.1.1.100>/255.255.255.0
          <CLIENT CLASS IDs>: "IW_IP_PHONE"
          <CLIENT> ID:01:00:04:35:00:22:24 "IW_IP_PHONE"   'Stas    '
         9.1.1.151       <BOUND>    since 01/01/2003 01:01:47
          <FREE RANGE>   9.1.1.152       - 9.1.1.169        =18

 <RESERV> ID:01:00:04:35:00:22:23 "IW_IP_PHONE"   'Andrew  ' 9.1.1.170
         <BOUND>    since 01/01/2003 01:01:37
          <FREE RANGE>   9.1.1.171       - 9.1.1.200        =30
          <OPTION>       Router  9.1.1.3
          <OPTION>       H323_GK_ADDRESS 195.38.45.84

         OK
```

В этом случае, после удаления резервации, сервер аннулировал аренду клиентом **ID:01:00:05:90:02:1F:C8** IP адреса 9.1.1.200 в пуле PHONES, т.к. класс клиента не удовлетворяет фильтру классов пула PHONES. Спустя некоторое время этот клиент получил в аренду другой адрес из пула OTHER.

```
#1> dhcpd scope phones delete
                reservation ID:01:00:04:35:00:22:23
         OK
         #1> dhcpd show scope phones
         >SCOPES:
         (PHONES)        9.1.1.151       - 9.1.1.200       [*] ATTACHED
         [eth0] <9.1.1.100>/255.255.255.0
          <CLIENT CLASS IDs>: "IW_IP_PHONE"
          <CLIENT> ID:01:00:04:35:00:22:24 "IW_IP_PHONE"   'Stas    '
         9.1.1.151       <BOUND>    since 01/01/2003 01:01:47
          <FREE RANGE>   9.1.1.152       - 9.1.1.169        =18
          <CLIENT> ID:01:00:04:35:00:22:23 "IW_IP_PHONE"   'Andrew  '
         9.1.1.170       <BOUND>    since 01/01/2003 01:01:37
          <FREE RANGE>   9.1.1.171       - 9.1.1.200        =30
          <OPTION>       Router  9.1.1.3
          <OPTION>       H323_GK_ADDRESS 195.38.45.84

         OK
```

А для клиента **ID:01:00:04:35:00:22:23** удаление связанной с ним резервации не привело к аннулированию его аренды адреса **9.1.1.170** , т.к. этот клиент удовлетворяет всем правилам пула.

Внимание! При выполнении команды dhcpd scope <SCOPE_NAME> set range <START_IP> <END_IP>, те ранее созданные в пуле резервации, которые перестают удовлетворять устанавливаемому диапазону, будут удалены из пула автоматически!

Конфигурационные параметры (options)

Конфигурационные параметры (options) – это параметры, которые клиенты могут запрашивать у сервера для более точной настройки хоста. К таким параметрам относятся, например, *Address Time* (время аренды), *Router* (шлюзы по умолчанию), *NTP Servers* и т.п. Клиенты могут запрашивать различный набор этих параметров. Значения этих параметров сервер передает клиентам только, если клиент включил этот параметр в список запрашиваемых и только если серверу известно значение этого параметра. Набор и значения параметров (опций) определяется при конфигурировании DHCP сервера. Наборы (подразделения) опций могут быть определены для:

1. Резервации пула. Значения опций из этого подразделения будут передаваться клиенту этой резервации.
2. Резервации интерфейса. Передаются, если значение запрошенной опции отсутствовало в подразделениях резервации пула.
3. Пула. Значения опций из этого подразделения будут передаваться клиенту получившему аренду из этого пула, если запрашиваемая клиентом опция отсутствует в подразделении резервации пула или интерфейса для этого клиента или клиент не является резервацией.

Copyright © 2004-2005 by InfiNet Wireless

4. Интерфейса. Передаются клиентам, получившим аренду в одном из пулов, подключившихся к данному интерфейсу, а значения запрашиваемой опции отсутствовало в резервации пула, в самом пуле и в резервации интерфейса.
5. Всего сервера. Передаются клиентам, получившем аренду в одном из пулов сервера, если значение запрашиваемой им опции отсутствовало во всех выше перечисленных подразделениях. По сути, это значения опций по умолчанию.

Если значение какой-либо из запрошенных опций отсутствует во всех подразделения, то клиенту не передается никакого значения этой опции. Исключение составляют две опции:

- *Address Time* – значение передается всегда (без запроса клиента). Если значение этой опции не определено ни в одном из подразделении, то клиенту передается значение 120 (аренда адреса = 2 минуты)
- *Subnet Mask* – значение передается всегда (без запроса клиента). Значение этой опции определяется сервером автоматически и ее запрещено определять в подразделениях опций при конфигурации. Значение маски подсети для клиента всегда передается равное маске подсети интерфейса, к которой подключился пул, аренду IP адреса из которого получил клиент.

Сводную таблицу конфигурационных опций DHCP протокола можно получить по адресу http://www.iana.org/assignments/bootp-dhcp-parameters

Для определения набора опций, у DHCP сервера существует ряд команд для каждого подразделения. У этих команд есть параметры, формат ввода которых одинаков для всех этих команд:

OPTION_NAME - название опции (см. сводную таблицу). При указании названия опции, если в названии присутствуют пробелы, то их следует при вводе заменить на знак «подчеркивание» (_). Название опции безразлично к регистру символов.

OPTION_VALUE – значение опции. Формат ввода значения опции зависит от ее предназначения и разделяется DHCP сервером на три категории (формата):

1. Символьная. Строка символов, как, например, для опции *Bootfile-Name.* Если значение этой опции содержит пробелы, то при вводе значение опции следует заключить в кавычки.
2. Двоичная. Одно или несколько десятичных чисел. Если по смыслу опции необходимо задать несколько чисел, то при вводе их необходимо разделить запятыми. Пример таких опций: *Address Time, Time Offset.*
3. IP адрес. Одно или несколько значений, задаваемых в виде IP адреса. Примером такой опции может служить *Router* (шлюз по умолчанию). Если значение такой опции должно состоять из нескольких IP адресов, то при вводе следует разделить их запятыми.

Команда определения (добавления) опций для:

1. Подразделения резерваций пулов

Синтаксис:

dhcpd scope <SCOPE_NAME> reservation
<CLIENT_ID> option <OPTION_NAME> <OPTION_VALUE>

где

- *SCOPE_NAME* – это имя пула для резервации которого необходимо определить значение опции.

Copyright © 2004-2005 by InfiNet Wireless

- ***CLIENT_ID –*** идентификатор клиента резервации. Если опция с таким именем ранее была определена в данном наборе, то команда заменит ее значение на указанное в этой команде.

2. Подразделения резерваций интерфейсов

Синтаксис:

```
dhcpd interface <INTERFACE> reservation
                <CLIENT_ID> option <OPTION_NAME> <OPTION_VALUE>
```

где

- ***INTERFACE –*** имя интерфейса на котором задана резервация для клиента ***CLIENT_ID.*** Если у данного интерфейса не существовало резервации для указанного клиента, то команда автоматически создаст эту резервацию и добавит в набор ее опций указанную в команде.

Резервации интерфейсов предназначены для определения специфического набора конфигурационных параметров для выбранного клиента, вне зависимости от того, в каком из пулов данного интерфейса этот клиент получит аренду. Резервация интерфейса отличается от резервации пула двумя основными свойствами:

- *Не определяет фиксированное значение IP адреса для клиента*. Т.о. предоставляет серверу определить динамически из какого пула и какой IP адрес выдать этому клиенту в аренду.
- *Позволяет изменить класс клиента.* Т.е., если определить для резервации интерфейса опцию *Class Id,* то при поступлении запросов от клиента этой резервации, его класс будет изменен на значение этой опции. Это бывает необходимо в случаях, когда ПО DHCP клиента не передает своего класса. Тем не менее, этот клиент может быть отнесен к какому-либо, например, указанному в фильтре классов одного из пулов данного интерфейса.

Создание резервации интерфейса не противоречит созданию резервации пула для того же клиента.

3. Подразделения пулов

Синтаксис:

```
dhcpd scope <SCOPE_NAME>
              option <OPTION_NAME> <OPTION_VALUE>
```

4. Подразделения интерфейсов

Синтаксис:

```
dhcpd interface <INTERFACE>
            option <OPTION_NAME> <OPTION_VALUE>
```

5. Подразделение сервера

Синтаксис:

```
dhcpd option <OPTION_NAME> <OPTION_VALUE>
```

Соответственно, существует набор команд для удаления выбранных опций из подразделений :

Copyright © 2004-2005 by InfiNet Wireless

Синтаксис:

dhcpd scope <SCOPE_NAME>
reservation <CLIENT_ID> delete option <OPTION_NAME>

dhcpd scope <SCOPE_NAME> delete option <OPTION_NAME>

dhcpd interface <INTERFACE>
reservation <CLIENT_ID> delete option <OPTION_NAME>

dhcpd interface <INTERFACE> delete option <OPTION_NAME>

dhcpd delete option <OPTION_NAME>

Особо следует обратить внимание на удаление опций из подразделения резерваций интерфейса. Если после удаления окажется, что набор опций этой резервации пуст, то сама резервация интерфейса будет уничтожена автоматически.

Не любые опции могут быть заданы в любом подразделении. Кроме *Subnet Mask*, о которой говорилось ранее в допустимом наборе есть опции, которые можно задавать только для определенных подразделении. Для указанного набора данная опция неприменима, сервер сообщит о недопустимости применения данной опции для данного подразделения. Например, опция *Class Id:*

Пример:

#1> dhcpd scope phones option class_id "TestClass"

ERR: This option cannot contain in the given division.

Кроме того, существует ряд служебных опций, которые включены в сводную таблицу, но на самом деле не являются конфигурационными параметрами, а выполняют служебную роль. Список служебных опций DHCP сервера:

- `Subnet Mask`
- `Address Request`
- `Overload`
- `DHCP Msg Type`
- `DHCP Server Id`
- `Parameter List`
- `DHCP Message`
- `DHCP Max Msg Size`
- `Client Id`

При попытке указать (добавить) одну из этих опций для любого подразделения сервер сообщит об ошибке **ERR: This option cannot contain in the given division**.

DHCP сервер поддерживает опции **Classless Static Route** и **MS Classless Static Route**

Предназначены для описания статических бесклассовых маршрутов на клиентах. Формат опций одинаковый:
IP/MASK>GATEWAY[,IP/MASK>GATEWAY,…]
Пример:

dhcpd scope TEST option Classless_Static_Route 192.168.12.0/24>192.169.1.1,192.168.15.0/24>192.169.1.2

Описывает два статических маршрута на подсети 192.168.12.0/24 через gateway 192.169.1.1 и 192.168.15.0/24 через gateway 192.169.1.2

Для контроля опций, запрошенных клиентом и выданных ему сервером существует команда:

Copyright © 2004-2005 by InfiNet Wireless

Синтаксис:

dhcpd interface <INTERFACE|> show client <CLIENT_ID|*>*

где

- ***INTERFACE –*** имя сетевого интерфейса, информацию о клиентах которого необходимо распечатать
- ***CLIENT_ID*** – идентификатор клиента, информацию о котором необходимо распечатать. Вместо имени интерфейса можно указать *, и тогда будет распечатана информация о клиентах всех интерфейсов сервера. Вместо идентификатора клиента так же допускается указывать *, что будет означать запрос информации о всех клиентах. Информация о запрашиваемых клиентом опциях распечатывается только для клиентов, получивших к текущему моменту времени аренду IP адреса у одного из пулов, подключившихся к данному интерфейсу.

Например:

```
#2> dhcpd interface * show client *
          >INTERFACES CLIENTS
          --------- [eth0] ---------
          (IPHONES) <CLIENT> ID:01:00:04:35:00:22:24 "IW_IP_PHONE" 'Unknown node'
          192.168.0.101   <BOUND>    since 25/04/2005 11:32:57
          SUPPLIED OPTIONS:
          #1       .  .  .  .  . DF Subnet Mask              255.255.255.0
          #2       .  .  .  .  .  . Time Offset              <not supplied>
          #3       .  .  S  .  .  . Router                   192.168.0.1
          #7       .  .  .  .  .  . Log Server               <not supplied>
          #42      .  .  S  .  .  . NTP Servers              192.168.0.1
          #230     .  .  S  .  .  . H323 GK ADDRESS          192.168.0.1
          #231     . IR  .  .  .  . H323 LOGIN ALIAS         IWPhone/V. Pupkin/101
          #232     .  .  .  .  .  . H323 GK ID               <not supplied>
```

Здесь список запрошенных клиентом опций (supplied options) состоит из записей (строк), каждая из которых содержит номер (#<N>) запрашиваемой клиентом опции, карты подразделений сервера из которого эта опция была предоставлена клиенту (если была предоставлена), названия опции и ее значения, если эта опция была предоставлена клиенту. Если запрашиваемая клиентом опция не определена ни в одном из подразделении сервера, то в списке она показывается со значением `<not supplied>`. На карте опций подразделения условно обозначаются символами:

1. SR – подразделение резервации пула
2. IR – подразделение резервации интерфейса
3. S – подразделение пула.
4. I – подразделение интерфейса клиента
5. SV – подразделение сервера.

Кроме того, опции, которые были запрошены клиентом и выданы сервером в обязательном порядке, но не определены ни в одном из подразделении (например, *Subnet Mask)* обозначаются DF.

Время аренды (Address Time)

Любая аренда IP адреса выдается клиентам на время, определенное опцией *Address Time.* Если клиент, которому данная аренда была предоставлена, не подтвердит продление аренды за время *Address Time*, сервер аннулирует данную аренду. Значение этого времени может быть установлено и самим клиентом, но не более максимального значения. За максимальное значение этого времени принимается то, которое установлено в опции *Address Time* одного из подразделений, к которому относится данный клиент. Если у сервера не определена такая опция ни в одном из подразделений, то

значение максимального времени принимается равным 120 секундам. В случае, если клиент не запрашивает какого-либо значения этого параметра, то сервер выдает аренду на максимальное время, определяемое так, как было описано выше.

Клиент, получивший аренду, периодически подтверждает ее. Периодичность обычно равна половине *Address Time.* В ответ на подтверждение продления аренды клиентом, сервер подтверждает для клиента конфигурационные параметры аренды (опции). Т.о. если в течение срока аренды, в конфигурации сервера поменялись состав или значения опций подразделения, к которому относится данный клиент, то клиент об этом «узнает» только в моменты подтверждений продления аренды.

Если по истечении времени аренды клиент ни разу не подтвердил ее, пул аннулирует эту аренду. Если этот клиент не был клиентом резервации пула, то пул, отметит IP адрес этой аренды как «условно свободный». На распечатках состояний пулов (dhcpd show scope *) это состояние обозначается как <OBIND>. Т.о., при наличии свободных адресов, пул не будет предоставлять <OBIND> адреса для новых клиентов. Если в течение одних суток с момента аннулирования аренды, данный клиент вновь запросит аренду, то сервер предоставит ему тот же самый IP адрес, что и был у него во время не продленной им аренды.

```
#1> dhcpd show scope MSOFT
        >SCOPES:
        (MSOFT)      192.168.177.20  - 192.168.177.22  [eth0] ATTACHED
        [eth0] <192.168.177.12>/255.255.255.0
         <CLIENT CLASS IDs>: "IW_BRI_GATEWAY" "MSFT 5.0"
         <CLIENT> ID:01:00:C0:DF:10:AF:69 "MSFT 5.0"    'wad     '
        192.168.177.20  <BOUND>    since 01/01/2003 01:01:14
         <O_BIND> ID:01:00:0F:EA:05:29:C6 "MSFT 5.0"
        'win2k3sbs' 192.168.177.21  <OBIND>
         <FREE RANGE>   192.168.177.22  - 192.168.177.22   =1
        OK
```

Одновременно пул записывает параметры аннулированной аренды в специальную базу данных своего интерфейса (boundhistory).

```
#1> dhcpd interface eth0 show boundhistory

[eth0]

>BOUND_HISTORY 1

(MSOFT) ID:01:00:0F:EA:05:29:C6 BOUND=192.168.177.21    until
        02/01/2003 13:25:37

OK
```

Информация об аннулированных арендах хранится в этой базе данных 24 часа и распечатывается в порядке времени хранения записей в ней. По истечении 24 часов запись автоматически уничтожается, а связанный с ним адрес из соответствующего пула переводится в состояние обычного свободного адреса.

Сервер будет повторно использовать условно свободные адреса для других клиентов, если у всех пулов, которые подходят этим клиентам, закончились свободные адреса. При этом сервер в первую очередь будет использовать самые «старые» записи в boundhistory интерфейса клиента.

Сервер также аннулирует аренду клиента по запросу самим клиентом завершения им использования данного адреса (например, штатная парковка компьютера).

Copyright © 2004-2005 by InfiNet Wireless

Проверка допустимости выдачи IP адреса в аренду

Проверка производится в целях предотвращения конфликта IP адресов. После того, как сервер определил IP адрес свободный для передачи в аренду клиенту, но до собственно передачи ее клиенту, сервер обязательно проверяет допустимость предоставления данного IP адреса этому клиенту. Другими словами, сервер убеждается, что этот IP адрес не занят в данный момент ни одним хостом (кроме, может быть самим целевым клиентом) на интерфейсе целевого клиента. Для этого сервер проводит ARP запросы в интерфейсе клиента. Если на запрос никто не ответил или ответил только сам же клиент, то аренда этого IP адреса предоставляется.

Эта проверка производится в любом случае, и для резерваций пулов то же. Исключение составляют только виртуальные интерфейсы, где обязанность такой проверки возложена на самих клиентов.

Если обнаруживается конфликт IP адресов, то данный IP адрес не будет предоставлен в аренду и сервер попытается выделить следующий свободный из адресов подходящего пула. Если в конце концов бесконфликтных свободных адресов не остается, сервер задействует *boundhistory* интерфейса клиента. Если же высвободить IP адрес и там не удалось, сервер заносит клиента в базу данных «не выданных аренд».

Невыданные аренды (unleases)

Клиенты, которым DHCP серверу не удалось выдать IP адрес в аренду, записываются в специальный список – *unleases*. Записи в этом списке хранятся в течение 15 минут, если клиент не повторяет попыток получить аренду. Каждая запись в этом списке состоит из полей:

1. Имя сетевого интерфейса, через который от клиента был получен запрос на получение аренды (интерфейс клиента).
2. Идентификатор клиента
3. Идентификатор класса клиента
4. Имя компьютера клиента (*host name*)

Просмотр содержимого этого списка можно осуществить при помощи команды:

Синтаксис:

dhcpd show unleases <SUBSTR|>*

где

- ***SUBSTR*** – это подстрока для выборочного просмотра списка *unleases*. При исполнении команды, сервер будет распечатывать только такие записи *unleases*, у которых одно из полей содержит указанную подстроку. При этом учитывается регистр символов. Если вместо подстроки указать *, то сервер распечатает содержимое этого списка полностью.

Например:

```
#1> dhcpd show unleases *
        >UNLEASES 1
        eth0    ID:01:00:C0:DF:10:AF:69 "MSFT 5.0"      wad
        OK
```

Виртуальные интерфейсы (virtual interface)

После старта, DHCP клиенты производят запросы на аренду IP адреса. Эти запросы клиенты рассылают широковещательно. Т.к. клиент в это время еще не имеет выделенного ему IP адреса, сервер также производит начальный диалог с клиентом при помощи широковещательных пакетов (*broadcast*). Как известно, широковещательные пакеты не подлежат маршрутизации, и таким образом, в этот момент времени диалог между

Copyright © 2004-2005 by InfiNet Wireless

DHCP клиентом и DHCP сервером может состоятся только в пределах одной сети (физического сегмента). Таком образом, если DHCP клиент физически подключен к другой сети, то прямого диалога с DHCP сервером состоятся не может. Однако, на маршрутизаторе, логически соединяющем сеть клиента с сетью DHCP сервера может работать специальная программная компонента – DHCP Relay Agent (DRA). DRA производит ретрансляцию DHCP пакетов (включая и широковещательные) от DHCP клиентов DHCP серверу и обратно. При этом обмен DHCP пакетами между DRA и DHCP сервером происходит только адресно (unicast). Из этого следует, что DRA и DHCP сервер должны знать IP адреса друг друга еще на этапе своей конфигурации. Для этой цели в DHCP сервере существуют *виртуальные интерфейсы.* В сущности виртуальный интерфейс в DHCP сервере – это некий физический сетевой интерфейс, но находящийся у DRA. Т.к. набор подсетей этих интерфейсов DHCP серверу неизвестен, то при конфигурации виртуальных интерфейсов необходимо указать эти подсети.

Для создания виртуального интерфейса существует команда:

Синтаксис:

dhcpd add virtual interface <GATEWAY>

где

- **GATEWAY** – это IP адрес DRA, у которого находится соответствующий физический интерфейс. После исполнения такой команды в конфигурации сервера образуется еще один интерфейс с именем, образованным из значение IP адреса DRA, следующим образом: v. ***GATEWAY.*** Например: v.192.168.177.81

Пример:

```
#1> dhcpd add virtual interface 192.168.177.81
        [v.192.168.177.81]
          Virtual interface v.192.168.177.81 added
        OK

#1> dhcpd show interface *
        >INTERFACES
        [eth0] UP
         <SUBNET> 9.1.1.100/255.255.255.0
              <SCOPE> (PHONES) 9.1.1.151 - 9.1.1.200
         <SUBNET> 192.168.177.12/255.255.255.0
              <SCOPE> (MSOFT) 192.168.177.20 - 192.168.177.50
         <SUBNET> 192.168.15.55/255.255.255.0
         <RESERVATION>  for ID:01:00:05:90:02:1F:C8
              <OPTION>      Class_Id      "Swissvoice"
        [vlan0] DOWN
         <SUBNET> 192.168.178.1/255.255.255.0
        >VIRTUAL INTERFACES
        [v.192.168.177.81] UP
        OK
```

Таким образом, в конфигурации сервера появился еще один интерфейс с именем v.192.168.177.81. Правила работы с ним точно такие же, как и с обычными сетевыми интерфейсами. Однако, прежде чем к этому интерфейсу сможет подключится какой-либо пул, ему необходимо сконфигурировать набор подсетей. Для этого существует команда:

Синтаксис:

```
dhcpd virtual interface <GATEWAY> add subnet <IP_ADDRESS>
        <SUBNET_MASK>
```

где

Copyright © 2004-2005 by InfiNet Wireless

- ***GATEWAY*** – IP адрес DRA, соответствующего данному виртуальному интерфейсу
- ***IP_ADDRESS*** – IP адрес, который имеет DRA на данной подсети.
- ***SUBNET_MASK*** – маска этой подсети.

Например:

```
#1> dhcpd virtual interface 192.168.177.81
        add subnet 192.168.188.1 255.255.255.0
    [v.192.168.177.81]  Subnet added
            192.168.188.1/255.255.255.0
    OK

#1> dhcpd show interface *
    >INTERFACES
    [eth0] UP
     <SUBNET> 9.1.1.100/255.255.255.0
        <SCOPE> (PHONES) 9.1.1.151 - 9.1.1.200
     <SUBNET> 192.168.177.12/255.255.255.0
        <SCOPE> (MSOFT) 192.168.177.20 - 192.168.177.50
     <SUBNET> 192.168.15.55/255.255.255.0
     <RESERVATION>  for ID:01:00:05:90:02:1F:C8
        <OPTION>    Class_Id    "Swissvoice"
    [vlan0] DOWN
     <SUBNET> 192.168.178.1/255.255.255.0
    >VIRTUAL INTERFACES
    [v.192.168.177.81] UP
     <SUBNET> 192.168.188.1/255.255.255.0
    OK
```

Теперь можно создать пул адресов из которых DHCP сервер будет выделять IP адреса в аренду клиентам, прямой доступ к которым имеет данный DRA.

Например:

```
#1> dhcpd add scope VIRTUAL_TEST
        v.192.168.177.81 192.168.188.20 192.168.188.50
    [v.192.168.177.81] <192.168.188.1> (VIRTUAL_TEST):
       192.168.188.20-192.168.188.50   Scope attached
    OK

#1> dhcpd show interface v.192.168.177.81
    >VIRTUAL INTERFACES
    [v.192.168.177.81] UP
     <SUBNET> 192.168.188.1/255.255.255.0
        <SCOPE> (VIRTUAL_TEST)
          192.168.188.20 - 192.168.188.50
    OK

#1> dhcpd show scope virtual_test
    >SCOPES:
    (VIRTUAL_TEST)  192.168.188.20  - 192.168.188.50
    [v.192.168.177.81] ATTACHED [v.192.168.177.81]
    <192.168.188.1>/255.255.255.0
     <FREE RANGE>   192.168.188.20  - 192.168.188.50   =31

    OK
```

Удалить подсеть из набора виртуального интерфейса можно командой:

Синтаксис:

```
dhcpd virtual interface <GATEWAY>
        delete subnet <IP_ADDRESS> <SUBNET_MASK>
```

Copyright © 2004-2005 by InfiNet Wireless

Например:

*#1> dhcpd virtual interface **192.168.177.81***
*delete subnet **192.168.188.1 255.255.255.0***
[v.192.168.177.81] <192.168.188.1> (VIRTUAL_TEST):
*192.168.188.20-192.168.188.50 **Scope detached***
[v.192.168.177.81] Subnet dropped
192.168.188.1/255.255.255.0
OK

#1> dhcpd show scope virtual_test
>SCOPES:
(VIRTUAL_TEST) 192.168.188.20 - 192.168.188.50
[v.192.168.177.81]

OK

Естественно, раз мы удалили подсеть, к которой был подключен пул, то этот пул немедленно отключился и, т.к. разрешенный интерфейс у него v.192.168.177.81, то этот пул будет оставаться неподключенным до тех пор, пока не будет сконфигурирована подходящая для него подсеть у виртуального интерфейса v.192.168.177.81.

Очистка конфигурации DHCP сервера.

При необходимости очистить конфигурацию сервера, его надо остановить командой *dhcp stop.* После остановки, полностью очистить конфигурацию можно командой

dhcpd clear

16.DHCP relay. Команда dhcpr

Общее описание

Для работы протокола DHCP-сервер должен находиться в том же сетевом сегменте, что и пользователи сети. Между ними не должно быть маршрутизаторов. Если сеть состоит из нескольких сегментов, разделенных маршрутизаторами - необходим DHCP-сервер в каждом сегменте, поскольку начальный широковещательный запрос клиента через маршрутизаторы не проходит. Возможная альтернатива такому подходу - установка в каждом "бессерверном" сегменте DHCP-ретранслятора (Relay Agent) для переадресации запросов от пользователей сегмента на основной DHCP-сервер. В такой роли могут выступать и некоторые маршрутизаторы.

Полный синтаксис:

dhcpr [add]|delete SERVERIP

dhcpr (flush|trace|notrace)

dhcpr (lock|unlock) INTERFACE

dhcpr (info|noinfo)

dhcpr (start|stop)

Copyright © 2004-2005 by InfiNet Wireless

Запуск/остановка ретранслятора

Синтаксис:

dhcpr {start | stop}

Команда запускает или останавливает DHCP ретранслятор.

Пример:

dhcpr start

Список DHCP серверов

Синтаксис:

dhcpr [add]|delete SERVERIP

Команда служит для изменения списка DHCP серверов для которых будет производиться ретрансляция запросов клиентов.

Пример:

dhcpr add 125.12.100.12

dhcpr 125.12.100.13

dhcpr delete 125.12.100.12

Блокировка интерфейсов

По умолчанию DHCP ретранслятор принимает запросы от клиентов со всех сетевых интерфейсов типа ethernet. Если необходимо запретить прием запросов от клиентов на каком-либо сетевом интерфейсе, то необходимо заблокировать данный интерфейс.

Синтаксис:

dhcpr (lock|unlock) INTERFACE

INTERFACE – здесь имя одного или более (разделенных пробелом) сетевого интерфейса, прием запросов с которых необходимо заблокировать.

Пример:

dhcr lock eth0

Использование DHCP опции “DHCP Relay agent information”

Для идентификации интерфейса клиента при приеме ответов сервера, ретранслятор может использовать специальную DHCP опцию, которую он добавляет к пакету запроса клиента при ретрансляции. Не все DHCP серверы поддерживают эту возможность. Поэтому, по умолчанию DHCP ретранслятор не пользуется этой возможностью. Разрешить/запретить использование такой опции можно командой:

Copyright © 2004-2005 by InfiNet Wireless

Синтаксис:

dhcpr (info|noinfo)

Пример:

dhcpr info

17.DHCP клиент. Команда dhcpc

Общее описание

DHCP клиент предназначен для автоматического получения от DHCP сервера параметров для одного или нескольких сетевых интерфейсов устройства. К параметрам относятся IP – адрес, маска сети, шлюз по умолчанию и проч.

Управление работой DHCP клиента осуществляется командой:

Полный синтаксис:

dhcpc [options] [IFNAME] [commands]

IFNAME – имя сетевого интерфейса к которому относятся опции и команды.

Опции

Опции определяют параметры работы DHCP клиента на данном интерфейсе, либо значение этих параметров по умолчанию если имя интерфейса не указано. Для любой опции можно указать специальные значения: *none* и *default*. Значение опции *none* означает отсутствие данного параметра для данного интерфейса не взирая на наличие значения по умолчанию данного параметра. Значение опции *default* означает отсутствие специального значения данного параметра. При этом, к данному интерфейсу применяется значение по умолчанию, если оно задано. Отметим, что значение опции *default* не отображается в конфигурации DHCP клиента.

- *-l (none|default|$ACLNAME|acl:ACLNAME)* – устанавливает список IP адресов DHCP серверов, от которых клиенту разрешено принимать параметры. Здесь ACLNAME – имя системного списка доступа (см. команду *acl*). Если указанный список доступа отсутствует в системе (не создан командой *acl*), то DHCP клиент допускает получение параметров от любого DHCP сервера.
- *-k (none|default|key:KEYVALUE)* - устанавливает ключ авторизации. Авторизация DHCP проводится в соответствии с *"RFC 3118 - Authentication for DHCP Messages"*.
- *-a (none|default|NUMBER)* – устанавливает количество повторных *arp* запросов, которые проводит DHCP клиент после получения предложения IP адреса от DHCP сервера. В соответствии с DHCP, клиент обязан проверить предложенный IP адрес на предмет отсутствия в сети устройств с таким адресом. Для надежности DHCP клиент проводит несколько *arp* запросов с интервалом ¼ секунды. Если количество *arp* запросов не определено ни у заданного интерфейса, ни по умолчанию, то DHCP клиент проводит 16 запросов.
- *-t (on|off)* - опция не устанавливает значения какого-либо параметра. Она включает или выключает вывод отладочной информации в

Copyright © 2004-2005 by InfiNet Wireless

системный журнал. Эта опция действует DHCP клиент безотносительно к интерфейсам.

Команды

- *start* - запускает DHCP клиент на указанном интерфейсе
- *stop* - останавливает DHCP клиент на указанном интерфейсе
- *delete* - останавливает DHCP клиент на указанном интерфейсе и очищает все опции.
- *dump* - отображает текущее состояние DHCP клиента.

Примеры

```
dhcpc –a 5
dhcpc –l $DHCP_SERVERS eth0 start
dhcpc –a none –k key:qwerty  rf4.0 start
```

Данная конфигурация устанавливает значения количества ARP запросов по умолчанию равным 5. Для интерфейса eth0 устанавливается список допустимых серверов DHCP_SERVERS. Производится запуск DHCP процесса для eth0. Для интерфейса rf4.0 отменяется значение количества ARP запросов. Т.о. для интерфейса rf4.0 DHCP клиент будет производить 16 arp запросов. В то же время DHCP процесс на интерфейсе rf4.0 будет использовать авторизацию с ключом *qwerty.*

```
dhcpc dump
```

Команда распечатывает текущее состояние DHCP клиента

```
ID  I-face   IP address/mask   Gateway address    Server ID    Lease exp.
== ======== ================== ===============
 0  eth0   192.168.61.29/26    192.168.61.1    192.168.61.1 000:35:16
 1  rf4.0  ------------------ ---------------
```

На распечатке видно, что DHCP процессы запущены для интерфейсов eth0 и rf4.0.

Для интерфейса eth0 DHCP клиент получил IP адрес 192.168.61.26 с сетевой маской длиной 26 от DHCP сервера 192.168.61.1. До истечения срока аренды этих параметров осталось 35 минут и 16 секунд.

DHCP процессу на интерфейсе rf4.0 получить сетевые параметры пока не удалось.

Copyright © 2004-2005 by InfiNet Wireless

18.VRRP сервер. Команда vrrp

Общее описание

В настоящее время большинство крупных ЛВС строятся по схеме с маршрутизаторами в центре. В таких ЛВС обычно организуются виртуальные сети с маршрутизацией, предусмотрены избыточные связи между устройствами, установлен резервный центральный маршрутизатор. При кажущейся абсолютной надежности подобной структуры ее слабым звеном является именно центральный маршрутизатор. В случае выхода его из строя возможно несколько вариантов развития событий, требующих вмешательства системного администратора: настройка рабочих станций на работу с другим маршрутизатором в качестве шлюза, изменение конфигурации дублирующего сетевого устройства, установка дополнительного маршрутизатора или что-нибудь еще.

Сервер VRRP предназначен для ускорения процедуры восстановления работоспособности сети именно в описываемом случае. Фактически он обеспечивает процесс перехода "обязанностей" от одного устройства к другому при отказе первого. При использовании сервера VRRP резервный маршрутизатор включается в работу автоматически, и все заботы администратора сводятся лишь к ремонту вышедшего из строя устройства.

Каждый маршрутизатор, предназначенный для резервирования, должен войти в виртуальный маршрутизатор (VR). В пределах VR существует список IP адресов этого VR. В тот момент, когда какой-либо из маршрутизаторов берет на себя функции главного (в случае отказа последнего), он начинает «сопровождать» каждый из IP адресов из этого списка (т.е. отвечать ARP запросами и, если маршрутизатор имеет достаточный приоритет, брать на себя функции хоста с этими IP адресами).

VR идентифицируется своим идентификатором – числом в диапазоне 1…255 (VRID).

Таким образом, логика работы VRRP сервера заключается в следующем:

1) На нескольких VRRP маршрутизаторах создаются VR, каждый из которых имеет равный VRID и идентичный список IP-адресов;

2) из VRRP маршрутизаторов выбирается главный (режим MASTER) все остальные становятся подчиненными маршрутизаторами (режим BACKUP); Маршрутизатор, выполняющий функции главного периодически рассылает специальные пакеты в пределах (производит перекличку). По наличию этих пакетов BACKUP маршрутизаторы судят о работоспособности MASTER.

3) при выходе из строя главного маршрутизатора (длительное время нет пакетов переклички от MASTER) один из подчиненных становится главным и начинает обрабатывать пакеты, адресованные виртуальному маршрутизатору.

Выбор главного виртуального маршрутизатора производится автоматически: им становится маршрутизатор с наибольшим приоритетом или (при равенстве) с большим primary IP-адресом сетевого интерфейса.

Полный синтаксис:

vrrp {start|stop|dump [IFNAME:VRID]}

Copyright © 2004-2005 by InfiNet Wireless

vrrp IFNAME:VRID [start|stop|clean|flush]

vrrp IFNAME:VRID [add]|delete IPADDRESS[/(MASK|MASKLEN)] …

vrrp IFNAME:VRID [-(password|key)=[PASSWORD]]

vrrp IFNAME:VRID [-priority=[PRIO|own]] [-interval=AINT]

vrrp IFNAME:VRID [-preempt=(on|off)] [-owner=[on|off]] [-learn=(on|off)]

Описание команд

Запуск/остановка сервера

Синтаксис:

vrrp {start | stop}

Команда запускает или останавливает VRRP сервер.

Пример:

vrrp start

Создание виртуального маршрутизатора

Синтаксис:

vrrp IFNAME:VRID add IPADDRESS[/{MASK|MASKLEN}] …

Команда создает на интерфейсе с именем IFNAME виртуальный маршрутизатор с идентификатором VRID. VRID – это число в диапазоне 1....255. Кроме того, команда добавляет указанный в параметрах IP адрес(а) в список IP адресов VR. Ни один IP адрес VR не должен совпадать с первичным адресом (primary IP address) интерфейса, на котором он создан. VRRP сервер позволяет создавать несколько виртуальных маршрутизаторов на одном сетевом интерфейсе, но списки их IP адресов не должны пересекаться.

Пример:

vrrp eth0:10 add 9.8.7.6/24

Запуск/остановка виртуального маршрутизатора

Синтаксис:

Vrrp IFNAME:VRID {start|stop}

Команда запускает/останавливает участие данного маршрутизатора в указанном виртуальном.

Пример:

vrrp eth0:10 start

Установка приоритета маршрутизатора

Синтаксис:

Copyright © 2004-2005 by InfiNet Wireless

vrrp IFNAME:VRID -priority=[PRIO|own]

Команда устанавливает приоритет данного маршрутизатора в VR. Значение приоритета имеет диапазон 2…255. Приоритет маршрутизатора имеет значение при выборе главного в случае его отказа. При этом главным маршрутизатором выбирается маршрутизатор с наибольшим приоритетом.

Приоритет 255 имеет особое значение. Оно определяет, что данный маршрутизатор будет главным в пределах заданного VR. Главный маршрутизатор с таким приоритетом всегда является владельцем (owner) всех IP адресов VR.

Пример:

vrrp eth0:10 –priority=200

Режим owner

Синтаксис:

vrrp IFNAME:VRID -owner=on|off

В режиме owner маршрутизатор является владельцем IP адресов VR вне зависимости от своего приоритета. Т.е. даже если в данный момент маршрутизатор не выполняет функции главного, IP адреса VR присутствуют в списках IP адресов сетевого интерфейса, на котором создан данный VR. В то же время эти адреса находятся в «пассивном» состоянии. Т.е. на эти адреса маршрутизатор отвечать не будет до тех пор, пока к нему не перейдут функции главного (например, при отказе главного).

По умолчанию режим owner включен.

Пример:

vrrp eth0:10 –owner=off

Режим преемственности

Синтаксис:

vrrp IFNAME:VRID –preempt=on|off

Если режим преемственности выключен, то, не смотря на свой приоритет, маршрутизатор никогда не возьмет на себя функции главного при наличии (работоспособности) других маршрутизаторов в составе VR.

По умолчанию преемственность включена.

Пример:

vrrp eth0:10 –preempt=off

Установка периода переклички

Синтаксис:

vrrp IFNAME:VRID -interval=AINT

Данная команда позволяет установить необходимый период переклички главного маршрутизатора. Значение параметра задается в секундах.

Copyright © 2004-2005 by InfiNet Wireless

Маршрутизатор, выполняющий в текущий момент времени функции главного (MASTER), периодически сообщает об этом другим маршрутизатором VR. По наличию этих сообщений другие (BACKUP) маршрутизаторы судят о работоспособности главного. Период этих сообщений по умолчанию равен одной секунде.

При установке другого значения этого параметра следует помнить, что у всех маршрутизаторов данного VR этот параметр должен быть одинаковым.

Пример:

vrrp eth0:10 –interval=2

Режим самообучения

Синтаксис:

vrrp IFNAME:VRID –learn=on|off

Данный режим позволяет маршрутизатору накапливать список IP адресов VR в моменты, когда он не выполняет функции главного (в режиме BACKUP). Этот режим служит для упрощения конфигурирования VRRP серверов. Список IP адресов VR достаточно заполнить на одном маршрутизаторе – владельце IP адресов (с приоритетом 255). Остальным маршрутизатором достаточно создать VR с пустым списком IP адресов и в режиме самообучения.

Пример:

vrrp eth0:10 –learn=on

Удаление VR

Синтаксис:

vrrp IFNAME:VRID clean

Команда удаляет заданный VR.

Пример:

vrrp eth0:10 clean

Удаление IP адрес из списка VR

Синтаксис:

vrrp IFNAME:VRID delete IPADDRESS

Команда удаляет заданный IP адрес из списка VR.

Очистка списка IP адресов VR

Синтаксис:

vrrp IFNAME:VRID flush

Команда удаляет все IP адреса из списка VR.

Copyright © 2004-2005 by InfiNet Wireless

Пример:

vrrp eth0:10 flush

Авторизация VRRP

В соответствии с RFC 2338 VRRP сервер поддерживает два режима авторизации:

1. Простая парольная
2. IP Authentication Header

Включение этих схем авторизации производится соответственно двумя командами:

vrrp IFNAME:VRID –password=PASSWORD

vrrp IFNAME:VRID –key=PASSWORD

Как показывает опыт использования, никакая авторизация VRRP не может обеспечить полную безопасность VR. Этот факт был отражен в более позднем RFC 3768:

10. Security Considerations

VRRP does not currently include any type of authentication. Earlier versions of the VRRP specification included several types of authentication ranging from none to strong. Operational experience and further analysis determined that these did not provide any real measure of security. Due to the nature of the VRRP protocol, even if VRRP messages are cryptographically protected, it does not prevent hostile routers from behaving as if they are a VRRP master, creating multiple masters. Authentication of VRRP messages could have

prevented a hostile router from causing all properly functioning routers from going into backup state. However, having multiple masters can cause as much disruption as no routers, which authentication cannot prevent. Also, even if a hostile router could not disrupt VRRP, it can disrupt ARP and create the same effect as having all routers go into backup.

Отображение состояния VRRP сервера

Синтаксис:

vrrp dump

Команда распечатывает на экране текущее состояние сервера VRRP.

Пример:

Copyright © 2004-2005 by InfiNet Wireless

vrrp dump

```
VRRP  interface:ID   Prio  AInterval      Master  IP        STATE          Time
Stop reason

==================  ====  =========  ===============  =======  ===============
===========

          eth0:010 200o    001    192.168.15.50   BACKUP      0/0:0:3:000
```

Состояние VRRP сервера распечатывается в виде таблицы, состоящей из колонок:

- *VRRP interface:ID* – отображает VR в виде IFNAME:VRID
- *Prio* – отображает приоритет, который имеет данный маршрутизатор в данном VR. Если установлен режим owner, то после приоритета выводится маленькая буква **o**.
- *AInterval* – отображает установленный интервал переклички.
- *Master IP* – отображает primary IP адрес интерфейса маршрутизатора, который в данный момент выполняет функции главного (MASTER).
- *STATE* – отображает текущее состояние маршрутизатора:
    - *MASTER*
    - *BACKUP*
    - *STOP*

    Если у данного маршрутизатора включен режим самообучения, то перед состоянием печатается маленькая буква **l** (эль), например, *lBACKUP*
- *Time* – отображает интервал времени, в течение которого маршрутизатор находится в режиме STATE. Интервал представлен в виде ДНИ/ЧАСЫ:МИНУТЫ:СЕКУНДЫ:000
- *Stop reason* – в случае невозможности продолжения работы, маршрутизатор останавливает работу в данном VR. Для данного VR маршрутизатор переходит в состояние STOP. Данная колонка отображает причину отказа функционирования. Возможные причины остановки:
    - *Configuration conflict* – причиной остановки стал конфликт конфигурации VR. Данная ситуация случается, если разные VR на одном и том же интерфейсе имеют пересекающиеся списки IP адресов.
    - *IP Address list is empty* – не задано ни одного IP адреса.
    - *Interface has no primary IP address* – у интерфейса отсутствует или удален primary IP адрес.
    - *Interface is down* – интерфейс, на котором создан VR перешел в выключенное состояние (down)

Copyright © 2004-2005 by InfiNet Wireless

V.Специальные команды

1. Ctl (управление внешними устройствами)

Команды управления внешними устройствами

Синтаксис:

ctl

ctl heater

ctl switch

ctl temperature

ctl signaling

ctl trap

Описание:

Маршрутизатор имеет имеет датчик внутренней температуры и два порта для подключения внешних устройств: порт для подключения исполнительных устройств и порт датчиков сигнализации.

Порт сигнализации поддерживает датчики (реле) с нормально замкнутыми или нормально разомкнутыми контактами.

Порт для подключения исполнительных устройств может работать в двух режимах: с внешним и внутренним источником питания. Напряжение внутреннего источника 5V, максимальный ток нагрузки 200mA. При подключении с внешним источником его напряжение не должно превышать 30V, максимальный ток нагрузки 1A.

Команда **ctl**, выполненная без параметров показывает температуру внутри устройства.

Команда **ctl heater** устанавливает пороговую температуру внутри устройства, ниже которой нагреватель включается. Диапазон задаваемых температур: от -15 C до 0 C. Доступна только для устройств "**OT**":

ctl heater -10

В примере команда задает порог в -10C. Когда температура внутри устройства станет ниже заданной, нагреватель включится.

Команда **ctl switch** управляет портом исполнительных устройств и позволяет включать/выключать исполнительное устройство вручную или автоматически.

ctl switch on | off

- **on** - включить исполнительное устройство.
- **off** - выключить исполнительное устройство.

ctl switch auto signalling | temperature

- **temp**erature - автоматически управляет исполнительным устройством в зависимости от показаний датчика температуры.
- **sign**alling - автоматически управляет исполнительным устройством в зависимости от состояния внешнего датчика.

Команда **ctl temp**erature задает границы температуры и режим для автоматического управления исполнительным устройством.

ctl temperature high значение low значение cooler | heater

Copyright © 2004-2005 by InfiNet Wireless

- параметр **hi**gh задает верхнюю границу температуры, при которой необходимо включить или выключить исполнительное устройство.
- параметр **lo**w задает нижнюю границу температуры, при которой необходимо включить или выключить исполнительное устройство.
- параметр **co**oler говорит о том, что устройство является вентилятором, и автомат коммутации работает по следующей схеме:
  - если температура больше или равна значению **hi**gh, то исполнительное устройство включается
  - если температура меньше значения **lo**w, то исполнительное устройство выключается.
- параметр **he**ater говорит о том, что устройство является нагревателем, и автомат коммутации работает по следующей схеме:
  - если температура меньше значения **lo**w, то исполнительное устройство включается
  - если температура больше или равна значению **hi**gh, то исполнительное устройство выключается.

Команда **ctl sign**alling задает режим работы и тип внешнего датчика.

***ctl sign**alling [**en**able | **di**sable | **ma**niac] [**op**ened | **cl**osed] [**lo**g | **nol**og] [**bt** значение] [**fp** | **nof**p] [**tr**aps количество]*

- **en**able - включить распознавание срабатывания датчика.
- **di**sable - выключить распознавание срабатывания датчика.
- **ma**niac - включить режим сигнализации. В этом режиме, если сработает датчик, события об этом посылаются до тех пор, пока не будет выполнена команда **di**sable.
- **op**ened - нормально разомкнутый контакт.
- **cl**osed - нормально замкнутый контакт.
- **lo**g, **no**log - записывать или нет события от датчика в системный журнал.
- **bt** - время дребезга контактов датчика (по паспорту) в миллисекундах.
- **fp**, **nof**p - включить, выключить распознавание самого первого срабатывания датчика.

  В режиме **fp** по первому срабатыванию датчика отправляется событие, затем отрабатывает алгоритм против дребезга, и если это было ложное срабатывание, то посылается событие, что датчик вернулся в исходное состояние. Режим fp нельзя использовать совместно с **ma**niac.

  Если включен режим **nof**p, то сначала отрабатывается алгоритм против дребезга, а затем посылается событие от датчика.
- **tr**aps - задает количество событий, посылаемых по протоколу SNMP, на каждое срабатывание датчика, если **tr**aps равно 0, то ни один пакет не будет отправлен. Пакеты отправляются с интервалом 1 секунда.
  Событие о срабатывании датчика в формате SNMP имеет oid 1.3.6.1.4.1.3942.0.100
  Событие о возвращении датчика в исходное состояние имеет oid 1.3.6.1.4.1.3942.0.101
  Для того, чтобы посылать события в формате SNMP необходимо запустить сервис trapd командой "**trapd on**" и установить необходимые адреса (см. команду **trapd**).

Copyright © 2004-2005 by InfiNet Wireless

VI.Недокументированные служебные команды

Наличие и формат этих команд может быть изменён в любое время:

- **mem** - статистика системной памяти
- **mb** - состояние внутренних буферов данных
- **ps** - статистика активных процессов

Copyright © 2004-2005 by InfiNet Wireless

VII.Unify SoftSwitch

1. Gatekeeper

Управление параметрами коммутатора voip сигнализации

Команда **gatekeeper** системы WANFlex, позволяет управлять коммутатором сигнализации **voip**, создать начальную конфигурацию, запустить или остановить процесс. Конфигурация самого коммутатора ничем не отличается от **AquaGatekeeper**, (изменение файла конфигурации и мониторинг состояния) происходит с помощью программы **AquaGatekeeperMonitor**.

Синтаксис:

gatekeeper [options] [commands]

Команды:

- **start** - Start gatekeeper service
- **stop** - Stop gatekeeper service
- **create** - Create empty configuration

Опции:

- **--help -h -?** - Help
- **--version -v** - Show file version
- **--configuration=[filename]** - Use configuration file location, default is Gatekeeper.cfg

Пример:

console> gatekeeper start

console> gatekeeper stop

- **start**

  Запустить процесс коммутатора.

- **stop**

  Остановить процесс коммутатора.

- **create**

  С помощью этой команды можно создать конфигурацию во флеше, после выполнения этой команды, вы сможете подключится к коммутатору Монитором для управления.

  Убедитесь что перед выполнением этой команды у вас сконфигурирован IP-адресс на интерфейсе eth0, именно этот адрес будет указан в конфигурации в секции **system** конфигурации коммутатора.

- **configuration**

  С помощью опции **configuration** можно указать имя файла конфигурации, которое будут использовать команды **create** и **start**.

  Пример:

  console> gatekeeper --configuration='test.cfg' create

  console> gatekeeper start

  По умолчанию имя файла Gatekeeper.cfg

Copyright © 2004-2005 by InfiNet Wireless