УТВЕРЖДЕН

RU-НКБГ. 70007-01 90-ЛУ

Dionis-NX 2.0

специальная операционная система

Справочник команд

RU.НКВГ. 70007-01 90

Листов 428

Инв.№подл.	Подп.идата	Взам.инв.№	Инв.№дубл.	Подп.идата	

ОГЛАВЛЕНИЕ

Стр.

1 Общие сведения 9
1.1 Функциональные возможности ... 9
1.2 Дополнительные технические данные ... 15

2 Установка с флеш-диска 17

3 Основы работы с интерфейсом командной строки 19
3.1 Режимы работы с системой ... 20
3.2 Виртуальные консоли ... 21
3.3 Пространства имен ... 21
3.4 Работа с конфигурациями ... 22
3.5 Просмотр и копирование конфигураций ... 23
3.6 Команды режима configure и их влияние на действующую конфигурацию ... 24
3.7 Команды работы с файлами ... 28

4 Выключение и перезагрузка 33

5 Предварительная настройка 35
5.1 Начало работы (Установка и смена паролей) ... 35
5.2 Общие настройки ... 35
5.3 Включение IPv6 ... 36
5.4 Интерфейсы ... 37
5.5 Статическая маршрутизация ... 39
5.6 Клиент DNS ... 41
5.7 Просмотр и настройка ARP-таблицы ... 43
5.8 Просмотр и настройка таблицы соседей IPv6(neighbours) ... 44
5.9 Настройки стека TCP/IP ... 44
5.10 Основы работы со службами (сервисами) ... 48
5.11 Диагностика ... 49

6 Учетные записи 51
6.1 Учетная запись консольного доступа (учетная запись оператора) ... 51
6.2 Учетные записи администратора ... 51
6.3 Управление учетными записями ... 52

7 Ролевая модель 57
7.1 Права доступа учетной записи администратора ... 57
7.2 Полномочия системы ... 57
7.3 Команды управления полномочиями и ролями системы для учетных записей . 64
7.4 Управление ролями ... 65
7.5 Отображение полномочий и зависимости полномочий ... 66

8 Фильтрация и модификация 69
8.1 Создание ip access-list ... 69
8.2 Привязка ip access-list ... 70
8.3 Правила отбора ... 71
8.4 Другие правила списков контроля доступа ... 77
8.5 Модификация ... 77
8.6 IPv6 ... 78
8.7 Группирование IP адресов ... 78

9 Многоадресная передача 79
9.1 Общие сведения о настройке многоадресноймаршрутизации ... 81
9.2 Настройка протокола DVMRP ... 81
9.3 Настройка протокола PIM ... 83
9.4 Настройка протокола IGMP ... 90
9.5 Настройка статической многоадресной маршрутизации ... 92

9.6 Мониторинг работы многоадресной маршрутизации ... 92
9.7 Работа со службой ... 93
9.8 Пример настройки видео-трансляции при помощиVideoLAN Vic ... 93
9.9 Сокращения и термины ... 93

10 NAT 95
10.1 Создание ip nat-list ... 96
10.2 Другие типы NAT ... 97
10.3 Привязка ip nat-list ... 97
10.4 Просмотр и удаление активных соединений ... 98

11 Helper 99
11.1 Создание helper-list ... 99
11.2 Применение helper-list ... 99
11.3 tcpdump ... 99
11.4 Трассировка ... 100
11.5 Протоколирование правил фильтрации ... 104
11.6 Системные журналы ... 104
11.7 Сигнал тревоги ... 105
11.8 Служба watcher ... 107

12 RADIUS - аутентификация и авторизация с помощью AAA протокола 111
12.1 Введение ... 111
12.2 Настройка для подключения к RADIUS серверу ... 111
12.3 Настройка RADIUS-сервера ... 111

13 VLAN 113

14 IEEE 802.lad (QinQ) 115

15 VXLAN 117

16 WIFI-интерфейсы 119
16.1 Введение ... 119
16.2 Работа WIFI-интерфейса в режиме беспроводнойточки доступа ... 119
16.3 Работа WIFI-интерфейса в режиме беспроводного клиента ... 119
16.4 Прочие команды, доступные для работы с WIFI-интерфейсом ... 120

17 MODEM-интерфейсы 121
17.1 Введение ... 121
17.2 Команды доступные для настройки интерфейса ... 121
17.3 Номер порта модема (modem-backend) ... 121
17.4 Пример настрой™ интерфейса ... 121

18 Bonding-интерфейсы 123
18.1 Режимы агрегации ... 123
18.2 Режимы мониторинга ... 125

19 Сетевые мосты 127

20 GRE-туннели 129

21 GRETAP-туннели 131

22 IP6GRE-туннели 133

23 IPeGRETAP-туннели 135

24 VPN-туннели 137
24.1 Введение ... 137
24.2 Импорт, удаление и просмотр доступных ключейи сертификатов ... 137
24.3 VPN-интерфейс ... 138
24.4 SVPN-интерфейс ... 142

25 Экспорт статистики по Netflow 145

26 Служба NTP 147

27 Служба DNS 149
27.1 Контроль доступа ... 150
27.2 Виды ... 151
27.3 Зоны ... 152
27.4 Другие настройки ... 157
27.5 Динамическое обновление зон ... 159
27.6 Ограничения ресурсов службы ... 163
27.7 Журналы ... 164
27.8 Диагностика ... 165
27.9 Работа со службой ... 167

28 Служба DHCP 169
28.1 Общие настройки службы ... 169
28.2 Настройка статического назначения ... 171
28.3 Настройка динамического назначения ... 172
28.4 Сетевые DHCP-опции ... 172
28.5 Пользовательские DHCP-опции ... 173
28.6 Связь с DNS (динамическое обновление) ... 174
28.7 Работа со службой ... 174
28.8 Примеры ... 175

29 Служба DHCP6 177
29.1 Общие настройки службы ... 177
29.2 Настройка статического назначения ... 177
29.3 Настройка динамического назначения ... 178
29.4 Делегирование префиксов ... 179
29.5 Сетевые DHCP-опции ... 179
29.6 Пользовательские DHCP-опции ... 180
29.7 Связь с DNS (динамическое обновление) ... 180
29.8 Работа со службой ... 180
29.9 Примеры ... 180

30 Служба DCHPRELAY 181
30.1 Основные настройки ... 181
30.2 Дополнительные настройки ... 181
30.3 Пример ... 182

31 Служба DCHPRELAY6 183
31.1 Основные настройки ... 183
31.2 Дополнительные настройки ... 183
31.3 Пример ... 184

32 Служба PROXY 185
32.1 Общие понятия ... 186
32.2 Общая настройка службы ... 189
32.3 Настройка параметров кэширования ... 191
32.4 Настройка доступа к службе ... 192
32.5 Настройка фильтрации HTTP-заголовков ... 193
32.6 Настройка выборочного проксирования ... 193
32.7 Настройка выборочного кэширования ... 194
32.8 Настройка аутентификации ... 194
32.9 Настройка контроля пропускной способности сети ... 197
32.10 Правила проверки объектов в кэше на свежесть ... 198
32.11 Настройка адаптации содержимого ... 200
32.12 Прочие настройки службы ... 202
32.13 Настройка журналов службы ... 202
32.14 Работа со службой ... 204

32.15 Мониторинг службы 205

33 Служба SNMP 217
33.1 Общая настройка службы SNMP 217
33.2 Настройка базовой SNMP информации 217
33.3 Настройка правил доступа 217
33.4 Настройка правил нотификаций 218
33.5 Работа со службой 219

34 SSH 221
34.1 Сервер SSH 221
34.2 Клиент SSH 223
34.3 Соединение без использования паролей 224
34.4 Передача файлов 225

35 Telnet 227
35.1 Настройка 227

36 Сервис DIWEB 229

37 Сервис XMPP 231

38 Служба netperf 233
38.1 Настройка службы netperf режима configure 233
38.2 Команда netperf режима enable 233

39 Служба IPERF 235
39.1 Настройки службы iperf режима configure 235
39.2 Команда iperf режима enable 235

40 Служба SLAGENT 237
40.1 Настройка службы slagent из режима configure 237
40.2 Герерация UDP-запросов 237

41 Служба LLDP 239
41.1 Базовые настройки службы и настройки обязательных TLV 239
41.2 Настройка опциональных TLV (DOT1) 241
41.3 Настройка опциональных TLV (DOT3) 241
41.4 Настройка расширения LLDP-MED 243
41.5 Работа со службой 245

42 MAILER - служба пересылкипочтовых сообщений 247
42.1 Введение 247
42.2 Настройка службы пересылки почтовых сообщений 247
42.3 Отправка сообщения или файла с помощью службы MAILER 248
42.4 Настройка service-watcher для отправки сообщений с помощью службы MAILEF249

43 Служба автоконфигурирования IPv6 251
43.1 Секция настройки сервиса RA 251
43.2 Общие настройки интерфейса 252
43.3 Префикс 253
43.4 Маршруты 254
43.5 Клиентские хосты 255
43.6 Рекурсивный DNS-сервер 255
43.7 Список поиска DNS 256

44 L2TP-туннели 257
44.1 Введение 257
44.2 Настройка службы l2tp 258
44.3 Настройка клиентского интерфейса l2tp 266
44.4 Прочая работа 268
44.5 Пример настройки 268

45 PPTP-туннели 271
45.1 Введение ... 271
45.2 Настройка службы pptp ... 271
45.3 Настройка интерфейса pptp ... 272
45.4 Пример настройки ... 273

46 PPPOE-туннели 275
46.1 Введение ... 275
46.2 Настройка службы pppoe ... 275
46.3 Настройка интерфейса pppoe ... 277
46.4 Пример настройки ... 278

47 Механизмы качества обслуживания (QoS) 279
47.1 Классификация ... 279
47.2 Политика обслуживания policy-map ... 280
47.3 Пропускная способность интерфейса ... 282
47.4 Политики обслуживания prio-map и prio-tos-map ... 282
47.5 Политика обслуживания red-map ... 283
47.6 Политика обслуживания codel-map ... 283
47.7 Политика обслуживания fq_codel-map ... 284
47.8 Политика обслуживания sfq-map ... 284
47.9 Политика обслуживания gred-map ... 284
47.10 Ingress ... 284
47.11 Привязка политики к интерфейсу ... 285
47.12 Туннельный трафик ... 285

48 Расширенная статическая маршрутизация 287

49 Динамическая маршрутизация 289
49.1 Списки ... 289
49.2 RIP ... 289
49.3 RIPNG (RIP для IPv6 сети) ... 296
49.4 OSPF ... 297
49.5 OSPF6 (OSPFV3 для IPv6 сети) ... 312
49.6 BGP ... 313
49.7 Карты маршрутов ... 338

50 MPLS - многопротокольная коммутацияпо меткам 345
50.1 Введение ... 345
50.2 Статическая MPLS - маршрутизация ... 345
50.3 Динамическая MPLS - маршрутизация ... 346

51 VRF - Virtual Routing and Forwarding 347
51.1 Введение ... 347
51.2 Базовая настройка ... 347
51.3 VRF-Lite ... 347
51.4 MPLS-L3VPN ... 348

52 Криптография 353
52.1 Ключ доступа ... 353
52.2 Туннели Disec ... 358
52.3 Туннельные интерфейсы Ditun (ditun, ditap, ip6ditun, ip6ditap) ... 372

53 VRRP-кластер 377
53.1 Основные понятия ... 377
53.2 Настройка кластера ... 377

54 Отказоустойчивый кластер 381
54.1 Требования к оборудованию ... 381
54.2 Подготовка к организации кластера ... 382
54.3 Настройки кластера ... 383

54.4 Получение информации о кластере ... 385
54.5 Пауза в работе кластера ... 386
54.6 Синхронизация настроек между маршрутизаторами ... 386
54.7 Дополнительные команды ... 387

55 Балансировка прерываний 389

56 Обновление системы 393
56.1 DIP-пакеты ... 393
56.2 Инфраструктура DIP ... 393
56.3 Установка обновления ... 395
56.4 Параметры загрузки ... 396
56.5 Конфигурация системы и данные ... 398
56.6 Привязка данных ... 399
56.7 Миграция ОС ... 399
56.8 Экспериментальный файл конфигурации ... 400
56.9 Резервная копия пакета ОС ... 401

57 Обслуживание 403
57.1 Резервное копирование ... 403
57.2 Проверка файловых систем ... 405
57.3 Безопасная очистка внешнего носителя ... 405
57.4 Форматирование внешнего носителя ... 405
57.5 Сброс паролей в начальное значение ... 405

58 Приложение 407
58.1 Примеры конфигураций ... 407

RU.НКБГ.70007-01 90

1. Общие сведения

1.1 Функциональные возможности

Маршрутизатор Dionis-NXe представляет собой программно-аппаратный комплекс, включающий в себя аппаратную платформу и управляющую Операционную Систему (ОС Dionis-NX 2.0). ОС Dionis-NX может работать на широким спектре аппаратных платформ (Intel, Эльбрус, ВМ8-10 и др.). Характеристики используемых аппаратных платформ в документе не конкретизируются, приводится только описание команд ОС Dionis-NX. ОС Dionis-NX построена на базе ядра Linux 4.9.xx и обладает следующими функциональными возможностями.

1.1.1 Сетевые интерфейсы

Поддерживаются следующие типы интерфейсов:

- Ethernet (большой спектр плат, с поддержкой VLAN и QinQ);
- WiFi (в режиме клиента и в режиме точки доступа);
- HDCL (E1, платы parabel, в том числе в unframed режиме);
- bond (агрегирование каналов в режимах: rr, active-backup, 802.3ad, tlb, alb и другие, в разных режимах мониторинга состояний линий связи;
- bridge (работа в режиме коммутатора с поддержкой stp, igmp-snooping и igm-routing);
- ditun (поддержка туннельных интерфейсов L3 в режиме шифрования DISEC, с поддержкой icmp проб, автоопределением параметров (работа через NAT));
- ditap (поддержка туннельных интерфейсов L2 в режиме шифрования DISEC, с поддержкой icmp проб, автоопределением параметров (работа через NAT));
- gre (поддержка туннельных интерфейсов L3 в режиме GRE с поддержкой пинг-проб);
- gretap (поддержка туннельных интерфейсов L2 в режиме GRE с поддержкой пинг-проб);
- ip6ditun/ip6ditap (поддержка туннельных интерфейсов L2/L3 с шифрование ГОСТ (IPv6));
- ip6gre/ip6gretap (поддержка туннельных интерфейсов L2/L3 GRE IPv6);
- vxlan/ip6vxlan (поддержка технологии VXLAN);
- I2tp (реализована клиентская и серверная поддержка протокола L2TP);
- loopback (интерфейс - петля);
- modem (поддержка 3G/4G модемов);
- pppoe (реализована клиентская и серверная поддержка протокола PPPoE);
- pptp (реализрвана клиентская и серверная поддержка протокола PPTP);
- ppp (реализована клиентская и серверная поддержка PPP);
- vpn (реализована клиентская и серверная поддержка openvpn);
- serial controller (поддержка USB и COM портов).

Все интерфейсы поддерживают:

- произвольный набор адресов IPv4 (в т.ч. по DHCP) и IPv6;

- режим link-detect;
- правила фильтрации;
- правила NAT;
- правила модификации пакетов;
- правила QoS;
- полисер;
- правила RPS;
- L2 адрес;
- режим зеркалирования;
- режим отправки статистики по netflow (1, 5 и 9 версии протокола);
- mtu/multicast/arp и другие свойства.

1.1.2 Статическая маршрутизация

- Поддерживается статическая маршрутизация TCP/IP (v4 и v6) с метриками и автоматическим подниманием/опусканием маршрута в зависимости от состояния интерфейса (link-detect);
- Поддерживается расширенная статическая маршрутизация IPv4/IPv6 (policy route) со следующими возможностями:
    - поддержка icmp проб;
    - критерий выборки: ttl;
    - критерий выборки: адрес источника и/или адрес назначения;
    - критерий выборки: порты назначения и/или источника;
    - критерий выборки: интерфейс на который был принят данный сетевой пакет;
    - критерий выборки: мандатные метки МСВС;
    - критерий выборки: частота соединений;
    - критерий выборки: состояние соединения;
    - критерий выборки: состояние флагов TCP;
    - критерий выборки: MAC адрес источника;
    - критерий выборки: время/дата;
    - критерий выборки: tos/dscp;
    - критерий выборки: класс трафика;
    - другие критерии выборки;
    - маршрутизация осуществляется для первого пакета потока, все остальные пакеты потока маршрутизируются по выбранному правилу, что позволяет достичь высокой скорости маршрутизации даже для сложных правил отбора;
- Поддержка явного задания входов ARP таблицы IPv4;
- Поддержка явного задания входов NEIGHBOUR таблиц IPv6.

1.1.3 Динамическая маршрутизация

- Поддержка IGP;

- Поддержка OSPFv2;
- Поддержка OSPFv3;
- Поддержка BGPv4, v4+, v4-;
- Поддержка RIP/RIP2;
- Поддержка RIPNG;
- Поддержка LDP.

1.1.4 Поддержка мультикаст-трафика

- Поддержка статической мультикаст-маршрутизации;
- Поддержка DVMRP;
- Поддержка IGMP;
- Поддержка PIM.

1.1.5 Фильтрация

- Правила фильтрации задаются в виде списков;
- Поддерживается передача управления от спискам к спискам в режиме безусловного перехода или с возвратом управления;
- Правила фильтрации могут применяться на интерфейсах (с явным заданием направления трафика), на прием, на отправку и на маршрутизацию;
- Поддержка протоколов IPv4/IPv6;
- Поддерживается широкий набор критериев отбора и их комбинаций, применительно к отдельному интерфейсу или системе в целом:
    - протокола;
    - адресов и портов источника/назначения;
    - тас-адреса источника;
    - текущего времени;
    - поля TOS/DSCP;
    - содержимого поля данных пакета;
    - состояния соединения;
    - состоянию флагов TCP;
    - содержимому;
    - ttl;
    - с учетом недавней сетевой активности (списки недавних адресов и событий);
    - и другие критерии (около 20).

1.1.6 NAT

- Поддержка различных вариантов трансляции IP/IPv6-адресов (SNAT/DNAT):

- SNAT;
- DNAT;
- MASQUARADE (вариант SNAT);
- SMAP (SNAT для сети);
- DMAP (DNAT для сети);
- исключение трафика из NAT.

1.1.7 Модификация трафика

- Правила модификации трафика задаются в виде списков;
- Спики могут быть установлены в те же места, что и списки фильтрации;
- Критерии отбора такие же, как и у списков фильтраций;
- Поддержка протоколов IPv4/IPv6;
- Поддерживаются следующие операции:

  - модификация TOS/DSCP;
  - модификация MSS;
  - модификация DF;
  - модификация ttl;
  - модификация и анализ CoS.

1.1.8 Поддержка и отслеживание соединений и преобразование сетевых адресов (NAT)

- Для передачи audio, video и т.д.;

  - sip;
  - ftp;
  - pptp;
  - snmp;
  - tftp;
  - q931;
  - netbios_ns;
  - ire;
  - sane;
  - h245;
  - amanda;
  - ras.

1.1.9 Классификация трафика

- Dionis-NXe поддерживает классификатор трафика, который позволяет относить один и тот же поток к разным классам (до 64).
- Классификатор трафика может быть использован для шейпинга и модификации трафика, а также его маршрутизации;
- Число списков классификации не ограничено, но одновременно может быть использовано 64 класса;
- Поддерживаются широкие критерии отбора.

1.1.10 Криптографическая защита

- Поддержка криптографической защиты данных, передаваемых по каналам связи сетей общего пользования, использующих протоколы семейства TCP/IP (v4/v6) (компоненты СКЗИ):
    - создание и поддержка статических криптотуннелей между узлами Dionis-NXe с шифрованием и имитозащитой передаваемых IP/IPv6-пакетов с инкапсуляцией их в протокол "IP в IP" или "UDP";
    - обеспечение функционирования туннельных интерфейсов или туннелей без создания интерфейсов.

1.1.11 MPLS

- Протокол LDP (Label Distribution Protocol) - протокол распространения меток согласно RFC-3036 и RFC-5036;
- Технология MPLS (Multi-Protocol Label Switching): статическое и динамическое назначение меток, используя LDP;
- Поле MPLS TC согласно RFC-5462 - Multi-Protocol Label Switching Label Stack Entry: "EXP" Field renamed to TC (Traffic Class) Field;
- Установка LSP (Label Switching Path) посредством LDP согласно RFC-3212, а также статическое описание пути коммутации меток - LSP;
- Организация L2 туннелей поверх MPLS с сигнализацией LDP с помощью интерфейсов Ditap, объединённых в Bridge-интерфейс с физическими Ethernet интерфейсами (с шифрованием и без шифрования);
- Организация L3 туннелей поверх MPLS с сигнализацией LDP с помощью интерфейсов Ditun, используя правила маршрутизации в туннельный интерфейс Ditun (с шифрованием и безшифрования).

1.1.12 QoS

- Поддежка полисера (ограничение полосы на вход/выход);
- Поддержка иерархичного шейпера HTB (любой глубины вложенности);

- Поддержка codel/fqcodel;
- Поддержка GRED;
- Поддержка RED;
- Поддержка SFQ;
- Поддержка prio (в режиме до 16 очередей, с заданием правил по tos или с помощью классификатора);
- Поддержка комбинирования политик HTB и prio с другими политиками в 2-х уровневой иерархии.

1.1.13 Протоколирование

- Поддержка протоколирования событий фильтрации IP/IPv6-пакетов;
- Поддержка протоколирования цикла обработки IP/IPv6-пакетов при прохождении их через маршрутизатор (трассировка);
- Поддержка настраиваемого механизма ротации журналов проходимого трафика;
- Поддержка выборочного протоколирования и механизмов оповещения;
- Протоколирование выполняется в реальном времени с гарантией того, что информация не будет потеряна. Возможна запись полного содержимого трафика или только заголовков;
- Режим слежения за системными журналами и реагирование на заданные администратором события (служба watcher):
    - генерации сообщения в системный журнал;
    - отправки сообщения по электронной почте;
    - выполнения команды по ssh;
- Поддержка звукового оповещения (для сообщений высокой важности).

1.1.14 Другие службы

- Поддержка сервера доменных имен (DNS)- работа в режиме первичного или вторичного DNS-сервера, наличие DNS-кэша и т.д.;
- Поддержка сервера динамической конфигурации узла (DHCP)- начальное конфигурирование рабочих станций локальных сетей (включая возможность задания нескольких шлюзов и севис DHCPRELAY);
- Поддержка прокси-сервера HTTP/FTP с возможностями прозрачного перехвата и фильтрации трафика;
- Поддержка сервера и клиента удаленного доступа SSH - обеспечение доступа к комплексу для управления им с удаленной консоли;
- Поддержка сервера/клиента синхронизации часов по сети (NTP) - служба синхронизации времени;
- Поддержка сервера удаленного мониторинга (SNMP);
- Поддержка управления устройством через WEB по HTTP(S) протоколу;
- Службы для оценки пропускной способности канала и SLA;
- Служба LLDP;

- Служба XMPP;
- Служба RA IPv6 (Router Advertisement Daemon);
- Реализация VRRP (v2/v3);
- Служба Telnet.

1.1.15 Обслуживание

- Контроль целостности конфигурации и прошивки;
- Поддержка процедур резервного архивирования и восстановления;
- Поддержка функционирования узлов Dionis-NXe в режиме отказоустойчивого аппаратного кластера с синхронизацией состояния соединений.

В качестве основной системы управления маршрутизатором используется интерфейс командной строки. Для настройки основных функций существует возможность использовать web-интерфейс.

1.2 Дополнительные технические данные

Маршрутизаторы, работающие под управлением ОС Dionis-NX, для повышения надежности могут быть объединены в кластер. Кроме стандартного способа дублирования, основанного на применении протокола VRRP (п. 53), Dionis-NX представляет возможность аппаратного дублирования маршрутизаторов (создание отказоустойчивого кластера, п. 54). Этот режим обеспечивает высокую скорость переключения с основного на резервный маршрутизатор в случае сбоя. Количество маршрутизаторов, используемых в кластере "горячего" резервирования - два. Оба маршрутизатора, работающих в составе кластера, должны быть одинаковыми и иметь одинаковую конфигурацию и должны быть соединены между собой. По этому соединению от работающего маршрутизатора передается информация, характеризующая состояние компонента TCP/IP (и не происходит, например, передачи логов).

Число поддерживаемых сетевых интерфейсов и число каналов обслуживания прикладных сервисов TCP/IP зависит от аппаратной части (объем ОЗУ и число разъемов на материнской плате).

Использование программно-аппаратного комплекса подлежит лицензированию. Лицензия может ограничивать использование программных и аппаратных возможностей оборудования.

Максимальное число одновременно установленных TCP/IP-соединений зависит от конфигурации аппаратного обеспечения.

2. Установка ОС Dionis-NX

Установка ОС Dionis-NX обычно осуществляется производителем оборудования, однако может производиться и с помощью специального загрузочного флеш-диска самостоятельно. Кроме установки ОС, диск может использоваться и для других целей: сохранение конфигурации, восстановление конфигурации, запись новых версий ОС, создание резервных копий и т.д. В случае утери такого диска клиент имеет возможность получить новый образ диска. Установочный флеш-диск содержит один раздел с файловой системой FAT32.

Флеш-диск в общем случае может содержать любые файлы и доступен как в ОС Linux, так и в ОС Windows. Директория install-images имеет специальное значение. В этой директории могут храниться сжатые образы ОС Dionis-NX. Файлы со сжатыми образами ОС обычно именуются как dionisnx-< версия >.<архитектура>.dip.

Как указано выше, на флеш-диск может быть выполнено резервное копирование текущей конфигурации и данных ОС Dionis-NX. В этом случае на флеш-диске появится специальная директория dionisnx-backup.

Несмотря на то, что, как правило, начальная установка ОС Dionis-NX выполняется производителем оборудования, возможны специальные ситуации, когда такая установка выполняется пользователем, например, при отсутствии связи с сетью Интернет. В этом случае пользователь должен иметь полностью сформированный установочный флеш-диск с требуемой версией ОС, полученный от производителя до начала установки. Каждый образ ОС Dionis-NX предназначен для конкретного оборудования. Невозможно использовать на данной машине образ ОС, предназначенный для другого экземпляра аналогичной машины. Для установки системы Dionis-NX следует загрузиться с установочного флеш-диска. Если маршрутизатор оборудован специальной платой "Сторож", необходимо полностью обесточить маршрутизатор и перевести эту плату в технологический режим (режим SE). Иначе эта плата отключит клавиатуру и USB-шину на время загрузки, и, соответственно, не удастся загрузиться с внешнего носителя.

После загрузки с установочного флеш-диска на экран будет выведен список возможных действий (для управления используется псевдографический интерфейс):

- Установка системы
- Обслуживание системы ->
- Выбор установочного диска
- Выбор целевого диска
- Журналирование ->
- Идентификатор платформы
- Диагностическая информация ->
- Перезагрузка
- Выключение компьютера

После установки ОС Dionis-NX необходимо перевести плату "Сторож" обратно в рабочий режим (режим JL).

Пункт "Установка системы" позволяет установить ОС Dionis-NX на жесткий диск. На одном жестком диске может быть установлено сразу несколько экземпляров ОС Dionis-NX, однако только один из установленных экземпляров будет активным в конкретный момент времени. Вновь

установленная версия автоматически становится активной. Если старая версия системы больше не требуется, ее можно впоследствии удалить с жесткого диска.

При выборе пункта "Установка системы", программа предложит выбрать диск (как правило, маршрутизатор имеет единственный диск), на который будет произведена установка ОС Dionis-NX. Затем установщик предложит выбрать образ ОС Dionis-NX, который следует установить. Как было сказано выше, в общем случае установочный флеш-диск может содержать несколько образов с различными версиями ОС. Если на диске маршрутизатора уже присутствуют установленные системы, то в следующем диалоговом окне можно указать, наследовать ли данные (слот данных) более ранних версий ОС для текущей установки.

После выполнения всех описанных действий, будет произведена установка выбранной системы на жесткий диск, а затем пользователю будет предложено извлечь установочный флеш-диск и перезагрузить систему.

3. Основы работы с интерфейсом командной строки

В качестве основного средства управления маршрутизатором используется интерфейс командной строки. После входа в систему пользователь может набирать на клавиатуре команды, которые выполняют различные действия, в т.ч. и меняют конфигурацию устройства.

Команды вводятся в ответ на приглашение системы, например, такое:

DionisNX> _

или

DionisNX# _

Здесь DionisNX - имя узла

Команды Dionis-NXe в общем случае состоят из двух частей: из имени команды и параметров. Параметры отделяются от имени команды и друг от друга пробелами. В команде может не быть ни одного параметра.

Для удобства пользователей команды в Dionis-NX разделены на группы по функциональному назначению. Каждая группа может содержать подгруппы (группы следующих уровней). В соответствии с этим имя команды может быть составным - состоять из нескольких "слов": первое слово - имя первой группы команд, второе - имя группы следующего уровня и т.д. Слова в команде разделяются пробелами.

При вводе и редактировании команд можно использовать следующие клавиши:

- <TaЬ> или <Ctrl^I> - для автоматического дополнения имени команды или параметра (при однозначном варианте сразу выполняется дополнение; если возникает возможность выбора, выводится список вариантов);
- <?> - для вывода на экран списка команд или параметров, доступных в настоящий момент; список выводится вместе с краткой справкой по этим командам/параметрам;
- <стрелка вверх> - для вывода на экран предыдущих команд (для просмотра или повторного ввода);
- <Shift+PgUp/PgDn> - для постраничного просмотра содержимого экрана;
- <Ctrl^z> - выход на уровень выше в дереве вложенных настроек (режим конфигурации). Соответствует выполнению команды "exit";
- <Ctrl^Space> - просмотр конфигурации текущего уровня настроек (режим конфигурации). Соответствует выполнению команды "show";
- <Ctrl^C> - отмена ввода и переход на новую строку;
- <Home> или <Ctrl^A> - переход в начало строки;
- <End> или <Ctrl^E> - переход в конец строки;
- <Del> или <Ctrl^D> - удаление текущего символа;
- <Backspace> или <Ctrl^H> - удаление предыдущего символа;
- <Ctrl^L> - очистка экрана;
- <Ctrl^J> или <Ctrl^M> - ввод. Соответствует нажатию клавиши Enter ;
- <Ctrl^W> - удаление слова;

- <Ctrl^K> - удаление всей строки справа от курсора и копирование удаленной части в буфер;
- <Ctrl^U> - удаление всей строки слева от курсора и копирование удаленной части в буфер;
- <Ctrl^Y> - вставка из буфера.

Если по какой-либо команде на экран выводится длинный текст (командная строка оказывается за пределами экрана), то при просмотре такого текста можно использовать следующие клавиши:

- <стрелка вверх>/<стрелка вниз> - для перехода на предыдущую/последующую строку;
- <PgUp>/<PgDown> - для постраничного просмотра;
- <! — > - для выхода в режим командной строки.

Заканчивается ввод команды нажатием клавиши Enter .

Если строка начинается с символа то она содержит комментарий.

3.1 Режимы работы с системой

Все действия в ОС Dionis-NX всегда производятся от имени какой-либо учетной записи (п. 6). Перед началом работы пользователь должен войти в систему - ввести свое имя (имя учетной записи) и затем ввести свой пароль.

По умолчанию в системе существуют две учетные записи - учетная запись для получения консольного доступа к системе (учетная запись оператора с именем "cli") и учетная запись для администрирования ("adm"). В заводских настройках ОС Dionis-NX (если не было специальных указаний заказчика) для этих учетных записей установлены пароли, совпадающие с их именами (cli и adm соответственно).

При первом входе администратора adm в систему ему будет предложено сменить оба заводских пароля (cli и adm) на другие, которые и будут использоваться в дальнейшей штатной работе. Порядок смены пароля учетной записи описан в п. 6.3.3.

Администратор имеет возможность создать любое количество учетных записей администраторов (возможно, с разными правами доступа), поэтому в дальнейшем будет говориться об учетной записи администратора.

При входе в систему под учетной записью для получения консольного доступа к системе система предоставляет доступ к командам непривилегированного режима - это только часть информационных команд. С помощью команды "enable" можно перейти в привилегированный режим, но при этом потребуется указать имя учетной записи администратора и ввести пароль.

Администратор в системе имеет доступ к командам привилегированного режима enable. В этом режиме доступны команды управления, не меняющие конфигурацию системы. Для входа в режим конфигурации используется команда "configure terminal". Доступность команд конфигурирования для учетной записи администратора определяется совокупностью прав этой учетной записи.

В дальнейшем изложении будут использоваться следующие обозначения режимов командного интерфейса:

- user - режим непривилегированного пользователя (оператора);
- enable - режим администратора (разрешены команды, не меняющие конфигурацию системы);
- configure - основной режим конфигурирования (изменение текущей конфигурации);
- (остальные) - являются вложенными режимами конфигурирования.

Вложенные режимы конфигурирования возникают после задания команды конфигурирования какого-либо объекта, например, интерфейса. Конфигурирование этого объекта может использовать специфический набор команд, например, команды конфигурирования интерфейса отличаются для разных интерфейсов. Доступность этих команд для учетной записи администратора определяется в рамках ролевой модели (п. 7).

Из режима enable доступны все команды режима user.

Из режимов конфигурирования (основного и вложенных) можно выполнить все команды режима enable, снабдив их префиксом "do". Например, команда просмотра текущей версии и контрольных сумм имеет следующий формат:

В режиме enable:

```
# show version
```

В режиме configure:

```
(config)# do show version
```

3.2 Виртуальные консоли

Для удобства работы с системой реализована возможность одновременной работы на нескольких виртуальных консолях. Переключение с одной виртуальной консоли на другую выполняется нажатием клавиш <Alt+Fn>. На десятую виртуальную консоль (<Alt+F10>) (после ее активации клавишей "Ввод") выводится информация системы мониторинга (п.7.2.6, Журналирование и трассировка).

3.3 Пространства имен

ОС Dionis-NX может работать с файлами, расположенными:

- в файловом пространстве маршрутизатора;
- на внешних носителях;
- на FTP/TFTP-серверах и HTTP-серверах.

Файловое пространство системы Dionis-NX разделено на несколько непересекающихся пространств:

- Локальное дисковое пространство файлов работающей версии ОС (слот данных системы);

- Дисковое пространство файлов маршрутизатора, разделяемое между всеми версиями ОС, установленными в маршрутизаторе;
- Логи (журналы);
- running-config, startup-config, default-config - фиксированные файлы конфигураций (п. 3.4) (файловые объекты с фиксированными именами).

Более подробно описание слотов данных и разделяемого между всеми версиями ОС хранилища приведено в разделе п. 56.2.(ssp – там про DIP)

Ниже приведен список названий возможных пространств файлов (они используются как префиксы в командах, манипулирующих файлами для идентификации файлового пространства, в котором файл находится):

cdrom<число>:	носитель CD-ROM/DVD-ROM
floppy<число>:	носитель на гибких дисках
Flash<число>[.<раздел>]:	сменный носитель
file:	локальное файловое пространство работающей версии ОС (префикс, используемый по умолчанию, если он не указан)
share:	пространство файлов маршрутизатора, разделяемое между всеми версиями ОС, установленными в маршрутизаторе
config:	пространство конфигурации (обычно, не указывается явно)
log:	файловое пространство журналов
ftp:	файлы, доступные по протоколу FTP
tftp:	файлы, доступные по протоколу TFTP
http:	файлы, доступные по протоколу HTTP

Если обращение в команде происходит к файлам конфигурации, то никакой префикс перед именем файла конфигурации не ставится.

3.4 Работа с конфигурациями

Конфигурация представляет собой последовательность команд и определяет настройку системы.

В Dionis-NX существует три вида конфигурации:

default-config	заводская конфигурация системы
running-config	действующая конфигурация
startup-config	стартовая конфигурация

Заводская конфигурация (default-config) определяет заводские настройки системы. Она доступна только на чтение. Заводская конфигурация может быть использована для сброса всех текущих настроек Dionis-NX (установленных в процессе работы) и возврата к первоначальным

заводским настройкам.

Действующая конфигурация (running-config) определяет текущие настройки системы (настройки, которые действуют в данный момент). Если администратор вводит команду в режиме configure, то она в случае ее успешного выполнения немедленно влияет на действующую конфигурацию.

При выходе из системы/при перезагрузке действующая конфигурация будет потеряна. При необходимости её можно сохранить командой копирования (см. ниже).

Стартовая конфигурация предназначена для создания действующей конфигурации после включения/перезагрузки системы. Работа системы всегда начинается с выполнения команд стартовой конфигурации; успешно выполненные команды стартовой конфигурации автоматически заносятся в действующую конфигурацию. В результате конфигурация running-config через некоторое время после начала работы системы становится эквивалентной конфигурации startup-config, за исключением тех команд из startup-config, которые по каким-то причинам завершились с ошибкой и вследствие этого не были добавлены в running-config. Если в ходе дальнейшей работы администратор выполнит команды конфигурирования (например, вводя их с консоли), то стартовая и действующая конфигурация станут различаться.

3.5 Просмотр и копирование конфигураций

Просмотреть любую из конфигураций можно с помощью команды "show" с соответствующим параметром (команда режима enable):

```
#  show running-config
#  show startup-config
#  show default-config
```

Команда "show" без параметров, выполненная в режиме enable, эквивалентна команде "show running-config" и показывает действующую конфигурацию.

В системе реализован целый ряд команд, которые позволяют из режима enable просмотреть конкретные части действующей конфигурации, например:

"show interface <тип> <номер> config"	просмотр настроек интерфейса
"show ip access-list <имя> config"	просмотр списка доступа
"show ip route config"	просмотр статических маршрутов

Команда "do show" из основного режима конфигурации эквивалентна команде "do show running-config" и показывает всю действующую конфигурацию. Если команда "do show" вызывается в одном из вложенных режимов конфигурации, то она покажет только часть действующей конфигурации, относящейся к данному режиму/конфигурируемому объекту.

Конфигурациями можно оперировать с помощью команды "copy" в режиме enable (или "do copy" в режиме configure) (п. 3.7.2).

Как было сказано выше, действующая конфигурация не сохраняется при завершении работы системы. Кроме того, может возникнуть необходимость иметь несколько вариантов действующей конфигурации. Для сохранения действующей конфигурации ее можно скопировать в стартовую конфигурацию, а также - в файл.

Стартовую конфигурацию можно заменить конфигурацией из файла. Заводскую конфигурацию можно скопировать в стартовую. Любую конфигурацию можно скопировать в файл.

Примеры команд:

"copy running-config startup-config"	Сохранение действующей конфигурации в стартовую конфигурацию
"write"	Эквивалент команды "copy running-config startup-config"
"copy running-config <имя_файла>"	Сохранение текущей конфигурации в файле
"write "	Эквивалент команды "copy running-config <имя_файла>"
"copy startup-config <имя_файла>"	Копирование стартовой конфигурации в файл
"copy <имя_файла> startup-config"	Замена стартовой конфигурации конфигурацией из файла

С помощью команды "copy" можно задать выполнение команд из стартовой конфигурации, из заводской конфигурации, а также из файла с последующим изменением действующей конфигурации.

Формат соответствующих команд:

"copy startup-config running-config"	Выполнение команд из startup-config (поверх старой running-config)
"copy <имя_файла> running-config"	Выполнение команд из файла (поверх старой running-config)

Копирование команд из файла в действующую конфигурацию является достаточно опасным. Команды из копируемого файла сразу же будут выполняться, и их действие может повлиять на результаты уже выполненных команд действующей конфигурации.

3.6 Команды режима configure и их влияние на действующую конфигурацию

Команды в режиме configure по способу их воздействия на действующую конфигурацию делятся на три основных типа:

- уникальные команды;
- списковые команды;
- отменяющие команды.

Для всех типов команд действует правило: если команда не нарушает консистентность конфигурации - она меняет running-config. Реальное применение действующей конфигурации может произойти позже. Например - задание ip адреса интерфейса Wifi может производиться до того, как данный интерфейс будет реально создан и начнет обрабатывать сетевые пакеты. Таким

образом, выполнение команды режима configure это конфигурация системы, а применение конфигурации в рамках ОС происходит асинхронно в те моменты времени, когда это необходимо.

При выполнении команды возможно получение диагностических сообщений следующих типов:

- Info: команда выполнилась успешно, предоставляется дополнительная информация по выполнению команды;
- Warning: команда выполнилась успешно, но с некоторыми замечаниями, которые могут означать логическую ошибку администратора;
- Error: при применении команды возникла ошибка. Невозможно применить данную конфигурацию;
- Fatal: критическая ошибка.

Уникальные команды после выполнения добавляются в текущую конфигурацию. Если аналогичная команда уже существовала в конфигурации, то она будет заменена командой с новыми параметрами.

Списковые команды после выполнения добавляются в текущую конфигурацию. Старые аналогичные команды не удаляются. В результате в конфигурации получается список аналогичных команд с разными параметрами. В некоторых случаях порядок команд в списке может иметь значение.

Отменяющие команды после выполнения не добавляются в текущую конфигурацию, они служат только для удаления из конфигурации других команд.

Пример уникальной команды ("hostname <имя_хоста>" - заменить имя хоста):

```
(config)# hostname Routerl

(config)# do show
!по команде "просмотреть" выводится вся действующая конфигурация

hostname Routerl

(config)# hostname DionisNX
(config)# do show

hostname DionisNX

!в конфигурации заменена команда "hostname" с параметром "<имя_хоста>"
```

Пример списковой команды ("ip secondary-address <1P-адрес>" - задать вторичный IP-адрес интерфейса):

```
(config)# interface ethernet 0
!выполнен переход в следующий (вложенный) режим конфигурации
(config — if—ethernetO)# do show
!по команде "просмотреть" на экран выводится только часть конфигурации
ip address 192.168.56.3/24
enable
(config —if—ethernetO)# ip secondary—address 192.168.56.4/24
```

```
 (config — if— ethernetO)# do show
ip address 192.168.56.3/24
ip secondary—address 192.168.56.4/24
enable
 (config —if—ethernet0)# ip secondary—address 192.168.56.5/24
 (config —if—ethernet0)# do show
 ip address 192.168.56.3/24
 ip secondary—address 192.168.56.4/24
 ip secondary—address 192.168.56.5/24
 enable

! в конфигурацию добавлены две аналогичные команды с именем "ip secondary—address" и
        разными параметрами
```

Пример отменяющей команды ("no ip secondary-address <IP-адрес>" - удалить указанный вторичный адрес интерфейса):

```
 (config —if—ethernet0)# no ip secondary—address 192.168.56.5/24
 DionisNX(config — if—ethernet0)# do show
 ip address 192.168.56.3/24
 ip secondary—address 192.168.56.4/24
 enable
 (config —if—ethernet0)# no ip secondary—address 192.168.56.4/24
(config —if—ethernet0)# do show
ip address 192.168.56.3/24
enable
```

Режим конфигурирования организован по принципу "дерева". Из основного режима можно перейти в первый вложенный режим (с помощью команд), затем во второй и т.д. На каждом уровне вложенности формируется своё приглашение на ввод команды - приглашение текущего (вложенного) режима конфигурации. Будем называть его приглашением "текущего контекста".

Чтобы выйти из вложенного режима в вышестоящий, можно ввести команду "exit" или команду из любого режима меньшего уровня вложенности - в этом случае будет выполнен переход на этот уровень вложенности. Команды всех уровней вложенности одной ветви дерева не пересекаются по именам.

Внутри одного уровня вложенности команды заносятся в действующую конфигурацию в определенном порядке. Очередность определяется, в первую очередь, приоритетом команды (приоритет является атрибутом команды, присваивается командам разработчиками системы). Вне зависимости от типа команды, сначала располагается команда с более высоким приоритетом. Если приоритеты у двух команд одинаковые, то для их размещения в конфигурации (внутри одного уровня вложенности) действуют следующие правила:

- Если из двух команд одна или обе уникальные, то они располагаются в алфавитном порядке;
- Если обе команды являются списковыми и хотя бы у одной из них неважен порядок ввода, то они располагаются в алфавитном порядке;
- Если обе команды являются списковыми и у обеих важен порядок ввода, то они располагаются в порядке ввода.

Это означает, что при сохранении, например, действующей конфигурации в файл, команды могут в нем оказаться не в том порядке, как они выполнялись при добавлении их в действующую конфигурацию.

Пример вложенных режимов конфигурирования (настройка 4-х сетевых интерфейсов)

```
# configure terminal

! — конфигурируем первый интерфейс
(config)# interface ethernet 0 (config—
if—ethernet0)# enable
(config — if—ethernet0)# ip address 1.1.1.1/24
(config—if—ethernet0)# exit

! — вышли в вышестоящий режим командой exit
(config)# _

! — конфигурируем следующий интерфейс
(config)# interface ethernet 1 (config—if—
ethernet1)# enable
(config —if—ethernet1)# ip address 2.2.2.2/24

! — переходим на предыдущий уровень заданием команды вышестоящего режима —
        например, команды "ip forwarding" (включить транзит).
(config —if—ethernetl)# ip forwarding
(config)# _

! — конфигурируем следующий интерфейс
(config)# interface ethernet 2
(config —if—ethernet2)# enable
(config —if—ethernet2)# ip address 3.3.3.3/24

! — сразу переходим к конфигурации следующего интерфейса заданием команды
вышестоящего режима
(config — if—ethernet2)# interface ethernet 3

! — система выполнила неявный переход на верхний уровень и сразу переход во вложенный
режим, но к другой ветви "дерева" команд
(config—if—ethernet3)# enable
(config —if—ethernet3)# ip address 4.4.4.4/24

! — просматриваем часть действующей конфигурации в данном контексте
(config — if—ethernet3)# do show
ip address 4.4.4.4/24
enable
! — просматриваем всю конфигурацию
(config—if—ethernet3)# exit
(config)# do show
!
hostname DionisNX
```

```
!
interface ethernet 0
ip address 1.1.1.1/24
enable
!
interface ethernet 1
ip address 2.2.2.2/24
enable
!
interface ethernet 2
ip address 3.3.3.3/24
enable
!
interface ethernet 3
ip address 4.4.4.4/24
enable
!
ip forwarding
```

Из примера видно, как меняется приглашение системы на ввод команды в зависимости от режима конфигурирования.

Приглашение основного режима конфигурирования имеет вид:

```
(config)#
```

приглашения следующего (первого) уровня вложенности имеют вид:

```
(config — if— ethernet0)#
(config—if—ethernet1)#
```

и т.д.

Приглашение каждого "текущего контекста" указывает администратору, в каком режиме он находится в данный момент.

Из примера также видно, что при просмотре полной конфигурации вложенные команды выводятся на экран с отступами.

Обратите внимание, что команда "ip forwarding" оказалась в конце конфигурации - она имеет самый низкий приоритет.

3.7 Команды работы с файлами

Команды работы с файлами доступны только администратору из режима enable.

3.7.1 Просмотр файлов

Для просмотра файлов используется команда "ls".

По команде с параметром «*» на экран выводится список доступных внешних устройств и доступных пространств файловой системы, например:

```
DionisNX# Is *
config:
file:
flash0:
ftp:
http:
```

Команда "Is" с названием устройства в качестве параметра позволяет просмотреть содержимое указанного устройства.

Например, для просмотра файлов на сменном носителе служит команда:

```
DionisNX# Is flash0:
```

Для просмотра списка файлов в директории PUB на ftp-сервере (пусть адрес сервера 192.168.33.160) служит команда:

```
DionisNX# Is ftp://192.168.33.160/pub
```

Если параметр команды "Is" не содержит названия устройства, то подразумевается, что файл находится в локальном файловом пространстве. Т.е для просмотра списка файлов в локальном файловом пространстве можно использовать команду с параметром "file:", "/" или команду без параметра:

```
DionisNX# Is file:
```

или

```
DionisNX# Is
```

или

```
DionisNX# Is/
```

3.7.2 Копирование

Для копирования файлов используется команда: "copy <откуда> <куда>".

Использование этой команды для работы с конфигурациями описаны выше (п. 3.4. Кроме путей, включающих в себя имена устройств, команда copy может принимать следующие имена файлов:

default-config	заводская конфигурация системы по-умолчанию (только для чтения)
running-config	действующая конфигурация
startup-config	сохраненная конфигурация

Если аргумент команды copy не содержит префикса устройства, то подразумевается file: Таким образом, администратор с помощью команды copy может поддерживать набор конфигураций, копировать их на внешние носители или получать с внешних носителей. Приведем типовые при-

меры использования команды copy.

Сохранение конфигурации на флеш-носитель:

```
DionisNX(config)# do copy running—config flash0:/dionisnx—config
```

Получение конфигурации с ftp-сервера:

```
DionisNX(config)# do copy ftp://192.168.33.160/pub/dionisnx—config startup—config
```

Копирование конфигурации в локальное файловое пространство:

```
DionisNX(config)# do copy startup—config config
```

Копирование журналов:

```
DionisNX(config)# do copy log:/auth.log flash0:/log
```

Просмотр списка файлов в локальном файловом пространстве:

```
DionisNX(config)# do Is file:
```

При использовании команды copy для некоторых типов устройств допустимо указывать в качестве источника и приемника каталоги. При этом каталог-приемник интерпретируется следующим образом. Если каталог-приемник не существует - то исходный каталог будет скопирован под именем каталога-приемника. Если каталог-приемник существует - то содержимое исходного каталога будет скопировано в каталог-приемник.

При копировании по протоколу SSH (п. 34) используется не команда copy, а команды ssh get и ssh put (п. ??sshgetput)}.

3.7.3 Контрольная сумма файла

Просмотр контрольной суммы (ГОСТ-2489) файла осуществляется следующей командой:

```
DionisNX(config)# do gostsum file:/example.tar.gz
```

3.7.4 Другие команды

К другим командам относятся:

rm <что>	удаление файла или каталога (рекурсивно)
mkdir <что>	создание каталога
less <что>	просмотр содержимого файла (с возможностью прокрутки вверх-вниз)
cat <что>	вывод содержимого файла на экран (без возможности прокрутки)

Например:

```
DionisNX# mkdir saved
DionisNX# copy running—config saved/1.config
```

```
DionisNX# ls saved
DionisNX# less saved/l.config
DionisNX# cat saved/l.config
DionisNX# rm saved
```

4. Выключение и перезагрузка

Чтобы выключить маршрутизатор Dionis-NX, нужно выполнить команду привилегированного режима:

```
# poweroff
```

Нажатие кнопки выключения на корпусе эквивалентно данной команде.

Так как данная команда довольно опасна, администратор должен будет подтвердить свои намерения в ответ на заданный вопрос. Чтобы принудительно выключить маршрутизатор (без необходимости ответа на вопрос), используйте:

```
# poweroff force
```

Для перезагрузки марщрутизатора нужно выполнить команду:

```
# reboot
```

При выключении/перезагрузке вся несохранённая текущая конфигурация (running-config) будет потеряна. При повторной загрузке системы будет применена сохранённая конфигурация (startup-config). Во время выполнения команды reeboot, если имеется несохранённая конфигурация, администратору потребуется подтвердить свои намерения в интерактивном режиме. Чтобы принудительно перезагрузить Dionis-NX используйте команду:

```
# reboot force
```

5. Предварительная настройка

5.1 Начало работы (Установка и смена паролей)

В самом начале работы с Dionis-NXe необходимо сменить пароли администратора и оператора. Порядок смены паролей и настройки учетных записей описан в п. 6.

5.2 Общие настройки

Предварительная настройка Dionis-NX включает в себя:

- установку имени узла;
- установку времени и часового пояса;
- нумерацию сетевых интерфейсов.

Имя узла задаётся с помощью команды hostname из режима конфигурации:

```
DionisNX(config)# hostname router—1
router—l(config)#
```

Часовой пояс задаётся с помощью команды timezone из режима конфигурации:

```
(config)# timezone MSK—3
```

При задании часового пояса необходимо ввести буквенную аббревиатуру часового пояса (MSK, OMST, VLAT и т.д.) и часовое смещение (со знаком + или -), которое необходимо прибавить к локальному времени, чтобы получить время UTC. Например, OMST-7, PST+8. По умолчанию в Dionis-NX задан часовой пояс MSK-3.

Чтобы задать время и дату, из привилегированного режима нужно выполнить команду clock. Например:

```
#  clock 13:58 27 02 2018
```

Время и дата задаются в формате «часы:минуты[:секунды] число_ номер месяца_год».

Чтобы скорректировать время без изменения даты, следует выполнить команду:

```
#  clock 13:58
```

Посмотреть текущую дату, время и часовой пояс можно с помощью команды непривилегированного режима:

```
> show clock
```

Нумерация сетевых интерфейсов (сопоставление имён «ethernet <n>» с MAC-адресами) обычно делается заводом-изготовителем. Однако, если по каким-то причинам потребуется заново перенумеровать интерфейсы, то это можно сделать с помощью интерактивной команды привилегированного режима:

```
#  interface enumerate ethernet
```

После подачи команды будет предложено сопоставить MAC-адреса с номерами. Чтобы введенные изменения вступили в силу, необходимо будет произвести перезагрузку:

```
#  reboot
```

Порядок нумерации интерфейсов может не совпадать с их физическим расположением на передней панели маршрутизатора. Поэтому, администратор может посчитать удобным перенумеровать интерфейсы в естественном порядке. В случае, если необходимо перенумеровать некоторые интерфейсы, следует воспользоваться командами:

```
 show interface bindings
interface blink ethernet <n>
interface bind ethernet <n> <mac>
```

Первая из этих команд покажет список всех интерфейсов с указанием MAC-адресов. Чтобы посмотреть, где физически расположен определенный интерфейс (с номером <n>) на панели маршрутизатора, следует применить вторую из команд. Если у интерфейса есть на панели цветовой индикатор, то он будет "мигать". После этого, с помощью третьей из команд можно задать новый номер этого интерфейса. Последовательность второй и третьей команд следует повторить столько раз, сколько интерфейсов следует перенумеровать.

5.3 Включение IPv6

Если маршрутизатор должен работать в IPv6 сетях, необходимо активировать реализацию протокола IPv6. Для этого, в configure режиме выполните команду: ip6 enable

```
(config)# ip6 enable
```

После этого, необходимо выполнить перезагрузку командой reboot (предварительно записав конфигурацию).

Для того, что бы включить режим передачи IPv6 пакетов с интерфейса на интерфейс, выполните команду: ip6 forwarding из режима configure.

```
(config)# ip6 forwarding
```

Для выключения работы с IPv6 выполните команды: no ip6 forwarding и ip6 disable. Затем выполните перезагрузку системы.

```
 (config)# no ip6 forwarding
(config)# ip6 disable
(config)# do write
(config)# do reboot
```

5.4 Интерфейсы

В данном разделе рассматривается базовая настройка сетевых интерфейсов Ethernet. Однако многие команды настройки интерфейса типа ethernet существуют и для других типов интерфейсов, например pptp, I2tp и др., и работают похожим образом. Это такие команды, как: disable, enable, ip address и многие другие.

Для настройки конкретного интерфейса необходимо ввести команду в режиме конфигурации:

(config)# interface ethernet номер_интерфейса

Данная команда осуществляет вход в режим конфигурации интерфейса.

Следующие команды выполняют минимально необходимую настройку интерфейса (активация и назначение IPv4-адреса/маски подсети):

(config — if—ethernet0) # ip address 192.168.1.1/24
(config—if—ethernet0) # enable

Если интерфейс будет настраиваться с использованием службы DHCP (п. 28), то необходима команда:

(config —if—ethernetO)# ip address dhcp

В этом случае будет невозможно использовать часть команд настройки системы разрешения имен:

- ip resolver nameserver
- ip resolver domain

Чтобы использовать любые команды системы разрешения имен и одновременно иметь возможность получить IP-адрес по DHCP, следует подать команду:

(config —if—ethernetO)# ip address dhcp iponly

Данная команда только присваивает интефейсу IP-адрес, не трогая остальные сетевые настройки, такие как адрес сервера имен (DNS-сервер) и др.

Если необходимо, интерфейсу может быть назначено несколько IP-адресов из разных сетей.

Пример:

(config —if—ethernet)0# ip address 10.1.1.1/24
(config —if—ethernet0)# ip address 10.2.2.2/24

Если необходимо задать несколько адресов для одной сети, используйте команду ip secondary-address (только для IPv4):

(config —if—ethernet0)# ip address 10.1.1.1/24
(config —if—ethernet0)# ip secondary—address 10.1.1.2/24
(config —if—ethernet0)# ip address 10.2.2.2/24
(config —if—ethernet0)# ip secondary—address 10.2.2.3/24

Команды с префиксом «no» удаляют соответствующие настройки или возвращают значения по умолчанию.

Для задания IPv6 адресов, используйте команды ip6 address и no ip6 address.

При необходимости можно настроить другие параметры интерфейса с помощью команд:

- multicast - режим групповой передачи;
- speed - скорость интерфейса;
- mac - изменить MAC-адрес по умолчанию;
- mtu - изменить MTU;
- arp - запрет/разрешение ARP-обмена.

Для просмотра текущей конфигурации интерфейса можно ввести команду (из текущего режима):

(config — if— ethernetO)# do show

или из привилегированного режима:

show interface ethernet номер

Для вывода текущего состояния интерфейса и статистики по интерфейсу можно использовать команды привилегированного режима:

show interface ethernet номер
show interface ethernet номер link
show interface ethernet номер stat
show interface ethernet номер device
show interface ethernet номер device stat

Вторая из команд выводит информацию по текущему состоянию интерфейса, а третья - по статистике.

Если необходимо деактивировать интерфейс (без потери настроек), то нужно выполнить команду конфигурации интерфейса:

(config—if—ethernet0) # disable

Следующая команда режима конфигурации деактивирует интерфейс и удаляет все настройки:

(config)# no interface ethernet номер

Существует возможность дублирования всего входящего в интерфейс трафика на другой ethernet-интерфейс. Данная возможность может быть использована для последующего анализа или мониторинга трафика сторонним ПО, которое получает доступ к входящему трафику, слу- шая его на выделенном интерфейсе. Для установки дублирования трафика следует использовать команду mirror, например:

(config —if—ethernet0)# mirror ethernet 1

Теперь, весь входящий трафик будет дублироваться (включая ethernet-заголовки) на интерфейс ethernet 1.

Для отключения режима дублирования трафика используется команда no mirror:

(config —if—ethernet0)# no mirror

5.5 Статическая маршрутизация

Для того, чтобы Dionis-NX мог выполнять функции IPv4 маршрутизатора, необходимо разрешить передачу транзитных пакетов от одного сетевого интерфейса к другому с помощью команды режима конфигурации:

(config)# ip forwarding

Данная команда уже присутствует в файле конфигурации по умолчанию (default-config). Процедура включения функции маршрутизации IPv6 описана в главе "Включение IPv6" (п. 8.6)

Если по каким-то причинам нужно запретить транзит пакетов между интерфейсами, то нужно выполнить команду:

(config)# no ip forwarding

В Dionis-NX могут существовать следующие типы IP-маршрутов:

- connected - маршруты, появляющиеся автоматически при назначении IP-адресов сетевым интерфейсам;
- static - принудительно назначенные статические маршруты;
- kernel - маршруты, загруженные в ядро системы, минуя систему конфигурации Dionis-NX;
- bgp, rip, ospf - маршруты, создаваемые соответствующими службами динамической маршрутизации.

Для добавления статических IPv4 маршрутов используется команда режима конфигурации «ip route ...». Для удаления статического маршрута используется аналогичная команда «no ip route

Формат команды:

[no] ip route <ip_prefix>|default <gw_ip>|<iface>|blackhole|reject|nullO [<distance>]

где:

- ip_prefix - A.B.C.D/M - шаблон IP-адресов назначения. Пакеты, IP-адреса назначения которых удовлетворяют данному шаблону, будут направляться по данному маршруту;
- default - маршрут по умолчанию (эквивалентно записи 0.0.0.0/0).
- <gw_ip> - A.B.C.D - IP-адрес соседнего маршрутизатора. Пакет будет направлен на данный маршрутизатор;
- <iface> - тип и номер интерфейса (например, ethernet 0). Пакет будет выпущен через данный интерфейс;
- blackhole, null0 - пакет будет удалён;
- reject - пакет будет удалён. Отправителю будет отправлено сообщение ICMP «Unreachable»;
- <distance> - административное расстояние (administrative distance). Данная величина имеет значение при выборе наиболее оптимального маршрута. (Имеет смысл вместе с динамической маршрутизацией).

Если задано несколько маршрутов, пересекающихся по адресам назначения, то более приоритетным будет более точный маршрут (с большей маской), а менее приоритетным - более общий маршрут (с меньшей маской). Маршрут по умолчанию (0.0.0.0/0) имеет наименьший приоритет.

Примеры настройки статических маршрутов:

Маршрут по умолчанию:

```
(config)# ip route default 192.168.1.1
```

Данная команда предписывает маршрутизатору направлять все проходящие/исходящие пакеты, не адресованные данному узлу и не попадающие под другие правила маршрутизации, на маршрутизатор 192.168.1.1.

Удаление статического маршрута:

```
(config)# no ip route default 192.168.1.1
```

Маршрут через интерфейс:

```
(config)# ip route 10.0.1.0/24 ethernet 1
```

Данная команда указывает маршрутизатору, что сеть 10.0.1.0/24 подключена непосредственно к интерфейсу ethernet 1, и что все пакеты, адресованные в данную сеть, будут направлены в данный интерфейс. (Если неизвестен MAC-адрес для IP-адреса назначения, то будет выполнен ARP-запрос через указанный интерфейс).

Тупиковый маршрут:

```
(config)# ip route 10.2.0.0/16 blackhole
```

Все пакеты, адресованные в сеть 10.2.0.0/16, будут отброшены.

Чтобы посмотреть все добавленные статические маршруты, нужно выполнить команду привилегированного режима:

```
#  show ip route
```

Для вывода информации о всех маршрутах нужно выполнить команду:

```
#  show ip route
```

Также имеется возможность выводить часть таблицы маршрутизации. Например:

```
#  show ip route summary
#  show ip route connected
#  show ip route static
#  show ip route 10.0.1.0/24
```

Данные команды выводят соответственно: количество маршрутов разных типов, только маршруты типа «connected», только статические маршруты, маршруты с префиксом назначения 10.0.1.0/24.

Для работы с IPv6 статическими маршрутами используйте команды с префиксом ip6: ip6 route, no ip6 route, show ip6 route и так далее.

5.6 Клиент DNS

Система Dionis-NX может рассматриваться не только как сервис, обеспечивающий различные сервисы клиентам, но и как клиент других сервисов, выполняющихся как на самой ОС, так и на других узлах(маршрутизаторах).

В данном разделе рассматривается настройка DNS-клиента ОС Dionis-NX. Эта настройка необходима, если будут использоваться команды системы (режима enable или режима configure, за исключением команд службы DNS), в качестве параметров которых вместо IP-адресов указываются доменные имена. Например, это могут быть такие команды:

```
DionisNX# ping factor-ts.ru
DionisNX# netperf np-server udp
```

Без правильно настроенной клиентской части DNS-системы, данные команды не смогут получить IP-адреса узлов, имена которых указаны в качестве их параметров.

5.6.1 Связь с DHCP и динамическими интерфейсами

Выполнение некоторых команд настройки клиента DNS невозможно, если один из интерфейсов системы настроен на обслуживание по DHCP. Это следующие команды:

```
(config)# ip resolver domain
(config)# ip resolver nameserver
```

Если необходимо использовать эти команды и применить DHCP (п. 28) на интерфейсе, то следует выполнить на этом интерфейсе команду ip address dhcp iponly.

В результате конфигурация сети, предлагаемая сервером DHCP (например, сервера имен и доменное имя), не будет использоваться, за исключением IP-адреса, который будет присвоен интерфейсу таким же образом, как и в случае использования для него команды ip address dhcp.

Взаимное использование resolver nameserver, dhcp и ppp dns:

- чтобы использовать ip resolver nameserver/domainlist: не должно быть команд ip address dhcp
- чтобы использовать ip address dhcp: не должно быть команд ip resolver nameserver/domainlist
- чтобы использовать ppp getdns (в динамических PPP-интерфейсах): если есть ip resolver nameserver, то полученные от провайдера адреса не будут использоваться

Команды ppp getdns и ip address dhcp могут быть использованы одновременно, в данном случае будут использоваться адреса серверов имен полученные последними по времени.

Чтобы настроить клиент DNS войдите в режим configure.

5.6.2 Базовая настройка клиента

Основные параметры клиента DNS:

- сервер имен, используемый системой для разрешения DNS-запросов (т.е. чтобы узнать IP-адрес узла, заданного по имени);
- доменное имя по умолчанию.

Рассмотрим пример настройки:

```
(config)# ip resolver domainlist zeta.int
(config)# 1 ip resolver domainlist factor—ts.int
(config)#
(config)# ip resolver nameserver 10.0.0.1
(config)# ip resolver nameserver 10.0.0.2
(config)# 1 ip resolver nameserver fcOO:: 1
(config)# ip resolver host 10.0.0.3 zeta.int zeta—alias.int
(config)# ip resolver host fc00::2 ipv6—server.int
```

В результате будет создана следующая конфигурация:

- список доменных имен (в порядке приоритета): factor-ts.int zeta.int;
- список IP-адресов серверов имен (в порядке приоритета): fc00::l, 10.0.0.1 10.0.0.2;
- список статической привязки IP-адресов к именам: имена zeta.int zeta-alias.int имеют адрес 10.0.0.3, а ipv6-server.int имеет адрес fc00: :2.

5.6.3 Дополнительная настройка клиента

В дополнительной настройке описаны различные опции клиента DNS, более подробно о которых можно узнать в подразделе **Команды настройки сервиса DNS.**

Кратко перечислим соответствующие команды:

```
(config)# ip resolver sortlist 10.0.1.0/24
(config)# ip resolver sortlist 10.0.2.0/24
(config)# ip resolver options attempts 3
(config)# ip resolver options ndots 2
(config)# ip resolver options timeout 3
(config)# ip resolver options ednsO
(config)# ip resolver options rotate
```

С помощью этих команд задаются следующие параметры сервиса DNS:

- первые две команды задают список сортировки: если имя соответствует нескольким IP-адресам, они будут возвращены в порядке определённом списком сортировки;
- число попыток запроса на сервер имен;

- минимальное число точек в имени домена, чтобы оно считалось абсолютным именем домена;
- начальный интервал ожидания ответа на запрос (в секундах);
- включение расширения DNS, позволяющего принимать/посылать сообщения DNS, размером больше 512 байт, по UDP-протоколу;
- включение механизма распределения нагрузки, связанной с разрешением имен, между серверами имен,которые указаны командами ip resolver namserver.

5.7 Просмотр и настройка ARP-таблицы

Чтобы вывести текущую таблицу соответствия IPv4- и MAC-адресов соседних узлов, нужно выполнить непривилегированную команду:

```
> show ip arp
```

Следующая команда выводит это соответствие для конкретного IP-адреса:

```
> show ip arp 192.168.1.1
```

Для очистки всей текущей ARP-таблицы (кроме принудительных соответствий IP-MAC) используется команда привилегированного режима:

```
# clear ip arp
```

Также можно удалить соответствие IP-MAC для конкретных адресов. Например:

```
# clear ip arp 192.168.1.1
```

Если необходимо установить принудительное соответствие IP-MAC для некоторых узлов, следует выполнить в режиме конфигурации команду:

```
(config)# ip arp <ip_addr> <mac_addr>
```

Удалить принудительное соответствие IP-MAC можно с помощью команды:

```
(config)# no ip arp <ip_addr>
```

Принудительные соответствия IP-MAC также отображаются командой «show ip arp» вместе с временными соответствиями. Чтобы отобразить только принудительные соответствия, можно использовать команду привилегированного режима:

```
# show ip arp config
```

Во время своей работы Dionis-NX поддерживает таблицы с информацией о хостах находящихся в том же сегменте сети, что и маршрутизатор. По умолчанию, максимальное число записей в таблице равно 8192. В некоторых случаях этого может оказаться недостаточно, поэтому существует команда, позволяющая изменить настройки кеша. Синтаксис команды:

ip arp thresh <минимальная граница> <ватерлиния> <максимальная граница>

Минимальная граница - это то число записей в кеше, которые могут находиться постоянно;

- ватерлиния - при достижении размера кеша, равного этому параметру, будет запущен сборщик мусора;
- максимальная граница - максимальное количество записей в кеше.

Например:

```
# ip arp thresh 1024 8192 16384
```

По этой команде устанавливается максимальный размер кеша в 16384 записей. Нижняя граница - 1024. Ватерлиния - 8192.

Для сброса параметров в первоначальное состояние следует использовать команду no ip arp thresh. Для просмотра - команду show ip arp thresh (из режима enable).

5.8 Просмотр и настройка таблицы соседей IPv6 (neighbours)

Команды работы с таблицей соседей IPv6 очень похожи на команды работы с ARP кеш-таблицей.

Команда	Назначение
[no] ip6 neighbour <адрес> <mac адрес>	Задание/отмена ручной привязки IPv6 адреса соседа
ip6 neighbours thresh ...	Задание параметров кеша
clear ip6 neighbours ...	Очистка кеша
show ip6 neighbours ...	Показать информацию о таблице

5.9 Настройки стека TCP/IP

Данные настройки могут повлиять на производительность и работу различных подсистем и служб Dionis-NX. Поэтому их следует использовать с особой аккуратностью.

Настройки осуществляются из режима configure.

5.9.1 Настройки протокола IP

Маршрутизация транзитных IP-пакетов, т.е. пакетов, не предназначенных для данной системы, называется IP-форвардинг. Если опция IP-форвардинга не включена, то система не будет пересылать транзитные пакеты через свои интерфейсы и будет обрабатывать пакеты, адресованные только ей.

IP-форвардинг включается командой:

```
(config)# ip forwarding
```

Обычно на маршрутизаторах всегда следует включать IP-форвардинг.

Процедура включения функции маршрутизации IPv6 описана в главе "Включение IPv6" (п. 8.6)

Следующей командой можно установить Time-To-Live для IP-пакета (т.е. максимальное число узлов, через которое может пройти данный пакет, прежде чем будет отброшен):

(config)# ip ttl 50

Установка таймаута сессии для неизвестных или неподдерживаемых протоколов уровня layer4 (все, что кроме TCP/UDP):

(config)# ip timeout 600

Следующая команда задает максимальное число обычных и транзитных соединений:

(config)# ip max—connections 10000

При оптимизации пропускной способности сетевой подсистемы могут оказаться полезными опции размеров буферов сокетов.

Следующая команда задает минимальное, заданное по умолчанию и максимальное значение буфера сокета для исходящих пакетов протоколов TCP и UDP (в байтах)

(config)# ip wmem 10000 40000 100000

Следующая команда задает минимальное, заданное по умолчанию и максимальное значение буфера сокета для входящих пакетов протоколов TCP и UDP (в байтах)

(config)# ip mnem 10000 50000 90000

Значения параметров этих двух команд требуют пояснения. При создании сокета ему выделяется буфер для отправки и приема. Эти команды задают размеры в байтах этих буферов (на примере ip mnem):

- минимальный (10000) : размер буфер не может быть снижен системой или пользователем ниже этого значения, т.е. это гарантированный размер буфера;
- заданный по умолчанию (50000) : размер буфера, выделяемый системой по умолчанию при создании сокета;
- максимальный (90000) : это максимальный размер буфера, который может быть выделен сокету.

5.9.2 Настройки протокола TCP

5.9.2.1 Базовая настройка

Период посылки сообщения keep-alive при соединении по протоколу TCP задается следующими командами:

(config)# ip tcp keepalive interval 50
(config)# ip tcp keepalive probes 5
(config)# ip tcp keepalive time 1000

В этом примере для проверки того, не сорвано ли соединение, каждые 1000 секунд будут посылаться до пяти Keep-Alive сообщений с интервалом в 50 секунд. Если даже на пятое сообщение ответа не пришло, соединение будет разрываться.

Чтобы включить режим БАСК (режим выборочных подтверждений. Selective Acknowlegment), следует выполнить команду:

(config)# ip tcp selective—ack

Использование выборочных подтверждений означает, что только те данные, которые не были получены, требуют повторной передачи, что повышает эффективность использования пропускной способности сети.

Чтобы включить режим syncookies, следует выполнить команду:

(config)# ip tcp syncookies

Использование этого режима позволяет защититься от DoS-атак типа SYN-спуфинг (посылке большого числа SYN-пакетов на систему.

Чтобы включить расширения TCP (RFC 1323) для сетей с большой пропускной способностью, следует выполнить команды:

(config)# ip tcp timestamps
(config)# ip tcp window—scaling

Чтобы включить ECN механизм (расширение TCP, RFC 3168), следует выполнить команду:

(config)# ip tcp ecn server—mode

Чтобы включить режим ABC механизма (расширение TCP, RFC 3465), следует выполнить комиду:
(config)# ip tcp abc aggresive

Установка таймаута установленной TCP-сессии:

(config)# ip tcp timeout established 432000

5.9.2.2 Настройка памяти

При тонкой настройке сетевой подсистемы бывает важно установить, как протокол TCP будет регулировать потребление памяти для своих нужд.

Регулирование осуществляется установкой минимального, среднего и максимального объема потребляемой памяти. Рассмотрим пример:

(config)# ip tcp mem 1000 50000 90000

Значения параметров задаются в 4kb-страницах. Опишем каждый из трех параметров команды:

- минимальный размер (1000) : если объем используемой TCP-протоколом памяти ниже 1000, протокол никак не будет снижать свое потребление;

- средний размер (50000): если объем используемой TCP-протоколом памяти выше 50000, протокол будет снижать свое потребление, пока не достигнет 1000.
- максимальный размер (90000): максимальный объем памяти, доступный для всех TCP-сокетов системы.

Аналогичная команда существует и для UDP протокола: она называется ip udp mem.

В настройках протокола TCP существует команда ip tcp mnem и ip tcp wmem, которые аналогичны командам ip mnem и ip wmem, рассмотренным выше. Однако в данном случае они задают размеры для буферов сокетов протокола TCP.

5.9.3 Настройки протокола UDP

При тонкой настройке сетевой подсистемы бывает важно установить, как протокол UDP будет регулировывать потребление памяти для своих нужд.

Регулирование осуществляется установкой минимального, среднего и максимального объема потребляемой памяти. Рассмотрим пример:

(config)# ip udp mem 1000 50000 90000

Значения параметров аналогичны команде ip tcp mem для протокола TCP.

Значения минимального, умалчиваемого и максимально размеров для буферов сокетов UDP неявно устанавливаются равными размерами, заданным командой ip rmem и ip wmem.

Установка таймаута UDP-сессии:

(config)# ip udp timeout 30

Установка таймаута UDP-сессии для UDP-потока:

(config)# ip udp timeout stream 180

5.9.4 Настройки протокола ICMP

Чтобы включить обработку ICMP-запросов типа ECHO, следует выполнить команду:

(config)# ip icmp echo

Чтобы включить обработку широковещательных ICMP-запросов типа ECHO, следует выполнить команду:

(config)# ip icmp broadcast—echo

5.9.5 Настройки протокола IPv6

Команда	Назначение
[no] ip6 autoconf disable|enable	Автоконфигурировать ли адреса
[no] ip6 tempaddr disable|enable|prefer	Настройка реализации RFC3041
[no] ip6 redirects accept|reject	Принимать ли редиректы
[no] ip6 mtu <mtu>	Задать mtu по умолчанию
[no] ip6 hopjimit <число>	Задать hop limit по умолчанию (64)

5.10 Основы работы со службами (сервисами)

Работа со всеми службами Dionis-NX имеет общие особенности.

5.10.1 Команды режима configure

Команда входа в конфигурацию службы:

(config)# service NAME

В данной команде и далее параметр NAME соответствует имени службы (например, dns или dhcp).

Команда запуска службы:

(config —NAME)# enable

Команда остановки службы:

(config —NAME)# disable

5.10.2 Команды режима enable

Просмотр информации о службе:

show service NAME ...

Изменение каких-либо данных службы, за исключением конфигурации (команда может отсутствовать):

service NAME ...

Команда быстрого перезапуска службы (команда может отсутствовать):

service NAME reload

Осуществляет быстрый перезапуск службы.

Эта команда может быть полезна для быстрой повторной инициализации службы, если поменялись какие-либо данные/настройки службы, которые администратор изменил командами режима

enable. Например, для службы DNS это может быть изменение в зонных файлах (произведенное из режима enable), для службы DHCP - изменение в базе данных выданных адресов (также произведенное из режима enable).

Эту команду можно также вызвать из режима configure в секции службы, она называется reload.

Команда обычного перезапуска службы (команда может отсутствовать):

service NAME restart

Осуществляет перезапуск службы.

Эта команда аналогична вызову команд disable и enable в секции службы в режиме configure.

Эту команду можно также вызвать из режима configure в секции службы, она называется restart.

5.11 Диагностика

Для диагностики проблем настройки и функционирования TCP/IP-сетей администратор может пользоваться довольно большим набором описанных ниже средств, большинство из которых доступны как непривилегированному пользователю, так и привилегированному пользователю (режим enable).

5.11.1 Утилита ping

Позволяет формировать icmp-пробы. В качестве обязательного параметра задается IP-адрес или имя хоста.

Для IPv6 используйте утилиту ping6

5.11.2 Утилита traceroute

Позволяет отследить маршрут, по которому движется пакет пробы.

5.11.3 Монитор sysmon

Позволяет следить за состоянием системы в реальном времени (только для enable режима). При доступе к локальной консоли, монитор можно активировать комбинацией Alt-F10. Затем активировать консоль, нажав "Ввод". Alt-Fl - осуществляет возврат на первую консоль. С помощью клавиши «пробел» - изменяется выводимая информация. Кроме этого, во время работы монитора, нажав клавишу h или ?, можно ознакомиться с описанием структуры выводимой информации и с краткой справкой по использованию sysmon.

5.11.4 Информация на LCD-мониторе

На LCD-мониторе показывается краткая информация о состоянии системы. С помощью клавиш на панели осуществляется навигация по информационным полям. Цвет индикатора в правой части панели индикатора описывает общее состояние системы:

- зеленый - нормальное функционирование;
- желтый - загрузка или выключение системы;
- красный - требуется внимание администратора.

5.11.5 Команды show

Существует множество команд show, которые могут использоваться администратором для выявления проблем. Ниже приводится список основных команд:

команда	краткое описание
show interface <тип интерфейса> <номер интерфейса >	информация о состоянии сетевого интерфейса
show interface <тип интерфейса> <номер интерфейса> link	информация о состоянии среды и низкоуровневых настройках интерфейса
show interface стип интерфейса> <номер интерфейса > stat	статистика интерфейса
show interface	информация обо всех интерфейсах
show interface stat	статистика по всем интерфейсам
show ip sock	информация по сокетам
show ip connections	информация по открытым соединениям и кэшу NAT
show ip stat	информация по 1P-статистике

Эти команды могут выполняться из любого режима, но в режиме configure требуется префикс do.

6. Учетные записи

При работе с ОС Dionis-NX существует два вида учетных записей — учетная запись для получения консольного доступа к системе и учетные записи для администрирования. Учетная запись для получения консольного доступа в системе всегда одна. Учетных записей для администрирования (учетных записей администратора) может быть несколько. Среди учетных записей администраторов выделяется учетная запись "adm", которая является учетной записью по умолчанию. Остальные учетные записи администратора создаются в ходе работы системы.

6.1 Учетная запись консольного доступа (учетная запись оператора)

Учетная запись консольного доступа (учетная запись оператора) - одна для всей системы и имеет имя "cli". Эту учетную запись невозможно удалить, но допустимо менять для нее пароль и изменять другие настройки, присущие учетным записям.

По умолчанию пароль для учетной записи оператора - "cli". При вводе системы в эксплуатацию необходимо сменить пароль для учетной записи оператора. Это может сделать администратор системы. Процедура смены пароля описана в п. 6.3.3 .

Учетная запись оператора позволяет просматривать некоторые параметры системы и часть информации о ее состоянии. Работа под учетной записью оператора не позволяет менять какие-либо настройки системы.

Администратор может войти в систему, используя учетную запись "cli", а при необходимости выполнения административных действий сменить непривилегированную запись на учетную запись администратора с помощью команды "enable", как показано на примере ниже:

```
DionisNX> enable ivanov
```

В данном примере "ivanov" - это имя учетной записи администратора. Если учетная запись администратора не указана явно, то будет использовано предопределенное имя "adm". Подробнее об учетной записи "adm" будет сказано далее.

6.2 Учетные записи администратора

В системе Dionis-NX может существовать множество учетных записей администратора. Все действия администратора отражены в системном журнале и привязаны к имени учетной записи.

Администраторы могут иметь различные права на изменение настроек системы. Например, одному администратору доступна настройка сетевых интерфейсов, а другому нет. Подробнее права доступа администраторов описаны в разделе "Ролевая модель", п. 7. Также администратор может иметь права супервизора. В этом случае ему доступны любые операции по настройке системы. По умолчанию, учетная запись "adm" имеет права супервизора. Как объявить администратора супервизором, описано в разделе по администрированию учетных записей. Необходимо ответственно относиться к назначению администратора супервизором, так как в этом случае система становится ему полностью подконтрольна.

При начале работы с системой Dionis-NX существует единственная учетная запись администратора "adm". Администратор "adm" является супервизором. Также как и учетную запись консольного доступа, учетную запись "adm" невозможно удалить, однако ее можно заблокировать, либо лишить прав супервизора. Это можно сделать после того, как будут настроены другие рабочие записи администраторов. По умолчанию пароль для учетной записи "adm" - "adm". При вводе системы в эксплуатацию необходимо сменить пароль этой учетной записи.

В случае утери пароля администратора и невозможности администрирования системы, существует способ сброса паролей учетных записей "cli" и "adm" в значения по умолчанию. Это можно сделать, загрузившись с инсталляционного флеш-диска. В случае сброса пароля одновременно сбрасываются и другие настройки этих учетных записей.

6.3 Управление учетными записями

Управление учетными записями выполняется в режиме enable.

6.3.1 Создание и удаление учетных записей

Создать или удалить можно только учетную запись администратора (кроме adm). Учетные записи консольного доступа создавать и удалять нельзя.

Создается учетная запись с помощью следующей команды:

```
DionisNX# account create ivanov
```

В этом примере будет создана учетная запись с именем "ivanov". Команда создания учетной записи может иметь параметры, как это показано на примере ниже:

```
DionisNX# account create ivanov realname "Иван Иванов" desc "Администратор" supervisor
```

В данном примере использованы три необязательных параметра команды. Параметр "realname" задает реальное имя администратора. Параметр "desc" задает текстовое описание для учетной записи. Параметр "supervisor" наделяет вновь созданную учетную запись правами супервизора. Необходимо осторожно относиться к использованию параметра "supervisor".

Удалить существующую учетную запись можно с помощью команды "account remove":

```
DionisNX# account remove ivanov
```

Команда удаляет учетную запись "ivanov". При удалении учетной записи не удаляются пользовательские данные учетной записи. Чтобы полностью удалить и учетную запись и ее пользовательские данные, используется параметр "purge" этой команды:

```
DionisNX# account remove ivanov purge
```

6.3.2 Просмотр учетных записей

Список существующих учетных записей можно получить с помощью следующей команды:

```
DionisNX# show account *
adm
cli
ivanov
```

Данный пример отображает существующие в системе три учетные записи: "adm", "cli", "ivanov".

Для просмотра подробной информации о выбранной учетной записи используется команда:

```
DionisNX# show account ivanov
realname "Иван Иванов"
description "Администратор"
supervisor
 expire period 99999 last 2015 4 16 warning 7
delegate @default
```

Значение отображаемых полей будет описано ниже, в разделе, посвященном настройке учетных записей (п. 6.3.4) .

В случае, если необходимо получить подробную информацию по всем учетным записям сразу, используется опция "verbose":

```
DionisNX# show account * verbose
account config adm
realname "Administrator"
description "Default administrator"
supervisor
  expire period 99999 last 2015 4 16 warning 7
delegate @default
account config cli
realname "Console"
description "Console access"
expire period 99999 last 2011 11 14 warning 7
delegate @default
account config ivanov
realname "Иван Иванов"
description "Администратор"
supervisor
  expire period 99999 last 2015 4 16 warning 7
delegate @default
```

6.3.3 Изменение пароля учетной записи

Пароль для учетной записи можно задать/изменить с помощью команды "passwd".

```
DionisNX# passwd ivanov
```

Система в интерактивном режиме попросит ввести новый пароль для учетной записи. Пароль может содержать любые символы латинского алфавита, цифры, знаки препинания. Длина пароля должна быть не меньше 8 символов. Если введенный пароль короче, смены пароля не произойдёт.

6.3.4 Настройки учетной записи

Настройки существующей учетной записи можно редактировать. Для входа в режим редактирования настроек учетной записи используется команда "account config", например для редактирования настроек учетной записи администратора "ivanov":

```
DionisNX# account config ivanov
```

В режиме редактирования настроек учетной записи существует набор специальных команд, позволяющих:

- Задать реальное имя владельца учетной записи;
- Задать описание учетной записи;
- Назначить или снять права супервизора;
- Заблокировать или снять блокировку с учетной записи;
- Задать срок действия пароля учетной записи;
- Задать дату последнего изменения пароля;
- Задать количество дней до истечения срока действия пароля, начиная с которого пользователь будет получать предупреждение о необходимости смены пароля;
- Изменять полномочия и роли, доступные учетной записи.

6.3.5 Реальное имя

Для задания реального имени владельца учетной записи используется команда "realname".

```
DionisNX(account—ivanov)# realname "Иван Иванов"
```

Данная настройка не является обязательной. Установленное значение реального имени может быть сброшено следующей командой:

```
DionisNX(account—ivanov)# no realname
```

6.3.6 Описание

Для учетной записи может быть задано описание. Это произвольное текстовое поле.

```
DionisNX(account—ivanov)# description "Администратор. Комната 256"
```

Данная настройка не является обязательной. Установленное значение описания может быть сброшено следующей командой:

```
DionisNX(account—ivanov)# no description
```

6.3.7 Супервизор

В обычном режиме учетной записи администратора могут быть доступны не все возможности системы. Какие именно возможности системы доступны администратору, определяется его полномочиями и ролями в рамках ролевой модели (п. 7). В случае, если администратор объявлен супервизором, ролевая модель игнорируется и ему доступны все без исключения полномочия системы. Объявить администратора супервизором можно следующей командой:

```
DionisNX(account—ivanov)# supervisor
```

Отменить права супервизора для учетной записи можно следующей командой:

```
DionisNX(account—ivanov)# no supervisor
```

Так как супервизору доступны любые возможности системы, то необходимо крайне ответственно подходить к назначению администраторам прав супервизора.

6.3.8 Блокировка учетной записи

Учетная запись может быть временно заблокирована. Если учетная запись заблокирована, то ее владелец не сможет войти в систему. В отличие от удаления учетной записи, блокировка позволяет сохранить все настройки учетной записи, включая пароль, и только временно запретить вход в систему.

Для блокирования учетной записи используется следующая команда:

```
DionisNX(account—ivanov)# disable
```

Для разблокирования учетной записи используется следующая команда:

```
DionisNX(account—ivanov)# enable
```

6.3.9 Срок действия пароля

Для учетной записи может быть ограничен срок действия пароля. Это делается в целях безопасности. Срок действия пароля определяется количеством дней с момента последней смены пароля.

```
DionisNX(account—ivanov)# expire period 90
```

В приведенном примере устанавливается, что пароль действителен 90 дней. Когда срок действия пароля истечет, владелец учетной записи обязан сменить пароль. Без смены пароля вход в систему будет блокирован.

В некоторых случаях бывает необходимо вручную задать дату последней (предыдущей) смены пароля.

```
DionisNX(account—ivanov)# expire last 2015 04 16
```

В примере "2015 04 16" - это год, месяц и день, соответственно. Существует синтаксис команды для задания текущей даты, как даты последней смены пароля:

```
DionisNX(account—ivanov)# expire last now
```

Система предоставляет возможность указать, за сколько дней до истечения срока действия пароля владелец учетной записи будет получать предупреждение о необходимости его смены. Следующая команда устанавливает количество дней до истечения срока действия пароля, определяющее момент времени, начиная с которого владелец учетной записи будет получать предупреждение:

```
DionisNX(account—ivanov)# expire warning 5
```

Администратор ivanov будет получать предупреждение о необходимости смены пароля, начиная с 5 дней до истечения срока действия пароля.

7. Ролевая модель

7.1 Права доступа учетной записи администратора

Если администратор не имеет статуса супервизора, то его права доступа при настройке различных параметров системы определяются назначенным ему списком полномочий и ролей. Если администратор имеет статус супервизора, то имеет доступ к любым возможностям системы вне ролевой модели.

Каждое из полномочий может определять доступ к:

- командам одной подсистемы, которые используются в каком-то конкретном режиме (например, к командам для одного из интерфейсов, используемых в режиме enable, или к командам в режиме конфигурирования и т.д.);
- к одной конкретной команде;
- к определенным операндам одной из команд.

В случае если какие-то полномочия определяются на группу команд, и есть другие полномочия на конкретную команду из этой группы, то для доступа к этой конкретной команде необходимо иметь все эти полномочия. Аналогично, если полномочия определены на отдельную команду, и есть другие полномочия на использование некоторых параметров этой команды, то для использования указанных параметров этой команды необходимо иметь все эти полномочия.

Отдельные полномочия могут быть связаны друг с другом. Подробнее о зависимости полномочий — в разделе 7.5.

Каждая роль представляет собой совокупность полномочий. Учетная запись администратора, имеющая какую-либо роль, получает доступ, определяемый всеми полномочиях этой роли. Все роли системы создаются в ходе ее настройки и функционирования. По умолчанию никаких ролей в системе нет.

Полные полномочия учетной записи — это все полномочия, назначенные этой учетной записи, плюс все полномочия всех ролей, которые имеет эта учетная запись. Подробнее о полномочиях и ролях можно узнать в разделах 7.2, 7.4, 7.5. По умолчанию учетная запись обладает полномочиями "©default". В основном, по умолчанию учетная запись получает права только на просмотр информации о системе.

7.2 Полномочия системы

Полномочия в системе являются предопределенными и не могут быть созданы пользователем. Однако набор полномочий может расширяться при появлении новых версий системы. Поэтому администратор должен учитывать в своей работе возможность появления новых полномочий при установке новой версии системы. Все предопределенные полномочия для удобства восприятия сгруппированы по подсистемам, к которым они относятся. При использовании полномочий полезно учитывать следующие мнемонические правила, используемые в именах полномочий: - Все имена полномочий начинаются с символа @ ; - Имена полномочий состоят из нескольких слов.

разделенных символом Первое из слов описывает подсистему, к которой относится данное из полномочий. Последующие слова, как правило, указывают на типы полномочий для указанной подсистемы.

Основные типы полномочий:

- conf — право на использование команд режима конфигурирования;
- oper — право на использование команд режима enable;
- show — право на использование команд просмотра информации;
- key — право на использование команд управления ключами;
- crypto — право на использование команд шифрования;
- server — право на работу с сервером в протоколах, предусматривающих наличие клиента и сервера;
- client — право на работу с клиентом в протоколах, предусматривающих наличие клиента и сервера.

Приведенные типы полномочий могут представлять собой иерархическую систему — например, право на работу с сервером может быть определено для конфигурирования сервера или только для просмотра информации сервера. В таких случаях имена полномочий могут представлять собой цепочку, состоящую более чем из двух слов. Например, с помощью имени @l2tp.server.conf задаются полномочия, которые позволяют конфигурировать сервер для протокола l2tp.

7.2.1 Крипто - средства

Имена полномочий	Описание
@key.conf	Базовые крипто-средства, управление ключами
@key. clear	Удаление ключей
@key.show	Получение информации о ключах
@disec.conf	Открытые туннели DiSEC (+ сжатие)
@disec.show	Открытые туннели DiSEC (без изменения конфигурации)
@disec.key	DiSEC-ключи
@disec. crypto	Шифрованные туннели DiSEC
@ca.conf	Конфигурация сервера DiCert
@ca.oper	Обслуживание сервера DiCert (без изменения конфигурации и БД)
@ca.show	Просмотр состояния и БД сервера DiCert

7.2.2 Сетевые настройки

Имена полномочий	Описание
@net.conf	TCP/IP-настройки, resolver, arp, conntrack, clear interface stat

Имена полномочий	Описание
@net.oper	TCP/IP-настройки, resolver, arp, conntrack, clear interface stat (без изменения конфигурации)
@net.show	TCP/IP-настройки, resolver, arp, conntrack, clear interface stat (только получение информации)
@net.tools	команды диагностики ping, traceroute (п. 5.11), whois, arping, nslookup (п. 27.8)

7.2.1 Контроллеры устройств

Имена полномочий	Описание
@serial.conf	Конфигурирование последовательного порта (RS232)
@serial.show	Получение информации о последовательном порте
@el.conf	Конфигурирование контроллеров E1
@el.show	Получение информации о контроллерах E1

7.2.2 Сетевые интерфейсы

Имена полномочий	Описание
@ethernet.conf	Конфигурирование Ethernet
@ethernet.bind	Привязка номера интерфейса Ethernet к контроллеру
@ethernet.show	Получение информации об Ethernet
@wifi.conf	Конфигурирование WiFi
@wifi.bind	Привязка номера интерфейса WiFi к контроллеру
@wifi.show	Получение информации о WiFi
@bond.conf	Конфигурирование bonding
@bridge.conf	Конфигурирование моста
@bridge.show	Получение информации о мосте
@dummy.conf	Конфигурирование псевдоинтерфейса
@gre.conf	Конфигурирование gre
@gretap.conf	Конфигурирование gretap
@hdlc.conf	Конфигурирование hdlc (для E1)
@l2tp. server, conf	Конфигурирование 1-2TP-сервера
@l2tp. server, show	Получение информации о 1_2TP-сервере
@l2tp.client.conf	Конфигурирование 1_2TP-клиента
@l2tp. client, show	Получение информации о 1_2TP-клиенте
@pptp.server, conf	Конфигурирование PPTP-сервера
@pptp.server, show	Получение информации о PPTP-сервере
@pptp.client.conf	Конфигурирование PPTP клиента
@pptp.client.show	Получение информации о PPTP-клиенте
@ovpn. server, conf	Конфигурирование OpenVPN-сервера

Имена полномочий	Описание
@ovpn. server, show	Получение информации об OpenVPN-сервере
@ovpn.client.conf	Конфигурирование OpenVPN-клиента
@ovpn.client.show	Получение информации об OpenVPN-клиенте
@ovpn. key.conf	Конфигурирование ключей OpenVPN
@ovpn.key.show	Получение информации о ключах OpenVPN
@pep.conf	Конфигурирование pep
@pep.show	Получение информации о pep

7.2.3 Управление траффиком

Имена полномочий	Описание
@acl.conf	Конфигурирование списков доступа
@acl.oper	Действия (кроме настройки) со списками доступа. Обеспечивает доступ к команде "clear ip recent-list" (очистить списки недавних пакетов, п. 8.3.4)
@acl.show	Получение информации об ACL
@nat.conf	Конфигурирование NAT
@nat.show	Получение информации о NAT
@qos.conf	Конфигурирование QoS
@qos.show	Получение информации о QoS

7.2.4 Сетевые сервисы

Имена полномочий	Описание
@dhcp. server	Настройка серверов DHCP и DHCP-relay
@dhcp.oper	Действия (кроме настройки) с серверами DHCP и DHCP-relay
@dhcp.show	Получение информации о сервере DHCP и DHCP-relay
@dns.server	Настройка сервера доменных имен DNS
@dns.oper	Действия (кроме настройки) с сервером доменных имен DNS
@dns.show	Получение информации о DNS-сервере
@netperf.server	Настройка сервера тестирования netperf и iperf
@netperf.client	Настройка клиента тестирования netperf и iperf
@netperf.show	Получение информации тестирования netperf и iperf
@lldp.server	Настройка сервера LLDP
@lldp.show	Получение информации о сервере LLDP
@ntp.server	Настройка сервера времени NTP
@ntp.show	Получение информации о сервере NTP
@ proxy, server	Настройка прокси-сервера
@proxy.oper	Действия с кешем прокси-сервера
@proxy.show	Получение информации о прокси-сервере
@slagent.server	Настройка сервера slagent

Имена полномочий	Описание
@slagent.show	Получение информации о slagent
@snmp. sever	Настройка сервера SNMP
@snmp.show	Получение информации о сервере SNMP
@ssh.server	Настройка сервера SSH
@ssh.client	Настройка клиента SSH
@ssh.auth	Возможность работать авторизованными ключами
@telnet. server	Настройка сервера telnet
@telnet.client	Настройка клиента telnet
@vrrp.server	Настройка сервера VRRP
@vrrp.show	Получение информации о VRRP
@diweb.server	Настройка через HTTP-протокол
@netflow.conf	Конфигурирование NetFlow
@netflow.oper	Использование NetFlow

7.2.5 Маршрутизация

7.2.7.1 Статическая маршрутизация

Имена полномочий	Описание
@route.conf	Конфигурирование статической маршрутизации
@route.show	Получение информации о статической маршрутизации
@policy-route.conf	Конфигурирование policy route
@policy-route.show	Получение информации о policy route
@mpls.conf	Конфигурирование MPLS
@mpls.show	Получение информации о настройках MPLS

7.2.7.2 Динамическая маршрутизация

Имена полномочий	Описание
@droute-common.conf	Общие команды конфигурирования для различных видов динамической маршрутизации
@droute-common.show	Получение информации об общих структурах данных для динамической маршрутизации
@ospf.conf	Конфигурирование OSPF
@ospf.oper	Действия с OSPF
@ospf.show	Получение информации OSPF
@bgp.conf	Конфигурирование BGP
@bgp.oper	Действия с BGP
@bgp.show	Получение информации BGP
@rip.conf	Конфигурирование RIP
@rip.oper	Действия с RIP
@rip.show	Получение информации RIP
@ldp.conf	Конфигурирование LDP (MPLS)

Имена полномочий	Описание
@ldp.oper	Действия с LDP
@ldp.show	Получение информации LDP (MPLS)

7.2.7.3 Маршрутизация с использованием VRF

Имена полномочий	Описание
@vrf.conf	Команды конфигурирования для различных видов маршрутизации с использованием VRF
@vrf.show	Получение информации о VRF

7.2.7.4 Мультикаст(multicast)-маршрутизация

Имена полномочий	Описание
@mroute.conf	Конфигурирование статической мультикаст-маршрутизации
@mroute.show	Получение информации о мультикаст-маршрутизации
@dvmrp.conf	Конфигурирование мультикаст-маршрутизации DVMRP
@dvmrp.oper	Действия с DVMRP
@igmp.conf	Конфигурирование мультикаст-маршрутизации IGMP
@igmp.oper	Действия с IGMP
@pim.conf	Конфигурирование мультикаст-маршрутизации PIM
@pim.oper	Действия с PIM

7.2.6 Журналирование и трассировка

Имена полномочий	Описание
@log.conf	Конфигурирование системных журналов
@log.oper	Действия (кроме конфигурирования) с системными журналами
@log.show	Получение информации о настройке системных журналов
@log.watcher	Работа со службой watcher
@log.watcher.priv	Работа со службой watcher (привилегированные команды)
@trace.conf	Конфигурирование трассировки
@trace.oper	Действия (кроме конфигурирования) с трассировкой
@trace.show	Получение информации о настройке трассировки
@tcpdump.oper	Действия с анализатором трафика tcpdump
@mailer.conf	Конфигурирование почтового клиента
@mailer.oper	Действия (кроме конфигурирования) с почтовым клиентом
@mailer.show	Получение информации о настройке почтового клиента

7.2.7 Системные операции

Имена полномочий	Описание
@dip.oper	Операции с DIP-пакетами
@dip.show	Получение информации о DIP-пакетах
@data.oper	Действия со слотами данных, командами над пакетами ОС и слотами данных (restore, os bind, schedule rebind, schedule restore и т.д.) (п. 56.2)
@data.show	Получение информации о слотах данных
@backup.oper	Команды создания резервной копии данных (os data backup), безопасной резервной копии (schedule backup) (п. 56.2)
@backup.show	Получение информации о резервных копиях
@boot.oper	Команды загрузки системы boot default, boot fallback, boot experemental, и команда миграции на другой пакет ОС schedule migrate (п. 56.2)
@boot.show	Получение информации о параметрах загрузки
@cluster.conf	Конфигурирование кластера (п. 54)
@cluster.oper	Действия с кластером (кроме конфигурирования)
@cluster.show	Получение информации о кластере
@host.conf	Общие настройки (времени, часового пояса, имени узла) (п. 5.2), принудительной проверки файловой системы на ошибки командой schedule fsck (п. 57.2)
@host.show	Получение информации об общих настройках системы
@schedule.show	Получение информации о действиях, которые должны быть выполнены после перезагрузки
@hw.show	Получение информации об оборудовании
@account.conf	Настройка учетных записей
@account.passwd	Управление паролями учетных записей
@account.show	Получение информации об учетных записях
@account.passwd-hash	Получение информации о хешированном пароле
@role.conf	Настройка ролей
@role.show	Получение информации о ролях
@poweroff.oper	Право выключения системы
@reboot.oper	Право перегружать систему
@conf.write	Право перезаписывать startup-config
@conf.show	Получение информации о startup-config

Имена полномочий	Описание
@watchdog.oper	Право использовать механизм сторожевого таймера для обеспечения перезагрузки удаленной системы (со старым конфигурационным файлом) в случае потери связи из-за ошибочного изменения настроек на этой удаленной системе
@file.oper	Право выполнения файловых операций
@file. net	Право выполнения сетевых операций с файлами
@removable.oper	Право выполнения операций с внешними устройствами (дисками)
@birq.conf	Право на конфигурирование балансировки прерываний

7.2.9.1 DiPool

Имена полномочий	Описание
@dipool. server	Пользовательский DiPool
@dipool.client.conf	Конфигурирование подключения к удаленному DiPool-сервера
@dipool. client, oper	Право получения образов с удаленного DiPool-сервера
@di pool, client.show	Право получения информации с удаленного DiPool-сервера

7.3 Команды управления полномочиями и ролями системы для учетных записей

Расширить права доступа учетной записи добавлением ей полномочий или ролей можно следующими командами:

```
DionisNX(account—ivanov)# delegate @ospf.conf
DionisNX(account—ivanov)# delegate myrole
```

Первая команда в примере добавляет учетной записи ivanov полномочия "@ospf.conf", вторая добавляет роль с именем "myrole". Если учетная запись имела полномочия какой-либо роли, и в ходе дальнейшей работы полномочия этой роли были изменены, то и учетная запись изменит свои полномочия. Однако это изменение произойдет только после завершения текущей сессии и открытия новой сессии для этой учетной записи.

Отменить определенные права доступа учетной записи можно командами:

```
DionisNX(account—ivanov)# no delegate @ospf.conf
DionisNX(account—ivanov)# no delegate myrole
```

Первая команда удаляет полномочия "@ospf.conf" из списка назначенных полномочий администратора "ivanov". Вторая команда удаляет роль "myrole" из списка назначенных данному администратору ролей администратора.

Существует команда для добавления администратору всех полномочий сразу.

DionisNX(account—ivanov)# delegate *

Добавление всех существующих полномочий может понадобиться в случае, когда необходимо создать администратора, обладающего большинством полномочий, за исключением небольшого списка отдельных полномочий. Тогда администратору добавляются все существующие полномочия, а затем из списка доступных полномочий исключаются те полномочия, которые не будут доступны данному администратору. При добавлении всех полномочий следует учитывать, что общий набор полномочий может быть изменен в случае обновления системы до новых версий (эти новые версии могут включать в себя новые, не существующие в данной версии, полномочия). При необходимости добавления учетной записи вновь появившихся полномочий следует либо добавить их в явном виде, либо снова выполнить команду "delegate

7.4 Управление ролями

Для перехода к конфигурированию роли следует выполнить команду (в режиме enable):

DionisNX# role <имя роли>

Если параметр соответствует существующей в системе роли, то произойдет переход к конфигурированию этой роли. Если же параметр не соответствует существующей в системе роли, то в системе будет создана новая роль с указанным именем, которая не будет иметь никаких полномочий. После того, как будет произведен вход в режим конфигурирования роли, могут быть даны команды добавления/удаления полномочий для этой роли, например:

DionisNX(myrole)# delegate @ospf.conf
DionisNX(myrole)# no delegate @ospf.conf

Первая из этих команд добавляет полномочия "@ospf.conf" роли myrole. Вторая из этих команд исключает полномочия "@ospf.conf" из роли myrole. В качестве параметров в команде delegate при конфигурировании роли могут быть указаны не полномочия, а уже существующие роли. Например, при выполнении команды:

DionisNX(myrole)# delegate testrole

роль myrole получит все полномочия роли testrole, причем связь между ролями сохранится. Это означает, что если для роли testrole полномочия будут расширены, то и роль myrole получит новые полномочия роли testrole.

Кроме управления полномочиями роли, в системе существует несколько команд управления ролью. В режиме конфигурирования роли можно добавить любое количество полномочий.

Команда:

DionisNX# no role <имя роли>

удаляет роль с именем. Если в качестве параметра задано значение то будут удалены все роли системы.

Команда:

DionisNX# show role <имя роли>

показывает указанную роль, если она есть. Если в качестве параметра задано значение "*", то будут показаны все роли системы.

Команда:

DionisNX# clone role <имя роли—источи и ка> <имя роли —получателя>

копирует роль-источник в роль-получатель, т.е. создает копию роли-источника с указанием имени новой роли.

7.5 Отображение полномочий и зависимости полномочий

В системе часто имеются отдельные полномочия на конфигурирование какой-либо подсистемы, на использование (в режиме enable) этой подсистемы и получение информации о подсистеме. Целесообразно, чтобы учетная запись, имеющая полномочия на конфигурирование подсистемы, по умолчанию имела права и на использование и получение информации о подсистеме. Таким образом, в системе появляется зависимости между полномочиями — например, зависимости между полномочиями conf, opeг и show для подсистемы. Набор таких зависимостей определен в системе по умолчанию. В системе существуют команды для отмены таких зависимостей или для создания новых зависимостей, а также команды возврата всех зависимостей в состояние, заданное в системе по умолчанию.

Как правило, зависимости полномочий, заданные по умолчанию, в наибольшей степени обеспечивают удобство работы администратора. Изменять зависимости для полномочий администратору следует с осторожностью и только в случае, если он точно понимает, с какой целью он меняет зависимости, заданные по умолчанию.

Команда:

DionisNX# show capability <имя полномочия>

Показывает указанные полномочия и те полномочия, которые включаются в данные полномочия (зависят от них). Например, по умолчанию полномочия @log.conf включают полномочия @1од.opeг и @log.show. После выполнения команды:

DionisNX# show capability @log.conf

Будут отображены эти полномочия, а вслед за ним, отдельной строкой, зависимые полномочия (т.е. @1од.opeг и @log.show). Если в качестве параметра команды "show capability" задано значение то будут показаны все полномочия системы и их зависимости. KoMaHfla"show capaЫ1Ку"может использоваться с параметром "delegate". В этом случае будут показаны только те полномочия, у которых есть зависимости. Например, команда:

DionisNX# show capability * delegate

отобразит список всех полномочий системы, имеющих зависимости, и сами эти зависимости.

Команда:

DionisNX# capability <имя полномочия>

позволяет перейти в режим конфигурирования зависимостей полномочий. Сами полномочия, зависимые от конфигурируемых полномочий, добавляются командами "delegate". Зависимости удаляются командами "no delegate". Например, последовательность команд:

```
DionisNX# capability @log.conf
DionisNX(log.conf)# no delegate @log.oper
```

удалит зависимость полномочий @log.oper от полномочий @log.conf.

Команда

```
DionisNX# clear capability <имя полномочий>
```

Возвращает состояние всех зависимостей для полномочий в состояние, принятое в системе по умолчанию. Если в качестве параметра задано значение то зависимости всех полномочий возвращаются в состояние, принятое в системе по умолчанию.

Команда

```
DionisNX# no capability <имя полномочий>
```

удаляет все существующие зависимости для полномочий . Если в качестве параметра задано значение то удаляются зависимости всех полномочий.

8. Фильтрация и модификация

Подсистема фильтрации является базовым средством обеспечения безопасности сети и позволяет управлять прохождением трафика через интерфейсы маршрутизатора, разрешая или запрещая передачу пакетов, удовлетворяющих указанным правилам отбора. Правила отбора объединяются в IP-списки контроля доступа (ip access-list). Списки контроля доступа могут быть применены к конкретным интерфейсам (с учетом направления трафика), а также к маршрутизатору в целом (с учетом логики маршрутизации).

Ниже приведена упрощенная схема движения пакета через фильтры Dionis-NX:

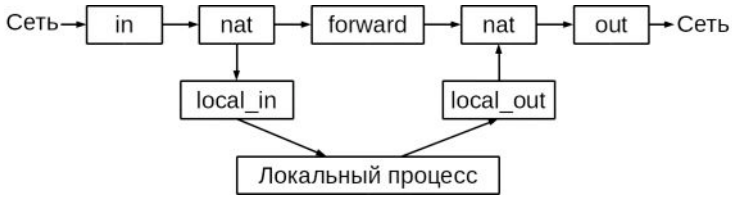

Рис. 8.1: Фильтрация

Название	Назначение
in	Входной фильтр интерфейса
nat	преобразование SNAT или DNAT
local_in	Внутренний фильтр для пакетов, направленных в систему
forward	Фильтр маршрутизации транзитных пакетов
local_out	Внутренний фильтр для сгенерированных в данной системе пакетов
out	Выходной фильтр интерфейса

8.1 Создание ip access-list

Для создания списка контроля доступа, в режиме configure необходимо выполнить команду ip access-list <acl_name>, где <acl_name> - это имя создаваемого списка, например:

```
DionisNX(config)# ip access-list mylist
```

После выполнения команды, задаются (или модифицируются) правила фильтрации этого списка. Каждое правило содержит критерии отбора трафика и может быть разрешающим (permit) или запрещающим (deny). Все правила в списке выполняются последовательно, до первого совпадения критериям отбора. Если ни одно из правил не удовлетворяет критериям, считается, что выполняется разрешающее правило. Например:

```
DionisNX(config—acl—mylist)# permit icmp
DionisNX(config—acl—mylist)# permit tcp
DionisNX(config—acl—mylist)# permit udp
DionisNX(config—acl—mylist)# deny
```

В данном примере, разрешается прохождение пакетов трех протоколов (icmp/udp/tcp), а все остальные протоколы запрещаются.

Для того, чтобы просмотреть правила отбора текущего редактируемого списка, следует выполнить команду:

DionisNX(config—acl — mylist)# do show

При этом будут выведены все правила текущего списка. Каждая строка снабжена числовым префиксом, указывающим позицию правила в списке.

Для того, чтобы удалить правило с конкретным номером, введите команду: по сномер правила> Например:

DionisNX(config—acl —mylist)# no 1

Для того, чтобы вставить новое правило в конкретную позицию, введите команду: сномер правила> <правило> Например:

DionisNX(config—acl —mylist)# 1 permit src 192.168.0.0/24
DionisNX(config—acl — mylist)# do show
1 permit src 192.168.0.0/24
2 permit tcp
3 permit udp
4 deny

Удаление всего содержимого текущего списка может быть осуществлено командой: no all. Для удаления списка, используется команда: no ip access-list симя списка>.

Для просмотра информации о списках, существует две команды, доступные из enable режима.

show ip access-list <acl_name|*> config	Информация о действующей конфигурации
show ip access-list <acl_name|*>	Низкоуровневая информация из ядра ОС

Если в командах имя списка задано как *, будет показана информация о всех списках.
Например (из режима configure):

DionisNX(config)# do show ip access—list mylist config

8.2 Привязка ip access-list

Создание списка контроля доступа не означает, что список начинает действовать. Для того чтобы начала действовать фильтрация в соответствии с правилами списка, список должен быть привязан к интерфейсу и/или определенной цепочке в логике маршрутизации. Один и тот же список может быть привязан к нескольким интерфейсам/цепочкам.

8.2.1 Привязка к интерфейсу

Для того, чтобы привязать список к интерфейсу, нужно войти в режим конфигурации интерфейса и выполнить команду (или команды) ip access-group <имя списка> <направление>

Под параметром <направление> понимается направление трафика относительно интерфейса. Входящий трафик обозначается как in, а выходящий как out. Например:

```
DionisNX(config)# interface ethernet 0
DionisNX(config — if—ethernet0)# ip access—group mylist in
DionisNX(config —if—ethernet0)# ip access—group mylist out
```

Для удаления связи с интерфейсом, необходимо выполнить команду: no ip access-group <имя> <направление>, например:

```
DionisNX(config — if—ethernet0)# no ip access—group mylist out
DionisNX(config — if—ethernet0)# do show
ip access—group mylist in
```

Иногда возникает необходимость фильтровать защищенный трафик (туннели DISEC). Фильтрация такого рода трафика означает, что правила access-list должны применяться на пакеты, которые уже были расшифрованы после приема их на интерфейсе, или еще не были зашифрованы, при их отправке через интерфейс. В этом случае для привязки фильтров применяется команда: ip access-group-xfrm, синтаксис которой аналогичен ip access-group. Для удаления связи с интерфейсом, следует использовать команду no ip access-group-xfrm.

```
DionisNX(config)# interface ethernet 0
DionisNX(config —if—ethernetO)# ip access-group-xfrm mylist in
DionisNX(config —if—ethernetO)# ip access-group-xfrm mylist out
DionisNX(config —if—ethernetO)# no ip access—group—xfrm mylist out
DionisNX(config —if—ethernetO)# do show
ip access—group—xfrm mylist in
```

8.2.2 Привязка к цепочке обработки

Существует возможность привязать фильтр к внутренним цепочкам маршрутизатора. Для этого в режиме configure достаточно выполнить команды: ip access-group <имя><цепочка>.

Где цепочка может принимать значения: local-in, local-out, forward, что соответствует цепочкам прохождения пакета в Dionis-NX.

Например:

```
DionisNX(config)# ip access—list mylist forward
```

8.3 Правила отбора

Выше были приведены примеры с очень простыми правилами отбора, фактически, единственным критерием задавался протокол датаграммы, или критерия не было вообще (deny). Списки

контроля доступа могут содержать правила с комбинацией различных критериев отбора, которые объединяются в логическое «и», что делает фильтры простым и мощным инструментом по обеспечению безопасности в сети. Если какой-то из критериев не задан (например, не задан протокол датаграммы), то данному правилу отбора будут удовлетворять пакеты с любым значением критерия (любые протоколы).

Полный список критериев находится в Полном списке команд Dionis-NX, далее будут описаны основные критерии.

8.3.1 Адреса источника и назначения

Для задания адресов источника используется параметр src после которого указывается сетевой адрес. Например:

permit tcp src 192.168.16.0/24

Адрес, содержащий маску, определяет сеть. Адрес с маской 32 или без задания маски определяет адрес хоста. Если параметр src не задан, то правило отбора распространяется на датаграммы с любым адресом источника.

Аналогично параметру источника, существует параметр адреса приемника dst, с таким же синтаксисом:

permit tcp src 192.168.16.0/24 dst 10.0.0.1
deny

8.3.2 Порты источника и назначения

Если в критериях правила отбора указан протокол tcp или udp, то появляется возможность задать критерий отбора по портам источника и/или назначения. Для задания портов источника используется параметр: sport <portl> [port2]. Может быть задан как единственный порт <portl>, так и диапазон портов <portl> <port2>. например:

deny tcp sport 1 1000

Для задания портов приемника используется параметр dport, с аналогичным синтаксисом,

deny tcp dport 22

8.3.3 Содержимое пакета

Существует возможность задания критерия по содержимому пакета. Для этого используется параметр content <правило>. Где <правило> записывается на специальном языке, и позволяет отслеживать содержимое байт/слов в пакете.

Все позиции байтов считаются с 0. В простейшем случае, правило выбирает 4 байта по заданному смещению (start), применяет маску(mask) и сравнивает результат с попаданием в диапазон

(range). В таком случае правило принимает вид: start&mask= range. Где range задается как диапазон (a:b) или значение (a).

Обычно нужно взять значение start меньшим на 3, чем позиция последнего байта, который нам нужен. Так, если нужны байты 4 и 5 из заголовка IP (поле ID), то start должен быть 5-3 = 2. mask убирает те байты, которые не нужны в критерии. Максимальная маска (которая означает анализ всех 4-х байт) может быть равной Oxffffffff. Так, чтобы взять байты 4 и 5, отбросим байты 2 и 3, что соответствует маске 0x0000ffff. Таким образом правило будет выглядеть так:

permit content 2&0xFFFF=0x2:0x0100

Пример критерия, который отбирает пакеты длиной, большей 256. Общая длина пакета лежит в байтах 2 и 3 IP датаграммы. Тогда стартовая позиция 3-3 = 0. Отбрасывая 2 байта получаем:

deny content 0&0xFFFF=0xl00:0xFFFF

Пример критерия, основанного на выборке одного байта. Выберем байт TTL (смещение 8). 8 -3 = 5, маска Oxff:

deny content 5&0xFF=0:3

Пример критерия, основанного на выборке четырех байт. Выберем IP-адрес назначения (байты 16-19). Маска не требуется:

permit content "16=0xE0000001" remark "224.0.0.1/32"

Если необходимо выбрать первые три байта (например, проверить сетевой адрес сети класса C), то необходима маска OxffffffOO.

deny content "12&0xFFFFFF00=0xC0A80F00" remark "src 192.168.15.0/24"

Чтобы посмотреть на поле TOS (1 байт заголовка IP), невозможно начать с байта 1-3 = -2. Нужно начать с байта 0, отбросить лишние байты и сдвинуть нужный байт в последнюю позицию. Для сдвига используются команды « и ».

permit content "0&0x00FF0000>> 16=0x08" remark "tos 8"

Пример анализа флага MF.

deny content "3&0x20>>5 = 1"

Анализ TCP-заголовка. Допустим, необходимо анализировать байты 4-7 заголовка TCP (sequence number). Для простоты, будем считать, что заголовок IP занимает 20 байт. Тогда критерий будет выглядеть, например, так:

permit content 24=0x29

Однако, в этом примере не анализируется протокол. Добавим проверку протокола (логическое «и»):

permit content "6&0xFF=0x6 && 24=0x29"

Но размер IP-заголовка не всегда равен 20 байтам, поэтому необходимо еще доработать правило отбора. Специальный оператор @ позволяет использовать значение последнего выражение как стартовую позицию нового.

Для того, чтобы взять размер заголовка, необходим байт 0>>24, но нужны только 4 бита, умноженные на 4, то есть: 0>>22&0x3C. Теперь необходимо, чтобы это выражение стало новым смещением. Для этого и нужен оператор @:

permit content "6&0xFF=0x6 && 0>>22&0x3C@4=0x29"

Еще один, более сложный, пример. Проверка на TCP, проверка на первый фрагмент или нефрагментированный пакет, проверка, что 4-7 байты TCP заголовка равны 41:

permit content "6&0xFF=0x6 && 4&0x1FFF=0 && 0>>22&0x3C@4=0x29"

Пример проверки соответствия пакета сообщению ICMP Host Unreachables (ICMP, type 3, code 1)

permit content "6&0xFF=1 && 4&0x1FFF=0 *&&* 0>>22&0x3C@0>> 16=0x0301"

8.3.4 Списки недавних пакетов

В Dionis-NX существует возможность создавать правила относительно накопленной информации о предшествующих пакетах, для этого используется критерий recent.

Для работы с критерием recent, вводится понятие списка недавних пакетов (ip recent-list). Каждый такой список идентифицируется по имени и заполняется динамически с помощью критерия recent <имя> set, например:

DionisNX(config—acl —ping)# permit icmp recent pings set

В данном случае, пакеты с протоколом icmp будут допущены к маршрутизации, при этом информация о пакетах (исходный адрес датаграммы и время) будет заноситься в список pings. Каждый список может сохранять до 1024 записей (адресов), в каждой записи может храниться информация о 20 последних пакетов.

Критерий recent может быть использован для проверки наличия записи в списке недавних пакетов. Для этого используется критерий: recent <имя> update|check.

Так, в следующем примере:

DionisNX(config—acl — ping)# permit tcp dport ssh recent pings update
DionisNX(config—acl —ping)# drop tcp dport ssh

Пакет, направляемый на порт службы ssh будет пропущен только в том случае, если ранее от хоста источника были получены icmp-пакеты.

Полный синтаксис критерия recent:

recent <имя списка> <операция> [аргументы]

Операции:

Название	Назначение
set	Добавить информацию о пакете в список
remove	Удалить информацию о пакете в список
check	Проверить информацию о пакете в списке
update	Проверить информацию о пакете в списке и обновить временную метку

Возможные аргументы для операции set:

Название	Назначение
dest	Запоминать адрес назначения, а не адрес источника

Возможные аргументы для операции remove:

Название	Назначение
dest	Работать относительно адреса назначения, а не адреса источника

Возможные аргументы для операций check и update:

Название	Назначение
dest	Проверять адрес назначения, а не адрес источника
seconds <число>	Проверять запись не старее заданного числа секунд
hitcount <число>	Проверять запись, число пакетов в которой больше или равно заданного числа
rttl	Проверять корректность ttl (соответствие ttl пакета с информацией в записи из списка)

Для работы с списками ip recent-list используются следующие команды в режиме enable:

Команда	Параметры	Назначение
show ip recent list	<* или имя списка> [IP адрес]	Просмотреть информацию о записях в recent списках
clear ip recent list	<* или имя списка> [IP адрес]	Очистить информацию о записях в recent списках

8.3.5 Состояние соединения

Ядро Dionis-NX содержит средства, обеспечивающие отслеживание состояния соединений и классификацию пакетов с точки зрения принадлежности к соединениям, что позволяет осуществлять полноценную фильтрацию трафика.

При этом, ядром поддерживаются следующие функции:

1. Отслеживание состояний отдельных соединений с тем, чтобы классифицировать каждый пакет либо как относящийся к уже установленному соединению, либо как открывающий новое соединение. При этом понятие «состояние соединения» искусственно вводится для протоколов, в которых оно изначально отсутствует (UDP, ICMP). При работе же с протоколами, поддерживающими состояния (например, TCP), активно используется эта возможность.
2. Отслеживание связанных соединений, например, ICMP-ответов на TCP- и UDP-пакеты.

В правилах отбора следует использовать критерий state для того, чтобы использовать информацию о состоянии соединения. При этом, можно указывать 4 состояния.

Название	Смысл
invalid	Пакет связан с неизвестным потоком или соединением и, возможно, содержит ошибку в данных или в заголовке
established	Состояние указывает на то, что пакет принадлежит уже установленному соединению, через которое пакеты идут в обоих направлениях
new	Пакет открывает новое соединение или пакет принадлежит однонаправленному потоку.
related	Пакет принадлежит уже существующему соединению, но при этом он открывает новое соединение.

В качестве примера, рассмотрим правила фильтрации для внешнего сетевого интерфейса, разрешающие только исходящие соединения (соединения из внутренней сети во внешнюю сеть).

```
DionisNX(config)# ip access—list wan
DionisNX(config—acl—wan)# permit state established
DionisNX(config—acl—wan)# permit state related
DionisNX(config—acl—wan)# deny
```

8.3.6 Другие критерии отбора

Синтаксис других критериев отбора описан в списке команд Dionis-NX, ниже приводятся некоторые из них.

Название параметра	Критерий
connlimit	Ограничение числа соединений с одного клиентами из сети)
syn	syn флаг в TCP-пакете
mac	MAC-адрес источника
tos	Значение TOS
dscp	Значение DSCP
datestart,datestop,timestart,timestop,monthdays	,\Bpetecl0ys

8.3.7 Протоколирование

Существует возможность протоколирования факта выполнения выбранных правил фильтрации. Для этого необходимо указать параметр: log [all]. Необязательный параметр all указывает на необходимость протоколирования полного тела пакета, а не только заголовка.

```
deny tcp dport 22 log
```

Настройка подсистемы протоколирования и выборка из журнала рассмотрены в соответствующей главе.

8.3.8 Комментарии

Администратор может комментировать отдельные правила отбора с помощью параметра remark, например:

deny tcp dport 22 log remark "Stop port scanning"

8.4 Другие правила списков контроля доступа

Кроме запрета и разрешения трафика (permit/deny), правила в списках могут содержать следующие действия:

Название правила	Параметры	Действие
call	access-list	Передать управление на другой access-list с возвратом
goto	access-list	Передать управление на другой access-list без возврата
log	правила отбора	Протоколировать пакет, не выполняя запрета/разрешения
clone	IP-адрес шлюза	Клонировать пакет и отправить копию в шлюз

8.5 Модификация

Для модификации пакетов используются списки модификации (ip mangle-list). По синтаксису они похожи на списки контроля доступа, однако позволяют отбор пакетов по классу (ip class-map, см. руководство по QoS). Однако при изменении классов возможен небольшой период (несколько секунд), когда классы определяются неверно, поэтому отбор пакетов по классу не реализован в подсистеме фильтрации, связанной с безопасностью.

Для модификации пакетов используется команда mangle в секции ip mangle-list. Возможные операции:

Операция	действие
adjust-mss	Поменять поле mss в пакете SYN
set-tos/set-dscp	Установить поле TOS/DSCP пакета
set-df/clear-df	Установить поле don't fragment IP-пакета
set/inc/dec-ttl	Изменить поле TTL
set-cos	Установить поле 802.lq CoS

Для отбора пакетов по классу используется параметр class, затем перечисляются классы через запятую. Восклицательный знак перед классом означает отрицание класса. Пример:

```
mangle inc—ttl 1 class cla,!clb
```

Увеличит ttl на 1 для пакетов, попавших в класс cla и одновременно не попавших в класс clb.

8.6 IPv6

Для создания и модификации списков доступа/модификации IPv6 используются команды, аналогичные командам IPv4, с префиксом ip6 вместо ip. Например:

```
ip6 access—list acl
1 deny src 2001:db8::21
!
interface ethernet 0
ip6 access—group acl
!
```

8.7 Группирование IP адресов

Зачастую необходимо применить одно и то же правило для достаточно большого набора IP адресов. При этом создание отдельного правила для каждой подсети имеет недостаток: слишком много времени затрачивается на прохождение пакета через эти правила. Команда ip(6) set позволяет создать список подсетей, принадлежность которому проверяется при помощи одного правила ip(6) access/mangle/trace/nat-list, class-map, или policy-route.

ip set состоит из правил match и exclude; единственный аргумент правила — подсеть. Если маска подсети не указана, подразумевается подсеть из одного ip адреса. Если одному ip адресу соответствует несколько правил, из них будет действовать правило с большей длиной префикса подсети (т.е. с меньшей подсетью).

Пример:

```
ip set ipset
  match 192.168.0.0/16
exclude 192.168.33.0/24
match 192.168.33.25
!
ip access—list acl
deny src—ipset ipset
```

9. Многоадресная передача

Dionis-NX может использоваться для организации многоадресной передачи (далее МП) на уровне IP-стека TCP/IP.

МП используется в тех случаях, когда по сети необходимо передавать нескольким пользователям одну и ту же информацию. Такая потребность может возникнуть, например, при распространении через IP-сеть телевизионного или радиосигнала, при организации телеконференций или селекторных совещаний. В этом случае можно не открывать отдельные соединения с каждым клиентом сети, с тем чтобы передавать по ним одинаковую информацию, а рассылать IP-пакеты без лишнего дублирования, но так, чтобы их получали все клиенты.

В Dionis-NX можно настраивать динамическую и статическую маршрутизацию.

В данном подразделе МП будет рассматриваться на основе динамической многоадресной маршрутизации(далее ММ).

Основные понятия МП:

- передача пакетов: пакеты МП рассылаются не отдельным узлам, а группе узлов;
- адресация группы: группа определяется одним групповым IP-адресом класса D в диапазоне 224.0.0.0-239.255.255.255:
  - 224.0.0.0/8 - зарезервированные IANA-адреса;
  - 232.0.0.0/8 - глобальное адресное пространство для МП, специфичной для конкретного источника (SSM, реализован в IGMPv3);
  - 233.0.0.0/8 - глобальное адресное пространство GLOP для групп внутри автономных систем (232.AA.AA.GG, где AAAA - 16 бит, включающих номер AS, GG номер группы в AS);
  - 239.0.0.0/8 - локальное адресное пространство для закрытых (частных) сетей; аналог LAN одноадресного пространства адресов;
- адрес источника не может быть адресом класса D;
- членство в группе: узлы сети могут входить в группу МП (далее группу) и выходить из нее по своему желанию;
- адресность: многоадресные датаграммы (далее МД) посылаются группе и только члены этой группы получают их.

Организация МП в Dionis-NX осуществляется тремя протоколами:

- управление группами при помощи протокола IGMPv3 (Протокол управления группами Интернет, версия 3);
- маршрутизация МД при помощи протокола DVMRP (Дистанционно-векторный протокол многоадресной маршрутизации);
- маршрутизация МД при помощи протокола PIM-SM (Независимая от протокола многоадресная передача (Разреженный режим)).

Рассмотрим вкратце механизм МП:

- исходный сетевой узел (например узел, транслирующий видеофильм) посылает МД на групповой адрес А (например по UDP- или RTP-протоколу);
- сетевые узлы назначения (клиенты, желающие смотреть данный видеофильм) сообщают маршрутизатору по протоколу IGMP о желании присоединиться к группе А;
- маршрутизатор при получении МД определяет, нужно ли ее пересылать дальше (протокол DVMRP или PIM);
- если МД нужно пересылать дальше, маршрутизатор определяет, на какой именно интерфейс ее послать, чтобы она быстрее достигла получателей (протокол DVMRP или PIM) из группы А.

Основные особенности протокола IGMP:

- служит для обмена информацией о членстве в группах между IP-маршрутизаторами, поддерживающими МП, и членами групп;
- узлы сами сообщают маршрутизаторам о своем членстве в группах (IGMP-сообщение REPORT);
- узлы сами сообщают маршрутизаторам о выходе из группы (IGMP-сообщение LEAVE);
- состояние членства узлов в группах периодически проверяется маршрутизаторами, поддерживающими МП (IGMP-сообщение QUERY)

Основные особенности протокола DVMRP:

- относится к внутренним протоколам маршрутизации, пригодным для использования в пределах автономной системы;
- обеспечивает эффективный механизм доставки МД хостам, входящим в группы, без организации соединений;
- использует сообщения протокола IGMP для обмена информацией с другими маршрутизаторами, поддерживающими МП.

Эффективность маршрутизации в протоколе DVMRP осуществляется путем использования алгоритма RPM (Reverse Path Multicasting): [1]

1 динамически генерирует деревья групповой доставки МД;
- если в зоне ответственности маршрутизатора нет членов группы, тогда маршрутизатор отсекает ненужные ветки дерева рассылки (pruning);
- сохраняет информации о пути возврата к отправителю МД (передача маршрутов).

Основные особенности протокола PIM-SM:

- используется для сетей с произвольным рассредоточением пользователей с ограниченной пропускной способностью сетевых каналов;
- эффективная поддержка работы «рассеянных» мультикастинг-групп: группы из разных автономных систем, находящихся на разных континентах;
- построение дерева маршрутов, разветвляющегося как можно ближе к получателям МД;
- передача трафика идет только по явному запросу.

9.1 Общие сведения о настройке многоадресной маршрутизации

В Dionis-NX имеется возможность настраивать:

- статическую ММ: командой ip mroute;
- динамическую ММ на основе IGMPv2: командой router igmp;
- динамическую ММ на основе DVMRP (с поддержкой IGMPv3): командой router dvmrp;
- динамическую ММ на основе PIM-SM (с поддержкой IGMPv3): командой router pirn.

Одновременно в системе может быть только одна из четырех типов настроек.

9.2 Настройка протокола DVMRP

Данная настройка включает также поддержку протокола IGMPv3, нужного для работы DVMRP.

Включение ММ на основе DVMRP осуществляется командой:

(config)# router dvmrp

Если данная команда успешно выполнилась, по умолчанию все доступные для МП интрефейсы НЕ будут участвовать в МП.

Чтобы интерфейс участвовал в МП необходимо выполнить команду

(config—dvmrp)# iface ethernet O

Это включит интерфейс ethernetO в участие в МП и ММ.

9.2.1 Настройка параметров интерфейсов

Рассмотрим настройки на примере интерфейса ethernet0. Для настройки параметров интерфейса сначала нужно войти в конфигурацию соответствующего интерфейса:

(config)# router dvmrp (config—
dvmrp)# iface ethernet0

Основные параметры МП, которые могут быть настроены для интерфейсов:

- метрика: задает стоимость прохождения МД через данный интерфейс;
- порог TTL: минимальное значение 1PTT1_для МД, нужное для прохода этой МД через данный интерфейс;
- пропускная способность многоадресного трафика.

Чтобы настроить метрику для интерфейса, следует выполнить команду:

(config—dvmrp—ethernet0)# metric 1

Для метрики следует устанавливать как можно меньшее значение, т.к. максимально сумма всех метрик маршрута МД в сети не может превышать 31. По умолчанию: 1.

Чтобы настроить порог TTL для интерфейса, следует выполнить команду:

(config—dvmrp—ethernet0) # threshold 5

По умолчанию: 1.

Если интерфейс получает МД из разных удаленных подсетей и необходимо,чтобы интерфейс обслуживал МП для этих подсетей, следует описать эти подсети следующей командой:

(config—dvmrp—ethernet0)# subnet 10.0.0.0/24

Полностью запретить ММ на интерфейсе можно с помощью следующей команды:

(config—dvmrp—ethernet0)# disable

9.2.2 Настройка ограничений

Обычно настройки порога TTL, приведенные в предыдущем подразделе, используются для достижения следующих целей:

- ограничить время жизни МД;
- уменьшить трафик из-за ограничений пропускной способности сети;
- уменьшить трафик для целей повторного использование адресов и приватности.

Для третьей цели лучше подходит использование не ограничения по полю TTL, а назначение определённого группового адреса как административной границы (далее АГ). Интерфейс, которому назначена АГ, не будет принимать и передавать МД, направленные адресам, принадлежащими АГ.

АГ определяется следующим интервалом групповых адресов: 239.0.0.0-239.255.255.255. Эти адреса могут быть использованы и назначены только внутри автономных систем, где гарантируется их уникальность.

Рассмотрим пример назначения АГ интерфейсу:

(config—dvmrp)# boundary bol 239.255.1.0/24
(config—dvmrp—ethernetO)# bound bol (config—
dvmrp—ethernetO)# bound 239.255.2.0/24

Эти команды определяют для интерфейса ethernetO две АГ 239.255.1.0/24 и 239.255.2.0/24. Любой трафик МП с адресами назначения, соответствующим указанным границам, не разрешён к передаче через данный интерфейс в обоих направлениях.

9.2.3 Настройка протоколирования

Для включения режима протоколирования динамической ММ следует выполнить команду:

(config—dvmrp)# log <TYPE>

Параметр TYPE задает тип протоколируемой информации и может принимать следующие значения:

- packet : входящие/исходящие пакеты;
- prune : отсечение маршрутов;
- route : маршрутизационные сообщения;
- route-details : более детальная информация о маршрутизации;
- peer : взаимодействие соседей (маршрутизаторов) между собой;
- route-cache : кэширование маршрутов;
- timeout : таймауты;
- interface : виртуальные интерфейсы;
- group : группы;
- mtrace : многоадресный traceroute;
- igmp : IGMP-сообщения;
- icmp : ICMP-сообщения;
- rsrr : RSRR-сообщения;
- default : igmp,route,route-cache,prune,peer,interface,group информация;
- all : все перечисленные типы информации.

9.3 Настройка протокола PIM

Далее под протоколом PIM будем понимать этот протокол в режиме SM (Sparse mode). Данная настройка включает также поддержку протокола IGMPv3, нужного для работы PIM.

Рассмотрим основные понятия протокола: [2]

---

2 режим SM протокола PIM - используется для сетей с произвольным рассредоточением пользователей с ограниченной пропускной способностью сетевых каналов;

- PIM-маршрутизатор - маршрутизатор, например Dionis-NX, поддерживающий протокол PIM;
- PIM-домен - набор смежных PIM-маршрутизаторов, сконфигурированных для совместной работы в рамках границ, определенных пограничными маршрутизаторами PMBR, соединяющими PIM-домен с остальным Интернет;
- PMBR-маршрутизатор - пограничный PIM-маршрутизатор; размещается на границе PIM-домена и взаимодействует с другими типами мультикаст-маршрутизаторов;
- точка встречи (RP) - PIM-маршрутизатор разветвления маршрута для потока данных; каждая мультикаст-группа должна иметь RP; RP выбираются динамически, либо назначаются статически; отправители используют RP для объявления о своем существовании, а получатели, чтобы узнать о новых отправителях путем посылки Join PIM-сообщений;

- дерево кратчайших маршрутов (SPT) - описывает кратчайший путь от RP к источнику МД; обозначается как (S,G) - для каждой пары источник(Б)-группа(С) строится свое SPT;
- дерево точки встречи (RPT): дерево кратчайших маршрутов от RP к получателям МД; обозначается как (*,G) - т.к. строится вне зависимости от адреса источника S, а только в зависимости от группы G;
- RPF - Маршрутизация Обратного Пути: МД пересылается на все интерфейсы, кроме того, с которого пришел, только если источник МД доступен через интерфейс получения данной МД (есть маршрут к источнику через интерфейс получения);
- выделенный маршрутизатор (DR) выбирается (по приоритету и затем по максимальному IP-адресу) из маршрутизаторов,подсоединенных к одной и той же сети с МП; он ответственен за посылку сообщений Join,Prune,Register к RP в данном сегменте сети; выбирается для того,чтобы только он передавал МД данной группы в данную сеть,что бы избежать дублирования пакетов;
- NHR - ближайший к получателю P1M-маршрутизатор;
- NHS - ближайший к источнику P1M-маршрутизатор;
- вышестоящий маршрутизатор - маршрутизатор, расположенный ближе к источнику МД;
- нижестоящий маршрутизатор - маршрутизатор, расположенный ближе к получателю МД;
- BSR - маршрутизатор, отвечающий за рассылку bootstrap сообщений; содержит полный список C-RP домена,который рассылается по домену на адрес 224.0.0.13; должен быть хотя бы один BSR, иначе информация о C-RP будет неизвестна маршрутизаторам сети;
- C-RP - кандидат в RP-маршрутизаторы; среди маршрутизаторов,объявленных как C-RP, происходит выбор RP по приоритету и затем по величине IP-адреса;
- C-BSR - кандидат в BSR-маршрутизаторы; среди маршрутизаторов,объявленных как C-BSR, происходит выбор BSR по приоритету и затем по величине IP-адреса;
- основные PIM-сообщения (юникастные):

    - Join - присоединение маршрутизатора к дереву маршрутов; сообщение посылается,если пакет получен на интерфейсе, прошедшем проверку RPF и есть локально присоединенные хосты или нижестоящие маршрутизаторы, желающие получать трафик данной группы;
    - Prune - отсоединение маршрутизатора от дерева маршрутов; сообщение посылается,если пакет получен на интерфейсе, не прошедшем проверку RPF и/или нет локально присоединенных хостов или нижестоящих маршрутизаторов, желающих получать трафик данной группы.
    - Register - сообщение посылается, когда источник отправляет данные группе в первый раз, его DR посылает это сообщение, в которое вкладывает МД источника
    - Register-stop - сообщение от RP к DR, в котором говорится,что не нужно больше инкапсулировать МД в Register-сообщение, т.к.:

        * в случае если есть получатели МД данной группы: на RP уже передается МД от источника по SPT, построенного после получения Register сообщения;
        * нет получателей МД данной группы;

    - Candidate-RP-Advertisement - C-RP периодически высылает в адрес BSR данное сообщение об обслуживаемых группах; BSR собирает эти данные и распространяет их далее по PIM-домену в сообщении bootstrap;
    - bootstrap - сообщения, которые воспринимаются всеми PIM-маршрутизаторами для получения RP-информации (о том, какие RP отвечают за какие группы) и для динамического выбора BSR-маршрутизатора; это многоадресные сообщения на адрес

224.0.0.13 (AII-PIM-routers).

Таким образом:

- каждая мультикаст-группа должна иметь хотя бы один C-RP;
- PIM-SM-домен должен иметь хотя бы один C-BSR, если только все маршрутизаторы домена не имеют статически заданной информации о RP всех доменов;
- каждая подсеть должна иметь хотя бы один DR.

Рассмотрим подробнее алгоритм работы PIM-SM:

Фаза 1. Выбор кратчайшего маршрута.

1. пусть хост посылает IGMP-Join-сообщение Л на DR, который не является RP для указанной в сообщении группы G1;
2. DR отправителя шлет сообщение Л по направлению к RP для группы G1 («upstream»), он определяет это из последнего присланного от BSR bootstrap-сообщения;
3. каждый PIM-маршрутизатор, через который проходит сообщение Л, записывает, что существуют члены группы Л на входящем интерфейсе;
4. в результате Л доходит либо до RP, либо до другого маршрутизатора, за которым есть члены группы;
5. если сообщения сходятся в RP и есть много членов группы G1, то эти сообщения Join формируют RPT, на основе информации от DR получателей, в результате образуются эффективные короткие маршруты пересылки МД к получателям
6. DR отправителя начинет посылку МД, вкладывая МД-данные источника МД в юникаст-пакет PIM Register и посылая данный PIM Register на RP группы;
7. RP группы, получив пакет PIM Register, дэинкапсулирует МД из пакета и посылает его по сформированному на основе RPT маршруту.

Фаза 2. Повышение эффективности и скорости посылки.

1. получив PIM Register, RP выбирает SPT к отправителю, в результате чего МД больше не нужно регистрировать на RP и, как следствие, инкапсулировать в PIM Register;
2. RP отправляет PIM RegisterStop сообщение в ответ на следующее инкапсулированное сообщение от DR отправителя.
3. DR, получив PIM RegisterStop сообщение, прекращает регистрацию/инкапсуляцию МД и посылает нулевое Register-сообщение, которое является вопросом к RP: «Все еще не требуются Register-сообщения?»;
4. если RP отвечает на нулевое Register сообщение сообщением RegisterStop, то DR начинает посылку оригинальных (неинкапсулированных) МД;
5. если DR не получает другого RegisterStop сообщения в течение некоторого периода времени, то DR продолжает посылать Register сообщения.
6. как только RP начинает получать оригинальные МД от источника, RP начинает перенаправлять их по кратчайшему RPT-маршруту к получателям.

Фаза 3. Выбор более оптимального маршрута.

1. когда DR получателя получает МД от отправителя, этот DR отправляет Join сообщение по направлению к отправителю;
2. когда DR отправителя получает указанное Join сообщение, этот DR начинает посылать МД напрямую к получателю;
3. таким образом формируется SPT для множества получателей;
4. когда МД приходят из SPT-маршрута на DR получателя или на общий для SPT и RPT маршрутизатор, то данный маршрутизатор начинает отбрасывать пакеты otRPT и посылает сообщения PIM Prune на RP, чтобы отсечь RPT-маршруты, т.к. уже сформирован путь к получателям в обход RP.

Рассмотрим пример МП посредством PIM-SM (см. рис. 9.1).

Рис. 9.1: Схема МП PIM-SM

1. предположим, что все PIM-маршрутизаторы уже имеют информацию о расположении RP и поддерживаемых ими групп (посредством bootstrap-сообщений или путем статического назначения RP на P1M-маршрутизаторах);
2. SRC - источник начинает трансляцию группы G;
3. NHS - ближайший к источнику PIM-маршрутизатор регистрируется на RP (из п.1 он знает,какая RP отвечает за группу G);
4. RCV - получатель заявляет о желании получать трафик группы G: шлет IGMP сообщение Join (*,G); его получает ближайший к RCV PIM-маршрутизатор NHR;
5. NHR - ближайший к получателю PIM-маршрутизатор шлет PIM-сообщение Join(*,G) в сторону RP;
6. RP шлет PIM-сообщение Join(S,G) в строну NHS;
7. построены деревья: SPT - между SRC и RP ; RPT - между RP и RCV

Включение ММ на основе PIM осуществляется командой:

(config)# router pim

В результате все доступные для МП интерфейсы НЕ будут участвовать в МП.

Чтобы интерфейс участвовал в МП необходимо выполнить команду

(config—pim)# iface ethernet O

Это включит интерфейс ethernetO в участие в МП и ММ.

9.3.1 Глобальные настройки

iface <IFACE>

Добавляет интерфейс IFACE к участию в МП по протоколу PIM и входит в настройки интерфейса по части МП. Если не выполнить данную команду, указанный интерфейс не будет участвовать в МП.

default-route-distance <VAL>

Задает приоритет маршрутизатора в выборе выделенного маршрутизатора (DR). Чем ниже значение ,тем выше приоритет. По умолчанию: 101.

bsr-cand [LOCAL_IP] [PRIO]

Устанавливает параметры C-BSR. Данная система Dionis-NX объявляется как кандидат в BSR.

Параметры:

- LOCAL_IP - один из локальных IP-адресов; по умолчанию: выбирается максимальный IP-адрес;
- PRIO - приоритет C-BSR; указывает насколько важен данный кандидат при выборе; чем ниже значениедем выше приоритет.

По умолчанию: 255

rp-cand [LOCAL_IP] [period TIME] [priority PRIO]

Устанавливает параметры C-RP данной системы. Данная система Dionis-NX объявляется как кандидат в RP.

Параметры:

- LOCAL_IP - один из локальных IP-адресов; по умолчанию: выбирается максимальный IP-адрес;
- PRIO - приоритет C-RP; указывает насколько важен данный кандидат при выборе; чем ниже значениедем выше приоритет;
- TIME - период времени между посылками PIM-сообщения Candidate-RP-Advertisement (сообщает BSR об RP); данное PIM сообщение ,будучи полученным,воспринимается только BSR для обновления знания об RP-узлах и поддерживаемы ими группах.

По умолчанию: приоритет: 0, период: бОсек.

group <IP/MSK>

Задает группу, за которую будет отвечать данная C-RP. Имеет смыслдолько если указана команда rp-cand-команда.

rp-static <IP> <GRP> [PRIO]

Задает статически C-RP и поддерживаемую ей группу.

Параметры: [3]

3 IP - IP-адрес RP;
- GRP - многоадресная группа, которую обслуживает указанная RP;
- PRIO - приоритет C-RP; указывает насколько важен данный кандидат при выборе; чем ниже значениедем выше приоритет.

При использовании rp-static необходимо задать ее на каждом P1M-маршрутизаторе.

По умолчанию: приоритет: 0.

log <all|default>

Включает лог:

- default - лог IGMP- и P1M-сообщений
- all - самый подробный лог.

spt-threshold <RATE> <INTERVAL>

Устанавливает порог перехода с RPT на SPT для DR- и RP-маршрутизаторов.

Параметры:

- RATE - порог скорости трафика (байт/сек);
- INTERVAL - интервал проверки RATE.

Если сообщения Register приходят на RP со скоростью выше RATE, RP шлет DR-сообщение RegisterStop и добавляет SPT-маршрут для передачи МД от источника.

Если сообщения Register приходят на NHR со скоростью выше RATE, DR также переходит на SPT-маршрут.

По умолчанию: rate: 6250 байт/сек; time: 20сек.

9.3.2 Настройка параметров интерфейса

Рассмотрим настройки на примере интерфейса ethernet0. Для настройки параметров интерфейса сначала нужно войти в конфигурацию соответствующего интерфейса:

(config)# router pim (config—
pirn)# iface ethernet 0

Чтобы настроить приоритет для интерфейса выполните команду

(config—pim—ethernet0)# preference 1

По умолчанию: значение указанное в default-preference. Приоритет влияет на выбор данного маршуртизатора как DR для данной сети LAN. Чем меньше данное значение, тем более вероятен выбор данного маршрутизатора как DR. При наличии параллельных проходов к источнику или RP для выбора маршрута применяются сообщения Assert. Используя сообщения Assert, адресованные 224.0.0.13 (группа all-pim-routers) в локальной сети, вышестоящий маршрутизатор может узнать, где осуществляется переадресация сообщений. Нижестоящие маршрутизаторы, получая сообщения Assert, узнают, какой маршрутизатор выбран в качестве DR, и куда следует посылать сообщение Join.

Чтобы настроить порог TTL для интерфейса выполните команду

(config —pim—ethernet0)# threshold 5

По умолчанию: 1. Команда задает минимальное значение IP TTL, требуемое для пересылки МД через данный интерфейс.

Если интерфейс должен обслуживать МП из разных удаленных подсетей, опишите эти подсети следующей командой:

(config — pim—ethernetO)# subnet 10.0.1.0/24

Если не указывать данной команды, то данный интерфейс будет обслуживать трафик только первичной подсети (заданной командой ip address в настройках интерфейса).

Чтобы запретить распространение МД указанной группы через интерфейс выполните команду:

(config —pim—ethernet0)# boundary 239.0.0.0/24

В данном случае запрещается передача МД группе 239.0.0.0/24. Команду рекомендуется использовать на PMBR-маршрутиазторах для создания границ распространения МД определенных групп.

Следующей командой вы можете полностью запретить МП на интерфейсе

(config —pim—ethernet0)# disable

Это равносильно удалению интерфейса из конфигурации router pim, однако в данном случае все прочие настройки интерфейса сохраняются, что более удобно, если в дальнейшем потребуется вновь включить интерфейс.

9.3.3 Пример настройки

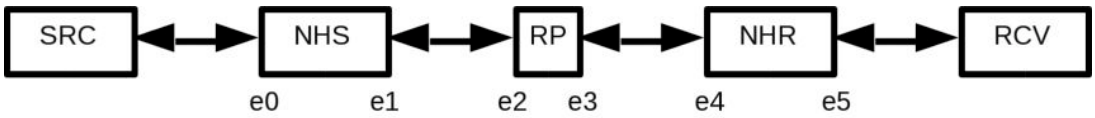

Рис. 9.2: Пример стенда PIM-SM

Пусть вещателя SRC и клиента RCV разделяет сеть из 3х маршрутизаторов (см. рис. 9.2). SRC вещает на группу 239.0.0.1/32. На схеме eN - обозначается интерфейс ethernetN, где N - его номер. Возможный упрощенный вариант настройки представлен далее.

Настройки NHS:

NHS(config — pim)# iface ethernet0
NHS(config —pim)# iface ethernetl

NHS работает как обычный PIM-маршрутизатор.

Настройки RP:

RP(config —pim)# iface ethernet2
RP(config —pim)# iface ethernet3
RP(config —pim)# rp—cand
RP(config —pim)# bsr—cand
RP(config —pim)# group 239.0.0.1/32

Настройки NHR:

```
RP(config—pim)# iface ethernet4
RP(config—pim)# iface ethernet5
RP(config—pim)# rp—cand
RP(config—pim)# bsr—cand
RP(config—pim)# group 239.0.0.1/32
```

RP и NHR работают как C-RP и C-BSR.

9.4 Настройка протокола IGMP

Включение ММ на основе только IGMP (без протокола динамической ММ) осуществляется командой:

```
(config)# router igmp
```

Если данная команда успешно выполнилась, по умолчанию все доступные для МП интрефейсы НЕ будут участвовать в МП.

Чтобы интерфейс участвовал в МП, необходимо определить один входящий интерфейс МП и один или более исходящих интерфейсов МП.

Определяем входящий интерфейс (интерфейс от источника многоадресного трафика):

```
(config—igmp)# input—iface
(config—igmp—in)# iface gre 1
```

Определяем исходящий интерфейс (интерфейс к потенциальным слушателям многоадресного трафика):

```
(config—igmp)# output—iface ethernet 0
```

Если в результате каких-либо настроек ММ на основе IGMP включится, будучи до этого выключенной (из-за недостаточных настроек), будет выведено сообщение: «Info: [igmp] igmp multicast routing enabled»

Если в результате каких-либо настроек ММ на основе IGMP выключится, будучи до этого включенной, будет выведено сообщение: «Info: [igmp] igmp multicast routing disabled»

9.4.1 Настройка интерфейса

Рассмотрим настройки интерфейса на примере исходящего интерфейса ethernet0:

```
(config)# router igmp
(config—igmp)# iface ethernet 0
```

Чтобы настроить порог TTL для интерфейса, следует выполнить команду

```
(config—igmp—out—ethernetO)# threshold 5
```

МД с TTL меньше указанного будут отбрасываться. По умолчанию: 1.

Чтобы настроить максимальную пропускную способность многоадресного трафика для интерфейса, следует выполнить команду

(config — igmp—out—ethernet0)# rate 100

Значение задается в Кбит/сек. По умолчанию: неограниченно.

Данные опции возможны и для входящего интерфейса. Однако для него добавляется дополнительная опция.

Если входящий интерфейс должен обслуживать МП из разных удаленных подсетей, следует описать эти подсети следующей командой:

(config —igmp—in)# subnet 10.0.0.0/24

9.4.2 Настройка протоколирования

Для включения протоколирования следует использовать следующую команду:

(config —igmp)# log [debug]

Необязательный параметр debug задает более подробный протокол.

9.4.3 Пример

Рассмотрим пример настройки IGMP (см. рис 9.3).

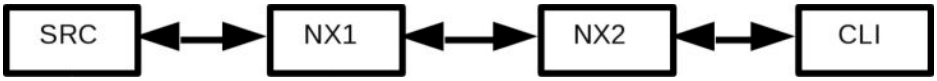

Рис. 9.3: Схема МП на основе IGMP

В данном примере:

- src - это источник МП, с адреса 10.0.0.0/24;
- cli - получатель МП;
- nxl, nx2 - маршрутизаторы Dionis-NX;
- между src и nxl: интерфейс ethernet0 на nxl;
- между nxl и пх2: интерфейс grel на nxl и пх2;
- между пх2 и cli: интерфейс ethernetl на пх2.

Настройка МП на основе двух igmp-маршрутизаторов Dionis-NX:

- настройки nxl:

(config—igmp)# input—iface
(config—igmp—in)# iface ethernet 0
(config—igmp)# output—iface gre 1

• настройки nx2:

(config—igmp)# input—iface (config—igmp—
in)# iface gre 1 (config—igmp—in)# subnet
10.0.0.0/24 (config—igmp)# output—iface
ethernet 1

9.1 Настройка статической многоадресной маршрутизации

Статическая ММ возможна только при отключенной динамической ММ (отсутствуют команды router pirn, router igmp, router dvmrp).

Для настройки статического маршрута следует ввести команду:

(config)# ip mroute <IFACE_IN> <MC_SRC_IP> <GROUP_IP> <IFACE_0UT>{1,8>

Параметры:

- IFACE_IN : интерфейс, откуда приходят МД (должен иметь 1P-адрес)
- MC_SRC_IP : IP-адрес источника МД
- GROUP_IP : групповой адрес
- IFACE_OUT : интерфейсы, куда должны направляться МД: может быть до 8 штук (должны иметь IP-адреса)

9.2 Мониторинг работы многоадресной маршрутизации

Мониторинг работы многоадресной маршрутизации осуществляется командой show multicast режима enable и ее подкомандами.

show multicast \<log | routes | vifs | igmp |igmp—joins| pirn |dvmrp [groups|cache] \>

Параметры:

- log - протоколы, в которых регистрируется работа МП;
- routes - таблица многоадресных маршрутов (для динамической и статической МП);
- vifs - таблица VIF-ов: интерфейсов используемых в МП (для динамической и статической МП);
- igmp - IGMP-информация о МП;

- igmp-joins - число присоединений к мультикаст-группам на интерфейсах
- pim - PIM-информация о МП;
- dvmrp - DVMRP-информация о МП;
    - groups - информация о группах DVMRP;
    - cache - информация о маршрутах DVMRP.

VIF - это виртуальный интерфейс, участвующий в МП и который на самом деле отображается на реальный интерфейс или локальный конец туннеля в системе.

9.3 Работа со службой

Для перезапуска службы МП выполните команду

router <igmp|pim|dvmrp> restart

9.4 Пример настройки видео-трансляции при помощи VideoLAN Vic

В примерах данной главы использовались источники и получатели МП.

Покажем, как можно осуществить передачу и прием МП при помощи ПО VidoeLAN Vic.

Для трансляции видео выполните на источнике МП:

- **vie -vvv c:\video.avi —sout "#udp{dst=239.0.0.1:1234}" —ttl 5 —sout-all**

Сервер осуществляет МП из файла c:\video.avi на мультикаст-адрес 239.0.0.1:1234 по протоколу UDP, причем TTL для МД устанавливается в значение 5, чтобы указанная МД достигла получателей.

Для получения трансляции выполните на получателе МП:

- **vie udp://@239.0.0.1:1234**

9.5 Сокращения и термины

В данной главе используются следующие сокращения:

- ММ - многоадресная маршрутизация
- МП - многоадресная передача
- МД - многоадресная датаграмма

10. NAT

NAT (Network Address Translation) — это механизм в сетях TCP/IP, позволяющий преобразовывать IP-адреса в заголовках пакетов. См. рис. 10.1. Различают два типа NAT: SNAT - замена адреса источника, и DNAT - замена адреса назначения.

SNAT используется для предоставления пользователям локальной сети с внутренними адресами доступа к внешней сети. DNAT используется для доступа из внешней к ресурсам внутренней.

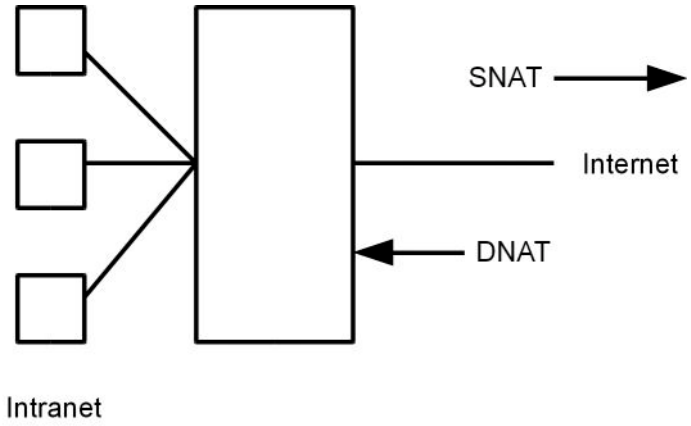

Рис. 10.1: Механизм NAT

В Dionis-NX NAT выполняется всегда на внешнем интерфейсе. При этом, SNAT начинает применяться для исходящего трафика, а DNAT - для входящего.

Очевидно, что и при SNAT и при DNAT для разных направлений трафика меняются как адреса источника, так и назначения. Например, в случае с преобразованием SNAT, исходящий с внешнего интерфейса пакет будет подвергнут изменению - его адрес источника будет заменен. При ответе, пакет также попадет в логику SNAT (для того, чтобы попасть во внутреннюю сеть), однако логика сопоставления внутренних адресов выполняется один раз для всего потока, и это происходит при выходе пакета с внешнего интерфейса.

Таким образом, преобразование SNAT для входящих во внешний интерфейс пакетов выполняется только для пакетов уже установленных изнутри соединений. Аналогично, преобразование DNAT для исходящих из внешнего интерфейса пакетов выполняется только если эти пакеты ассоциированы с установленным ранее соединением извне, подверженному преобразованию DNAT.

Следует иметь в виду, что если задано преобразования DNAT относительно какого-либо IP-адреса назначения, предполагается, что сетевые пакеты с таким адресом назначения дойдут до сетевого интерфейса. Чаще всего это означает необходимость задания дополнительного ip-адреса (ip secondary-address) для интерфейса, с которым связано преобразование DNAT.

Для созданий правил трансляции NAT в Dionis-NX используются списки NAT (ip nat-list).

Следует иметь в виду, что правила отбора в списках NAT применяются не для каждого пакета в отдельности, а только для первого пакета, устанавливающего соединение. NAT-преобразования пакетов выполняется только тогда, когда они принадлежат какому-либо соединению. Пакеты, не попадающие ни в одно соединение, считаются некорректными и не будут подвержены NAT-преобразованию.

10.1 Создание ip nat-list

Работа со списками NAT во многом совпадает с работой со списками контроля доступа. Так, для создания (редактирования) списка NAT необходимо в режиме configure выполнить следующую команду:

DionisNX(config)# ip nat—list mynat

При этом, произойдет переход в режим редактирования списка. Список содержит набор NAT-правил. Синтаксис правила выглядит следующим образом: nat скритерии отбора> <тип NAT> [параметры NAT], где критерии отбора являются подмножеством критериев списков контроля доступа и могут содержать:

Название	Значение
Протокол	IP-протокол потока
src	Адрес(а) источника потока
dst	Адрес(а) приемника потока
sport	Порт(ы) источника потока (для TCP и UDP)
dport	Порт(ы) приемника потока (для TCP и UDP)

Параметр <тип NAT> задает тип преобразования. Основные типы: snat, dnat или masquerade. MASQUERADE - это такой тип SNAT, который меняет адрес источника пакета (при выходе с внешнего интерфейса) на текущий адрес интерфейса.

Для преобразований snat и dnat необходимо указать ip-адрес замены, для преобразования masquerade этого не требуется. Например:

DionisNX(config)# ip nat—list mynat
DionisNX(config — nat—mynat)# nat src 192.168.0.0/24 masquerade
DionisNX(config — nat—mynat)# nat tcp dport 80 dnat ip 192.168.0.1

Для работы с элементами списка можно использовать числовые префиксы, также как и при работе со списками контроля доступа. Например:

DionisNX(config — nat—mynat)# do show
1 nat src 192.168.0.0/24 masquerade
2 nat tcp dport 80 dnat ip 192.168.0.1
DionisNX(config — nat—mynat)# no 1
DionisNX(config — nat—mynat)# do show
1 nat tcp dport 80 dnat ip 192.168.0.1

Для просмотра информации о NAT-списках, существует две команды, доступные из enable-режима.

show ip nat-list <имя|*> config	Информация о действующей конфигурации
show ip nat-list [имя]	Низкоуровневая информация из ядра ОС

Если в командах имя списка задано как *, будет показана информация о всех списках.

Например (из режима configure):

DionisNX(config)# do show ip nat—list mynat config

10.2 Другие типы NAT

Кроме основных типов преобразований (snat, dnat, masquerade) существуют также другие:

Название	Параметры	Действие
exclude in	критерии отбора	Исключает входящий трафик из NAT преобразования
exclude out	критерии отбора	Исключает исходящий трафик из NAT преобразования
redirect	для протоколов tcp/udp задается port <номер порта>	Перенаправлять трафик на локальный хост:порт (DNAT)
netmap src	критерии отбора, ip <адрес сети>	Отобразить целую сеть (SNAT)
netmap dst	критерии отбора, ip <адрес сети>	Отобразить целую сеть (DNAT)

10.3 Привязка ip nat-list

Создание списка преобразований NAT не означает то, что список начинает действовать. Для того, чтобы правила NAT начали действовать на проходящий трафик, список NAT должен быть привязан к интерфейсу.

10.3.1 Привязка к интерфейсу

Для того, чтобы привязать список к интерфейсу, нужно войти в режим конфигурации интерфейса и выполнить команду ip nat-group <имя списка>

Следует обратить внимание, что привязка nat списка осуществляется всегда к внешнему интерфейсу! Например:

DionisNX(config)# interface ethernet 0
DionisNX(config —if—ethernet0)# ip nat—group mynat

Для удаления связи с интерфейсом, необходимо выполнить команду: no ip nat-group <имя>, например:

DionisNX(config — if—ethernet0)# no ip nat-group mynat
DionisNX(config —if—ethernet0)# do show

Иногда возникает необходимость делать nat для трафика, который уходит в туннель DISEC. В этом случае, преобразование адресов должно выполняться до шифрования перед отправкой на интерфейс (SNAT) и после расшифрования, после приема на интерфейсе (DNAT). В этом случае.

привязка списков осуществляется командой: ip nat-group-xfrm, синтаксис которой аналогичен ip nat-group. Удаление связи осуществляется командой: no ip-nat-group-xfrm.

```
DionisNX(config)# interface ethernet 0
DionisNX(config—if— ethernetO)# ip nat-group—xfrnn mynat
DionisNX(config—if—ethernetO)# no ip nat-group—xfrm mynat
DionisNX(config—if—ethernetO)# do show
```

10.4 Просмотр и удаление активных соединений

Все проходящие через DionisNX соединения отслеживаются маршрутизатором и доступны для просмотра администратором. Для этого необходимо выполнить команду: show ip connections [параметры] из enable режима (или из режима configure, но с префиксом do).

Существует возможность просмотра соединений, над которыми выполняются NAT-преобразования, например:

```
DionisNX(config)# do show ip connections dnat tcp
```

В результате будут показаны TCP-соединения, над которыми выполнены преобразования DNAT.

При изменении параметров преобразований nat может оказаться необходимым очистить часть активных соединений, над которыми выполняются еще старые преобразования, для этого можно воспользоваться командой: clear ip connections [параметры] из enable режима, например:

```
DionisNX(config)# do clear ip connections dnat
```

Будут очищены все DNAT-соединения.

11. Helper

Некоторые протоколы используют различные соединения для передачи команд и данных. При использовании таких протоколов подсистема NAT должна заранее знать, что будет создано соединение и настроить его. Так, например, если создаётся FTP-соединение из внешней сети во внутреннюю, то FTP-сервер создаст соединение из внутренней сети во внешнюю. Чтобы это соединение корректно обработалось подсистемой NAT, необходимо создать helper.

Во многом команды управления helper-list похожи на команды управления access-list.

11.1 Создание helper-list

Для создания **helper-list** используется команда **ip helper-list.** Для добавления в неё правил используется команда **helper.** Пример:

ip helper—list ftp—ire—helper
1 helper ftp tep dst any dport 21
2 helper ire tep dst any dport 194

Аналогично при помощи команды ip6 helper-list создаются helper'bi для IPv6.

11.2 Применение helper-list

Для применения helper-list'oB используются команды ip helper-group и ip6 helper-group. Журналирование и отладка {#logdebug} =

В Dionis-NX существует несколько механизмов, которые могут быть использованы администратором для диагностики проблем, а также для выявления нарушений.

11.3 tepdump

Существует возможность мониторинга активности сети с помощью прослушивания сегмента Ethernet с выбранного интерфейса. Эти данные, в том или ином виде, в зависимости от указанных параметров выводятся на консоль. Для этого необходимо воспользоваться командой tepdump в режиме enable.

У tepdump существует множество параметров, с помощью которых можно выбирать формат вывода данных. Среди основных параметров можно выделить следующие:

Название параметра Описание	
proto <имя или числовое значение>	Задать интересуемый IP-протокол
тип интерфейса и его номер	Слушать на выбранном интерфейсе
dump <режим>	Вывод содержимого пакетов в заданном формате

Название параметра Описание	
host <1P-адрес или имя хоста>	Показ трафика относящегося к заданному хосту
src <1P-адрес или имя хоста>	Показ трафика с заданным исходным адресом
dst <1P-адрес или имя хоста>	Показ трафика с заданным целевым адресом
net <1P-адрес с маской>	Показ трафика относящегося к заданной сети
snet <1P-адрес с маской>	Показ трафика с адресом источника из заданной сети
dnet <1P-адрес с маской>	Показ трафика с адресом назначения из заданной сети
port сномер порта>	Показ трафика, относящемуся к заданному порту
sport сномер порта>	Показ трафика с заданным портом источника
dport сномер порта>	Показ трафика с заданным портом назначения
numeric	Не делать DNS-запросов и показывать адреса в числовом виде
ether	Выводить информацию по ethernet-заголовкам
count <число>	Закончить мониторинг после принятия <числа> пакетов, удовлетворяющих правилам выборки

Например:

DionisNX# tcpdump ethernet 0 numeric dump hex—ascii

Кроме того понимаются выражения со следующими служебными словами: **port, portrange, host** и **and, or, not.**

Например:

DionisNX# tcpdump ethernet 0 numeric dump hex—ascii not port 23

Для прерывания режима мониторинга необходимо нажать на клавиатуре Ctrl-C.

11.4 Трассировка

Механизм трассировки применяется для выявления проблем и ошибок администрирования, и позволяет проследить прохождение сетевого пакета внутри маршрутизатора.

Существует два режима работы трассировки:

1. Отладка фильтров;
2. Полная трассировка.

В режиме отладки фильтров протоколируется прохождение пакета по действующим фильтрам (входным и выходным). В режиме полной трассировки протоколируется весь путь прохождения пакета. Для задания режима трассировки необходимо перейти в режим конфигурации сервиса журналов (service log), для этого в режиме configure следует выполнить команду:

```
DionisNX(config)# service log
DionisNX(config—service—log)#
```

Режим работы трассировки задается с помощью команды trace. Так, например, для включения режима полной трассировки используется команда: trace all:

```
DionisNX(config—service—log)# trace all
DionisNX(config—service—log)# do show
log
 trace all
 size 262144 131072
```

Для отладки фильтров следует выполнить команду trace без параметров:

```
DionisNX(config—service—log)# trace
DionisNX(config—service—log)# do show
log
trace
 size 262144 131072
```

Существует возможность отключения механизма трассировки полностью, для этого используется команда no trace:

```
DionisNX(config—service—log)# no trace
DionisNX(config—service—log)# do show
log
 size 262144 131072
```

Для указания того, какой трафик должен быть подвержен трассировке, используются списки трассировки (ip trace-list и ip6 trace-list для протоколов IPv4 и IPv6 соответственно).

11.4.1 Создание списка трассировки

Для создания списка трассировки IPv4, в режиме configure необходимо выполнить команду ip trace-list <имя>, где <имя> - это имя создаваемого списка, например:

```
DionisNX(config)# ip trace—list ping
```

После выполнения команды, задаются (или модифицируются) правила трассировки списка. Каждое правило содержит критерии отбора трафика для трассировки. Если ни одно из правил не удовлетворяет критериям, трафик не будет трассироваться.

```
DionisNX(config—trace—ping)# trace icmp
```

В данном примере трассировке будет подвержен протокол icmp.

В качестве критериев отбора используется подмножество критериев списков контроля доступа (см. ip access-list).

Для того, чтобы просмотреть правила отбора текущего редактируемого списка, достаточно выполнить команду:

```
DionisNX(config—trace—ping)# do show
```

При этом будут выведены все правила текущего списка. Каждая строка снабжена числовым префиксом, указывающим позицию правила в списке.

Для того, чтобы удалить правило с конкретным номером, следует ввести команду: по сномер правила> Например:

DionisNX(config—trace—ping)# no 1

Удаление всего содержимого текущего списка может быть осуществлено командой: no all. Для удаления списка, используется команда: no ip trace-list <имя списка>.

Для просмотра информации о списках, существует две команды, доступные из епаЬle-режима.

show ip trace-list <имя|*> config (Информация о действующей конфигурации) show ip trace-list <имя|*> (Низкоуровневая информация из ядра ОС) Если в командах имя списка задано как *, будет показана информация о всех списках.

Например (из режима configure):

DionisNX(config)# do show ip trace—list ping config

Для работы со списками трассировки IPv6, используйте команды с префиксом ip6: ip6 trace-list, no ip6 trace-list и do show ip6 trace-list.

11.4.2 Применение списка трассировки

Для того, чтобы трассировка начала действовать, необходимо применить какой-то из списков трассировки. Применение списка осуществляется командой ip trace-group <имя списка> из режима configure для IPv4 и ip6 trace-group <имя списка> для IPv6.

Например:

DionisNX(config)# ip trace—group ping

Для отмена действия списка трассировки следует выполнить команду no ip trace-group <имя списка> (no ip6 trace-group <имя списка>):

DionisNX(config)# no ip trace—group ping

11.4.3 Выборка из журнала IP-пакетов

Если механизм трассировки включен, и в действующее правило отбора для трассировки попали пакеты, то содержимое пакетов и информация об их прохождении попадает в журнал IP-пакетов.

Для просмотра и выборки журнала используется команда show ip log [дополнительные параметры] из режима enable. Например:

```
DionisNX# show ip log
2011-12-08 16:50:46.749509 @in(ethernet0) TRACE: PREROUTING:rule: 1' IP (tos 0x60, ttl 59,
        id 24044, offset 0, flags [none], proto ICMP (1), length 84)
```

www.yandex.ru > 83.220.32.68: ICMP echo reply, id 7530, seq 1, length 64
2011-12-08 16:50:46.749517 @in(ethernetO) TRACE: outside:rule:4' IP (tos 0x60, ttl 59, id 24044, offset 0, flags [none], proto ICMP (1), length 84)
www.yandex.ru > 83.220.32.68: ICMP echo reply, id 7530, seq 1, length 64
2011-12-08 16:50:46.749533 @fwd(ethernet0->ethernet2) TRACE: FORWARD:policy: 1'IP (tos 0x60, ttl 58, id 24044, offset 0, flags [none], proto ICMP (1), length 84)
www.yandex.ru > 192.168.33.41: ICMP echo reply, id 7530, seq 1, length 64
2011-12-08 16:50:46.749541 @out(ethernet2) TRACE: POSTROUTING:policy: 1'IP (tos 0x60, ttl 58, id 24044, offset 0, flags [none], proto ICMP (1), length 84)

Запись в журнале содержит в себе: время, цепочку обработки, интерфейс, попадание в фильтры и информацию о пакете. Цепочка обработки может принимать значения, приведенные в таблице:

название	описание
in	вход в маршрутизатор
out	выход из маршрутизатора
fwd	логика маршрутизации
localjn	пакет предназначен для локального процесса
local_out	пакет порождается локальным процессом

К команде show ip log могут передаваться параметры, приведенные в таблице:

название	описание
число записей	показать последние п записей
all	показать все записи
proto < протокол >	выборка заданного IP-протокола
src <маска источника>	выборка для заданного источника
dst <маска приемника>	выборка для заданного приемника
in <тип интерфейса> сномер интерфейса>	выборка для пакетов, входящих в заданный интерфейс
out <ти интерфейса> сномер интерфейса>	выборка для пакетов, выходящих из заданного интерфейса
stat	вывод статистики
follow	режим непрерывного показа (слежение)
numeric	не разрешать адреса по DNS
quiet	кратко
dump <режим>	режим вывода содержимого пакета
date сдата или диапазон>	выборка по дате в формате T1 (конкретное время) или T1-T2(диапазон), где T1 или T2 записываются в формате: yy[mm[dd[hh[mm[ss]]]]]
file <файл>	выбор файла, из которого читать записи
export <файл>	сохранить вывод журнала в файл

11.5 Протоколирование правил фильтрации

Существует возможность протоколирования выбранных правил фильтрации. Для этого, в критериях отбора указывается параметр: log [all] [alert].

Например:

```
DionisNX(config)# ip access—list noicmp
DionisNX(config—acl — noicmp)# deny icmp log all
```

Присутствие ключевого слова all означает, что все данные пакета попадут в журнал. По умолчанию в журнал попадет только информация о заголовке пакета.

Присутствие ключевого слова alert означает повышенную важность сообщения.

Для произвольной выборки информации из журнала IP-пакетов используется команда: show ip log, которая описана в разделе «Трассировка».

11.6 Системные журналы

Кроме журнала IP-пакетов, в Dionis-NX поддерживается набор различных системных журналов. Для выборки данных журналов применяется команда: show log [имя журнала] [параметры] в режиме enable.

В качестве параметров могут присутствовать:

название	описание
число записей	показать последние n записей
all	показать все записи
follow	режим непрерывного показа (слежение)
archive <номер>	смотреть записи из архивных (старых)журналов

Для администратора доступны следующие журналы:

название	назначение
messages (журнал по умолчанию)	общесистемный журнал
system	системные события, относящиеся к конфигурации
dish	действия администратора
daemon	сообщения сервисов
kernel	сообщения ядра
router	сообщения от сервисов динамической маршрутизации
alert	сообщения, требующие внимания
auth	сообщения безопасности

Кроме этих системных журналов, существуют журналы для подсистем dhcp, dns, ntp, snmp и другие. Просмотр журнала для них возможен с помощью команды show service сназвание сервиса> log [параметры] в enable режиме:

DionisNX# show service dns log

11.7 Сигнал тревоги

Для оперативного информирования администратора о возникновении в системе важных событий предусмотрен сигнал тревоги (alert). Администратору следует отреагировать на сигнал тревоги. До этого момента сигнал тревоги снят не будет. Сигнал тревоги может проявляться следующим образом:

- Красный цвет лампочки на LCD-панели;
- Знак «!» вместо «#» в строке приглашения командной оболочки dish в привилегированном режиме;
- Звуковой сигнал от встроенного динамика;
- Отправка сообщений по протоколу syslog на удаленные syslog-сервера.

Важными событиями считаются все системные события, (см. раздел 11.6), уровень важности которых не ниже «критического». Такие события дополнительно попадают в системный журнал «alert». Для просмотра перечня важных событий из привилегированного режима (режим enable) необходимо выполнить следующую команду:

Router! show log alert

При этом сигнал тревоги будет снят. Также можно снять сигнал тревоги с помощью команды привилегированного режима:

Router! clear alert

При снятии сигнала тревоги системный журнал «alert» не очищается, поэтому администратор всегда может просмотреть важные события, происходившие в системе ранее.

11.7.1 Настройка звукового сигнала

Администратор имеет возможность настраивать возникновение звукового сигнала. Для включения звукового сигнала при возникновении сигнала тревоги в режиме конфигурации необходимо выполнить следующие команды:

Router(config)# service log
Router(config—service—log)# alert beep

Для выключения звукового сигнала при возникновении сигнала тревоги в режиме конфигурации необходимо выполнить следующие команды:

Router(config)# service log Router(config—
service—log)# no alert beep

При заводской установке системы звуковой сигнал по умолчанию включен.

11.7.2 Настройка удаленных оповещений

Сообщения о событиях в системе могут пересылаться на удаленные сервера. Для этого используется сетевой протокол syslog. На удаленном сервере для приема сообщений должен быть установлен и настроен syslog-сервер.

Администратор имеет возможность указать удаленные узлы, на которые будут отсылаться оповещения, и правила отбора сообщений для посылки:

```
Router(config)# service log Router(config—service—
log)# remote 192.168.1.2 4.crit Router(config—service—
log)# remote myhost.org dish.*
```

Приведенные выше команды добавят в список узлов два сервера: 192.168.1.2 и myhost.org. На первый из них будут отсылаться сообщения с приоритетом не меньшим, чем критический (что соответствует сигналу тревоги - alert). На второй сервер будут отсылаться все сообщения службы (facility) dish, что соответствует регистрации всех команд, введенных администратором в командной оболочке.

Для удаления узла из списка рассылки сообщений необходимо выполнить следующие команды в режиме конфигурации:

```
Router(config)# service log
Router(config—service—log)# no remote myhost.org dish.*
```

Правила отбора сообщений для отсылки на удаленный сервер соответствует формату syslog: »[служба].[приоритет]». Список допустимых служб:

---

4 auth;

- authpriv;
- daemon;
- kern;
- dish - соответствует localO в конфигурационном файле syslog;
- router - соответствует locall в конфигурационном файле syslog;
- dhcp - соответствует 1oca12 в конфигурационном файле syslog;
- dns - соответствует 1oca13 в конфигурационном файле syslog;
- watcher - соответствует 1oca17 в конфигурационном файле syslog;
- * - любая служба.

Список допустимых приоритетов:

- info;
- notice;
- warning;
- err;
- crit;
- alert;
- emerg;

- * - любой приоритет;
- none.

Также допустимы правила с использованием »,», »;», »!», » = ». Более подробную информацию по формату правил отбора и использованию специальных символов можно найти в документации по стандартному сервису syslog.

11.8 Служба watcher

Существует возможность отслеживать интересующие администратора события в журналах и собирать их в отдельный журнал, а также следить за доступным свободным местом файловой системы. Для этого необходимо настроить службу watcher. Вход в режим конфигурации службы выполняется командой:

```
Router(config)# service watcher
Router(config—service—watcher)#
```

Для слежения за доступным свободным местом файловой системы необходимо выполнить команду:

```
Router(config—service—watcher)# alert space 10mb
```

В данном случае, как только в системе останется менее 10 мегабайт свободного места, будет сформировано событие уровня alert.

Далее, настраиваются журналы за которыми необходимо осуществлять слежение. Для выбора журнала используйте команду watch <журнал>, например:

```
Router(config—service—watcher)# watch dish
Router(config—service—watcher—dish)#
```

При этом осуществляется переход в режим настройки правил слежения из выбранного журнала. Для удаления слежения за выбранным журналом используйте команду: no watch <имя журнала>. Каждое правило слежения состоит из команды (действия), строки сопоставления и дополнительных параметров.

Команда	Действие
alert	Послать сообщение в журнал watcher с уровнем alert
exit	Послать сообщение в журнал watcher с уровнем crit
emerg	Послать сообщение в журнал watcher с уровнем emerg
err	Послать сообщение в журнал watcher с уровнем err
info	Послать сообщение в журнал watcher с уровнем info
mailer	Послать сообщение по почте с помощью службы mailer
notice	Послать сообщение в журнал watcher с уровнем notice
warning	Послать сообщение в журнал watcher с уровнем warning
ssh	Послать команду на удаленный узел через ssh соединение

В качестве строки сопоставления используется регулярное выражение, на пример:

Router(config—service—watcher—dish)# alert "service watcher"

В данном случае, будет сформировано событие уровня alert при выполнении команды "service watcher". Регулярные выражения могут содержать шаблоны, состоящие из символьных классов:

Символьный класс	Соответствие символам
X	x - соответствует сам себе. (Он не может быть равен ни одному из специальных символов "$()%.[]*+-?
	Любой символ
%x	Где x - любой не алфавитно-цифровой символ), соответствует сам себе. Это - стандартный способ экранировки специальных символов
[set]	Соответствует любому символу из набора, заданного в set. Диапазон символов может быть определен, с помощью символа отделяющего начало и конец диапазона.
[^set]	Отрицательный набор символов. Соответствует любому символу, кроме тех, что заданы в наборе set.

Шаблон	Описание
Одиночный символьный класс	Соответствует любому одиночному символу из заданного класса
'*' за классом	Соответствует 0 или большему количеству повторений символов из заданного класса. Например: c* - символ с 0 или более раз.
'+' за классом	Соответствует 1 или большему количеству повторений символов из заданного класса. Эти элементы повторения будут всегда соответствовать самой длинной возможной последовательности.
'-' за классом	Соответствует 0 или большему количеству повторений символов из заданного класса. В отличие от *, элементы повторения будут всегда соответствовать самой короткой возможной последовательности
'?' за классом	Соответствует 0 или единственному вхождению символа из заданного класса;
^	Соответствует началу строки
$	Соответствует концу строки

Обратите внимание, что служебные символы требуют экранирования. Например:

Router(config—service—watcher—dish)# alert "ip access%—list"

Символ '-' здесь экранируется, так как он имеет специальный смысл.

Можно создавать несколько правил слежения. Для удаления правила используйте команду: по сномер правила>.

Обратите внимание, что правило "alert" удобно использовать для звукового и визуального оповещения о выбранных событиях.

Кроме правил, можно задать период слежения за журналом в секундах:

```
Router(config—service—watcher—dish)# period 1
```

Если период не задан, то по умолчанию период выбирается случайное значение периода от 3 до 5 секунд.

Для возвращения к настройке по умолчанию используйте: no period.

Для отправки сообщения по электронной почте необходимо предварительно настроить и активировать службу пересылки почтовых сообщений - MAILER.

```
Router(config—service—watcher— kernel)# mailer usb usrl@factor—ts.ru via gmail subject
    WARNING
```

В этом случае сообщение с темой "WARNING" будет отправлено на почтовый ящик usrl@factor-ts.ru, используя учетную запись gmail службы MAILER.

Для отправки команд на удаленный узел необходимо предварительно настроить ssh-соединение без использования паролей. Последовательность команд указывается через

```
Router(config—service—watcher— kernel)# ssh usb via usrl connect adm 192.168.33.79 command
    "configure terminal;do reboot"
```

В этом случае будут использованы ssh-ключи пользователя usrl

Кроме того можно указать подготовленный файл со списком команд:

```
Router(config—service—watcher— kernel)# ssh usb via usrl connect adm 192.168.33.79 file
    share :/remote.cmd
```

Файл должен быть следующего вида:

commandl;command2;command<N>;
command <N + l>;command<N+2>;

Для того, чтобы служба watcher стала активной, нужно выполнить команду enable из режима конфигурации службы:

```
Router(config—service—watcher)# enable
```

Для отключения службы, используйте: disable.

Для просмотра журналов службы используйте команду enaЬle-режима:

```
Router# show service watcher log
```

Для того, чтобы полностью удалить конфигурацию службы, воспользуйтесь командой:

```
Router(config)# no service watcher
```

12. RADIUS - аутентификация и авторизация с помощью AAA протокола

12.1 Введение

Dionis-NX поддерживает аутентификацию и авторизацию с помощью удаленного сервера по протоколу RADIUS. Это означает, что управление учетными записями пользователей возможно производить централизованно на удаленном сервере, а не на каждой машине в отдельности. В данном случае Dionis-NX является RADIUS-клиентом (сервером доступа NAS - Network Access Server)

Таким образом, когда пользователь хочет подключиться к Dionis-NX, вначале проверяется, существует ли пользователь на устройстве NAS и не является ли он radius-пользователем. Если пользователь существует и не является radius-пользователем, то аутентификация и авторизация происходит по обычной схеме. Если на устройстве NAS данный пользователь отсутствует или пользователь является radius-пользователем, то происходит подключение к RADIUS-серверу, где проверяются данные подключаемого пользователя и принимается решение о предоставлении доступа к системе. В случае положительного результата на устройстве NAS создается новый radius-пользователь или обновляются полномочия существующего.

12.2 Настройка для подключения к RADIUS серверу

Настройка подключения к RADIUS серверу происходит из режима **configure:**

```
adm@DionisNX(config)# radius

adm@DionisNX(config — radius)# server 10.10.10.1 testing123
adm@DionisNX(config — radius)# enable
```

Здесь указывается ip адрес RADIUS-сервера и пароль для подключения с нему.

12.3 Настройка RADIUS-сервера

Для того, чтобы RADIUS сервер мог принимать и обслуживать запросы от Dionis-NX необходимо подключить словарь, содержащий специфичные для DionisNX атрибуты. Словарь минимальный должен содержать следующее:

```
ATTRIBUTE    User-Name            1  string
ATTRIBUTE    Password             2  string
ATTRIBUTE    NAS—IP—Address       4 ipaddr
ATTRIBUTE    NAS—Port—Id          5integer
ATTRIBUTE    Service—Type 6 integer
VENDOR       Factor-TS            1156
```

```
BEGIN—VENDOR     Factor-TS
ATTRIBUTE       Nx—Usr—Realname      1 string
ATTRIBUTE          Nx—Usr—Status       2 string
ATTRIBUTE       Nx—Usr—Description 3 string
ATTRIBUTE          Nx—Role—Deps       4 string
END-VENDOR         Factor-TS
```

Пример конфигурирования пользователя u1 на RADIUS-сервере:

```
U1      Cleartext—Password := "555"
        Nx—Usr—Realname := "userl",
        Nx—Usr—Status                    :="supervisor",
        Nx—Usr—Description := "Default administrator USER1",
        Nx—Role—Deps             := "@default"
```

Где:

Cleartext-Password - пароль пользователя;

Nx-Usr-Realname - Настройка реального имени владельца учетной записи. Соответствует команде realname (смотри раздел "Учетные записи" данного руководства);

Nx-Usr-Status - Установка статуса супервизора (смотри раздел "Учетные записи" данного руководства);

Nx-Usr-Description - Настройка описания учетной записи пользователя. Соответствует команде description (смотри раздел "Учетные записи" данного руководства);

Nx-Role-Deps - Настройка полномочий пользователя. Соответствует команде delegate (смотри раздел "Учетные записи" данного руководства).

13. VLAN

Dionis-NX поддерживает стандарт IEEE 802.IQ (VLAN). Для создания интерфейса, выходящий трафик с которого будет помечаться идентификатором vlan-сети, а входящий трафик соответствующей vlan-сети, перенаправляться на вход этого интерфейса, используется команда: interface ethernet <номер интерфейса>.<номер vlan> (режим configure).

Например:

```
DionisNX(config)# interface ethernet 0.1
DionisNX(config —if—ethernet0.1)#
```

В дальнейшем, интерфейс настраивается так же, как и любой другой ethernet-интерфейс. Однако следует иметь в виду, что для активизации vlan-интерфейса необходимо, чтобы и соответствующий обычный интерфейс был активирован.

14. IEEE 802.lad (QinQ)

Dionis-NX поддерживает стандарт IEEE 802.lad (QinQ - вложенный в VLAN VLAN). Для создания интерфейса, выходящий трафик с которого будет помечаться идентификатором vlan-сети, а затем перенаправляться в VLAN интерфейс. А входящий трафик соответствующей под VLAN-сети, перенаправляться с VLAN интерфейса на вход этого интерфейса, используется команда: interface ethernet сномер интерфейса>.<номер vlan>ad.<НОМер> (режим configure).

Например (при условии, что VLAN ethernet 0.1 был создан ранее):

```
DionisNX(config)# interface ethernet 0.1ad.2
DionisNX(config — if—ethernetO.lad.2)#
```

В дальнейшем, интерфейс настраивается так же, как и любой другой ethernet-интерфейс. Однако следует иметь в виду, что для активизации QinQ-интерфейса необходимо, чтобы и соответствующие VLAN и обычный интерфейс были созданы и активированы.

15. VXLAN

VXLAN является технологией сетевой виртуализации, созданной для решения проблем масштабируемости в больших системах. Она использует схожую с VLAN технику для MAC инкапсуляции Layer 2 Ethernet кадров в UDP-пакеты, порт 4789.

Для создания интерфейса vxlan для работы с IPv4 используйте команду: interface vxlan сномер интерфейса>

```
DionisNX(config)# interface vxlan 0
DionisNX(config —if—vxian0)#
```

Для создания IPv6 vxlan интерфейса используйте команду: interface ip6vxlan сномер интерфейсам

Интерфейс vxlan в целом настраивается так же, как и интерфейсы других типов, однако имеется набор параметров, которые применимы только для vxlan-туннелей. Эти параметры приведены в таблице:

команда	параметр
id <ID>	Идентификатор (обязательный параметр)
local <1P-адрес>	IP-адрес локального конца туннеля (unicast режим)
remote <1P-адрес>	IP-адрес удаленного конца туннеля (unicast режим)
ttl <значение>	Принудительно устанавливать значение ttl для UDP-датаграммы, а не наследовать его из инкапсулируемого пакета
tos <тип_трафика> |inherit	Настройка типа трафика (по качеству обслуживания)
group смультикаст группа>	Задать группу мультикаст (multicast режим)
srcport спорт источник>	Явное задание UDP порта источника
dstport спорт назначения>	Явное задание UDP порта назначения

16. WIFI-интерфейсы

16.1 Введение

Dionis-NX имеет поддержку беспроводных интерфейсов и может работать как в режиме беспроводной точки доступа, так и в режиме беспроводного клиента.

Для работы с wifi-интерфейсом используется команда: interface wifi из режима configure.

```
adm@DionisNX(config)# interface wifi 0
adm@DionisNX(config — if—wifi0)#
```

16.2 Работа WIFI-интерфейса в режиме беспроводной точки доступа

Для перевода интерфейса в режим точки доступа необходимо выполнить следующую команду:

```
adm@DionisNX(config — if—wifiO)# mode master
```

Команды доступные для настройки интерфейса в режим точки доступа.

команда	параметр
ssid <Name>	Имя беспроводной сети
passphrase <Password>	Пароль беспроводной сети
channel <Num>	Номер канала беспроводной сети
hw_mode <a|b|g|ad>	Режим работы точки доступа
ignore-broadcast-ssid <1|2|0>	Скрывать SSID. (0 - параметр отключен, 1 - Передавать пустой SSID, 2 - Передавать пустой SSID, но сохронять длинну ssid)
max_num_sta <Num>	Максимальное количество подключаемых станций
wpa <WPA| WPA2| WPA/WPA2>	Настройка режима безопасности
wpa-key-mgmt PSK	Установить алгоритм управления ключами протокола
wpa-pairwise <CCMP|TKIP|CCMP/TKIP>	Установить алгоритм шифрования для WPA(WPA2)

16.3 Работа WIFI-интерфейса в режиме беспроводного клиента

Для перевода интерфейса в режим клиента необходимо выполнить следующую команду:

```
adm@DionisNX(config — if—wifiO)# mode managed
```

Настройка интерфейса в режиме клиента сводится к настройке профилей подключения к беспроводной сети и настройке ip адреса интерфейса.

Для интерфейса доступно несколько профилей подключения. Интерфейс поочередно пытается установить соединение с заданной сетью из каждого профиля.

Для создания нового профиля или входа в текущий необходимо выполнить команду network <Имя>

adm@DionisNX(config — if— wifi0)# network net1
adm@DionisNX(config —if—wifi0—net1)#

Команды доступные для настройки профилей подключения.

команда	параметр
ssid <Name>	Имя беспроводной сети
passphrase <Password>	Пароль беспроводной сети
priority <Num>	Установить приоритет подключения
wpa <WPA| WPA2| WPA/WPA2>	Настройка режима безопасности
wpa-key-mgmt <NONE|WPA-PSK|WPA-PSK/NONE>	Установить алгоритм управления ключами протокола
wpa-pairwise <CCMP|TKIP|CCMP/TKIP|NONE>	Установить алгоритм шифрования для WPA(WPA2)

Ниже представлен минимальный набор команд для подключения к сети TestNet с паролем 12345678:

adm@DionisNX(config — if— wifi0)# mode managed
adm@DionisNX(config — if— wifi0)# network net1
adm@DionisNX(config—if—wifi0—net1)# ssid TestNet
adm@DionisNX(config—if—wifi0—net1)# passphrase 12345678
adm@DionisNX(config—if—wifi0—net1)# exit
adm@DionisNX(config — if— wifi0)# ip address dhcp
adm@DionisNX(config — if— wifi0)# enable

16.4 Прочие команды, доступные для работы с WIFI-интерфейсом

Для просмотра информации о доступных беспроводных сетях необходимо выполнить команду:

adm@DionisNX(config — if— wifi0)# scan

Для просмотра только имен доступных беспроводных сетей необходимо выполнить команду:

adm@DionisNX(config — if— wifi0)# scan essid

17. MODEM-интерфейсы

17.1 Введение

Dionis-NX имеет поддержку 3G/LTE USB модемов.

Для работы с modem-интерфейсом используется команда: interface modem из режима configure.

adm@DionisNX(config)# interface modem 0
adm@DionisNX(config — if— modem0)#

17.2 Команды доступные для настройки интерфейса

команда	параметр
apn <Name>	Имя точки доступа
user <UserName>	Имя пользователя для аутентификации
password <Password>	Пароль для аутентификации
phone <Num>	Номер дозвона
speed <Speed>	Скорость модема (необязательный параметр)
modem-backend <Port>	Номер порта модема

17.3 Номер порта модема (modem-backend)

Модем может иметь несколько портов. На данный момент в dionis-nx указание конкретного рабочего порта модема производится вручную командой modem-backend.

17.4 Пример настройти интерфейса

Пример настройки модема Huawei E3272 для megafon

adm@DionisNX(config)# interface modem 0
adm@DionisNX(config—if— modem0)# adm@DionisNX(config—
if— modem0)# apn internet adm@DionisNX(config—if—
modem0)# user gdata adm@DionisNX(config—if—modem0)#
password gdata adm@DionisNX(config—if— modem0)# phone
*99# adm@DionisNX(config—if— modemO)# modem—backend
USBO

Пример настройки модема Huawei E3272 для beeline

```
adm@DionisNX(config)# interface modem 1
adm@DionisNX(config — if— modem 1)#
adm@DionisNX(config—if—modeml)# apn internet.beeline.ru
adm@DionisNX(config—if— modeml)# user beeline
adm@DionisNX(config—if—modeml)# password beeline
adm@DionisNX(config — if— modeml)# phone *99#
adm@DionisNX(config — if— modeml)# modem—backend USB0
```

18. Bonding-интерфейсы

Dionis-NX поддерживает агрегацию (bonding) интерфейсов путем объединения нескольких физических интерфейсов в один логический (мастер), что можно использовать для повышения пропускной способности или в целях резервирования.

Создание/редактирование мастер-интерфейса осуществляется с помощью команды: interface bond <номер> в режиме configure.

После создания bonding-интерфейса, управление им осуществляется в целом так же, как и любым другим интерфейсом, за исключением следующих особенностей:

1. С помощью команды slave должны быть указаны подчиненные интерфейсы (или один интерфейс), которые будут использоваться для агрегации;
2. Подчиненные интерфейсы не должны использоваться маршрутизатором (кроме как в агрегации) и должны находиться в отключенном состоянии;
3. С помощью команды mode должен быть выбран режим агрегации (если режим не задан - используется режим по умолчанию);
4. С помощью команды monitor желательно задать режим мониторинга состояния подчиненных интерфейсов;

18.1 Режимы агрегации

Существуют следующие режимы агрегации:

режим	описание
balance-rr	режим по умолчанию, циклическое использование подчиненных интерфейсов
active-backup	режим резервирования
balance-xor	распределение зависит от хеш-функции
broadcast	одновременная передача по всем интерфейсам
802.3ad	IEEE 802.3ad
balance-tlb	адаптивная балансировка передачи
balance-alb	адаптивная балансировка (в том числе и на приеме)

18.1.1 balance-rr

Этот режим используется по умолчанию, если в настройках не указано другое, balance-rr обеспечивает балансировку нагрузки и отказоустойчивость. В данном режиме пакеты отправляются «по кругу» от первого интерфейса к последнему и сначала. Если выходит из строя один из интерфейсов, пакеты отправляются на остальные оставшиеся. При подключении портов к разным коммутаторам необходимо выполнить их настройку.

18.1.2 active-backup

Работает только один интерфейс, остальные находятся в очереди горячей замены. Если ведущий интерфейс перестает функционировать, то его нагрузку подхватывает следующий (присвоив mac-адрес) и становится активным. Дополнительная настройка коммутатора не требуется.

Внимание: подпараметр fail-over-mac может быть изменен только в том случае, если в текущем bonding-интерфейсе нет подчиненных интерфейсов.

18.1.3 balance-xor

XOR-политика: Выбор подчиненного интерфейса выполняется на основе хеш-функции [по умолчанию: (исходный MAC-адрес □ XOR □ MAC-адрес получателя) %число интерфейсов]. Режим обеспечивает балансировку нагрузки и отказоустойчивость.

18.1.4 broadcast

Все пакеты передаются на все интерфейсы в группе. Режим обеспечивает отказоустойчивость.

18.1.5 802.3ad

IEEE 802.3ad Dynamic Link aggregation (динамическое объединение каналов). Создает агрегации групп, имеющие одни и те же скорости и дуплексные настройки. Использует все включенные интерфейсы в активном агрегаторе согласно спецификации 802.3ad.

Необходимы коммутаторы с поддержкой IEEE 802.3ad Dynamic Link aggregation. Большинство параметров потребует некоторой конфигурации для режима 802.3ad.

18.1.6 balance-tlb

Адаптивная балансировка передаваемой нагрузки: канал связи не требует какой либо специальной настройки. Исходящий трафик распределяется в соответствии с текущей нагрузкой (вычисляется по скоростям) для каждого интерфейса. Входящий трафик принимается текущим интерфейсом. Если принимающий интерфейс выходит из строя, то следующий занимает его место, приватизировав его mac-адрес.

18.1.7 balance-alb

Адаптивное перераспределение нагрузки: включает balance-tlb плюс receive load balancing (rib) для трафика IPv4 и не требует специального конфигурирования. То есть все так же как и при

режиме balance-tlb, только дополнительно и входящий трафик тоже балансируется между интерфейсами. Полученная балансировка нагрузки достигается опросом ARP. Драйвер перехватывает ответы ARP, направленные в локальной системе в поисках выхода, и перезаписывает исходный адрес сетевой карты с уникальным аппаратным адресом одного из интерфейсов в группе.

18.1.8 Настройка режимов хеширования

Если выбран один из режимов: 802.3ad или balance-xor, то можно задать режим хеширования для хеш-функции, которая используется для выбора подчиненного интерфейса. Доступны следующие режимы хеширования:

- layer2: исключающее «или» по MAC-адресам источника и приемника (функция по умолчанию);
- layer2+3: исключающее «или» по MAC-адресам источника и приемника, а также по IP-адресам приемника и источника;
- layer3+4: исключающее «или» по IP-адресам приемника и источника, а также портам источника и назначения. Порты источника и назначения используются при расчете для протоколов TCP и UDP только в случае не фрагментированных пакетов.

18.2 Режимы мониторинга

Существует два режима мониторинга состояния: с помощью arp-проб и состояния линии mii. Если выбран режим arp, то требуется указать IP-адреса удаленных подчиненных интерфейсов.

Для удаления подчиненных интерфейсов можно использовать команду no slave <интерфейс>:

```
DionisNX(config)# interface bond 0
DionisNX(config —if—bond0)# slave ethernet 0
DionisNX(config —if—bond0)# slave ethernet 2
DionisNX(config —if—bond0)# no slave ethernet 2
DionisNX(config —if—bond0)# slave ethernet 1
DionisNX(config —if—bond0)# monitor mii 100
DionisNX(config —if—bond0)# mode active—backup
DionisNX(config —if—bond0)# enable
```

19. Сетевые мосты

Dionis-NX может выступать в роли Ethernet-коммутатора, позволяя объединять сегменты сети с помощью сетевых мостов. Сетевой мост представлен в системе интерфейсом особого типа - bridge. Затем, в интерфейс добавляются порты, трафик с которых будет пересылаться.

Например:

```
DionisNX(config)# interface bridge 0
DionisNX(config — if— bridge0)# port ethernet 0
DionisNX(config — if— bridge0)# port ethernet 1
DionisNX(config — if— bridge0)# enable
```

В данном примере, порты ethernet 0 и etherent 1 объединяются в сетевой мост. Пакеты, приходящие в порты, передаются на основе Ethernet-адресов, а не IP-адресов (как в маршрутизаторе). Поскольку передача выполняется на канальном уровне (уровень 2 модели OSI), все протоколы более высокого уровня прозрачно проходят через мост.

Для удаления портов из моста следует использовать команду no port <интерфейс>. Существует также команда no port all - для удаления всех портов.

У сетевого моста существуют следующие параметры:

Команда	Значение
[no] ageing	Задание времени ageing - времени жизни MAC-адреса в базе данных маршрутизации в секундах
[по] fd	Задание задержки маршрутизации в секундах

Сетевой мост в Dionis-NX поддерживает протокол STP (Spanning Tree Protocol), который используется для того, чтобы избежать петель коммутации. Для настройки STP следует использовать следующие команды:

Команда	Описание
[по] stp	Включение/выключение протокола stp
[по] hello	Задание времени hello (stp)
[по] maxage	Задание времени maxage (stp)

Сетевой мост в Dionis-NX поддерживает более тонкую настройку его работы с многоадресным трафиком. В дальнейшем описании будем использовать обозначение МР для роутера осуществляющего многоадресную передачу (МП), а многоадресный трафик - МП-трафиком. Для данной настройки следует использовать следующие команды:

Команда	Описание
igmp-routing on	система сама определяет (отслеживание IGMP Query) пристуствие МР подключенных к порту сетевого моста; если МР обнаружен - разрешается МП через данный порт, иначе - МП через данный порт не будет осуществляться

Команда	Описание
igmp-routing off	порт не видит МР, подключенных к нему и МП через порты моста не будет осуществляться
no igmp-routing	по умолчанию: разрешается МП через порты моста, независимо оттого, есть ли МР, подключенные к портам моста
igmp-snooping on	сетевой мост начинает анализировать все IGMP-пакеты между подключенными к нему потребителями и поставщиками МП-трафика; обнаружив запрос IGMP-Join (присоединиться к группе) потребителя, мост включает порт, к которому тот подключён; обнаружив запрос IGMP-Leave (покинуть группу), удаляет соответствующий порт из списка группы.
igmp-snooping off	сетевой мост без разбора ретранслирует МП-трафик по всем своим портам
no igmp-snooping	по умолчанию поведение соответсвует igmp-snooping on.

Для просмотра информации о сетевом мосте используйте команду show bridge [<iface>], на пример:

DionisNX# show bridge 0

20. GRE-туннели

Dionis-NX поддерживает тунелирование по протоколу GRE. Для этого используются интерфейсы типа gre. После создания такого виртуального интерфейса, весь трафик, попадающий по правилам маршрутизации на этот туннель, инкапсулируется в GRE-пакеты.

Для создания gre туннеля используется команда interface gre <номер> из режима configure. При этом номер интерфейса может начинаться с 1.

```
DionisNX(config)# interface gre 1
DionisNX(config —if—gre1)#
```

Интерфейс gre в целом настраивается также, как и интерфейсы других типов, однако имеется набор параметров, которые применимы только для gre-туннелей. Эти параметры приведены в таблице:

команда	параметр
local <IP-адрес>	IP-адрес локального конца туннеля (обязательный параметр)
remote <IP-адрес>	IP-адрес удаленного конца туннеля (обязательный параметр)
ttl <значение>	Принудительно устанавливать значение ttl для GRE-датаграммы, а не наследовать его из инкапсулируемого пакета
keepalive <период> <число повторных попыток>	Включить механизм проб. Пробы посылаются через <период> секунд, и <число повторных попыток> + 1 раз
tos <тип_трафика> |inherit	Настройка типа трафика (по качеству обслуживания)
checksum	Включить генерацию/проверку контрольных сумм
sequence	Включить упорядочивание пакетов
key <id>	Установить ID для туннеля

Реальное создание интерфейса происходит после того, как заданы параметры local и remote. Например:

```
DionisNX(config)# interface gre 1
DionisNX(config —if—gre1)# local 192.168.0.1
DionisNX(config —if—gre1)# remote 192.168.1.1
DionisNX(config —if—gre1)# ip address 10.0.0.1/32
DionisNX(config —if—gre1)# enable
```

21. GRETAP-туннели

Существует возможность инкапсуляции ethernet-кадров на уровень IP, для этого используется особая разновидность туннелей GRETAP. Синтаксис команд для работы с GRETAP-туннелями полностью повторяет команды для работы с GRE-туннелями, за исключением того, что тип интерфейса задается как gretap, а не как gre. Например:

```
DionisNX(config)# interface gretap 1 DionisNX(config—if—
gretap1)# local 192.168.0.1 DionisNX(config—if— gretap1)#
remote 192.168.1.1 DionisNX(config—if—gretap1)# keepalive 5
DionisNX(config—if—gretap1)# ip address 10.0.0.1/32
DionisNX(config—if—gretap1)# enable DionisNX(config—if—
gretap1)# do show interface gretap 1 link
```

22. IP6GRE-туннели

Настройка IP6GRE-туннелей аналогична настройке GRE-туннелей. Например:

```
DionisNX(config)# interface ip6gre 2
DionisNX(config—if—ip6gre2)# local 2001:db8::5:8
DionisNX(config—if—ip6gre2)# remote 2001:db8: :3:5
DionisNX(config—if—ip6gre2)# keepalive 5
DionisNX(config—if—ip6gre2)# ip address 50.0.0.7/32
DionisNX(config—if— ip6gre2)# enable
```

23. IP6GRETAP-туннели

Настройка IP6GRETAP-туннелей аналогична настройке GRETAP-туннелей. Например:

DionisNX(config)# interface ip6gretap 5
DionisNX(config—if—ip6gretap5)# local 2001:db8::12:1 DionisNX(config—if—
ip6gretap5)# remote 2001:db8:: 11:2 DionisNX(config—if—ip6gretap5)#
keepalive 5 DionisNX(config—if—ip6gretap5)# ip address 30.0.0.4/32
DionisNX(config—if— ip6gretap5)# enable

24. VPN-туннели

24.1 Введение

Dionis-NX имеет поддержку технологии OpenVPN для создания зашифрованных каналов типа точка-точка или сервер-клиенты между компьютерами.

Настройка OpenVPN в системе Dionis-NX сводится к настройке динамических интерфейсов типа svpn (серверный интерфейс) и vpn (клиентский интерфейс).

24.2 Импорт, удаление и просмотр доступных ключей и сертификатов

Для использования vpn и svpn интерфейсов в режиме с TLS-аутентификацией или pre-shared ключом защиты необходимо предварительно импортировать соответствующие ключи и сертификаты (далее - объекты). Для импорта объектов используется команда vpn import <тип объекта> <путь к файлу> из enable-режима.

adm@DionisNX# vpn import ca share:/crt/ca.crt

Для удаления импортированного объекта используется команда vpn remove <тип объекта> сИмпортированный объект> из епаЬle-режима.

adm@DionisNX# vpn remove ca ca.crt

Внимание! Для удаления доступны только неиспользуемые объекты!

Для просмотра информации обо всех доступных объектах, а также о том, какие объекты используются необходимо ввести команду:

adm@DionisNX# show vpn

Команды доступные для работы с объектами.

команда	параметр
vpn import ca <CA>	Импорт корневого сертификата
vpn import cert <Cert>	Импорт сертификата сервера или клиетна
vpn import cert-key <Cert-key>	Импорт ключа сертификата сервера или клиетна
vpn import dh-key <Dh>	Импорт ключа Диффи-Хеллмана
vpn import psk <Psk>	Импорт pre-shared ключа защиты
vpn import tls-key <Tls-auth>	Импорт tls-auth ключа
vpn remove unused	Удаление всех неиспользуемых объектов
vpn remove ca unused	Удаление всех неиспользуемых ca объектов
vpn remove ca <CA>	Удаление импортированного корневого сертификата
vpn remove cert unused	Удаление всех неиспользуемых cert объектов

команда	параметр
vpn remove cert <Cert>	Удаление импортированного сертификата сервера или клиетна
vpn remove cert-key unused	Удаление всех неиспользуемых cert-key объектов
vpn remove cert-key <Cert-key>	Удаление импортированного ключа сертификата сервера или клиетна
vpn remove dh-key unused	Удаление всех неиспользуемых dh-key объектов
vpn remove dh-key <Dh>	Удаление импортированного ключа Диффи-Хеллмана
vpn remove psk unused	Удаление всех неиспользуемых psk объектов
vpn remove psk <Psk>	Удаление импортированного pre-shared ключа защиты
vpn remove tls-key unused	Удаление всех неиспользуемых tls-key объектов
vpn remove tls-key <Tls-auth>	Удаление импортированного tls-auth ключа
show vpn	Отображение списка всех импортированных объектов
show vpn ca	Отображение списка импортированных корневых сертификатов
show vpn ca <CA>	Просмотр корневого сертификата <CA>
show vpn cert	Отображение списка импортированных сертификатов сервера или клиетна
show vpn cert <Cert>	Просмотр сертификата <Cert> сервера или клиетна
show vpn cert-key	Отображение списка импортированных ключей сертификатов сервера или клиетна
show vpn dh-key	Отображение списка импортированных ключей Диффи-Хеллмана
show vpn psk	Отображение списка импортированных pre-shared ключей защиты
show vpn tls-key	Отображение списка импортированных tls-auth ключей

24.3 VPN-интерфейс

Для создания vpn-интерфейса используется команда: interface vpn <номер> из режима configure. При этом номер интерфейса может начинаться с 0.

```
adm@DionisNX(config)# interface vpn 1
adm@DionisNX(config — if—vpnl)#
```

Режимы работы vpn-интерфейса:

1. Простой туннель без защиты;

2. Туннель с pre-shared ключом защиты;
3. Туннель с TLS-аутентификацией;
4. Работа в режиме клиента openvpn для подключения к мультиклиент-серверу openvpn.

Настройка интерфейса происходит в два этапа:

1. Настройка connection-блока;
2. Прочая настройка.

Настройка connection-блока.

Для интерфейса vpn доступно несколько профилей подключения (connection-блоков). Интерфейс поочередно пытается установить соединение с удаленным концом из каждого блока.

Для входа в connection-блок необходимо выполнить команду connection <Имя>

```
adm@DionisNX(config — if—vpn1)# connection block1
adm@DionisNX(config —if—vpn1 —block1)
```

Команды доступные для настройки connection-блоков

команда	параметр
lport <Номер порта>	Номер порта на локальном конце туннеля. По умолчанию 1194 (Необязательный параметр)
rport <Номер порта>	Номер порта на удаленном конце туннеля. По умолчанию 1194 (Необязательный параметр)
port <Номер порта>	Номер порта на локальном и удаленном конце туннеля. По умолчанию 1194 (Необязательный параметр)
local <ip или имя хоста>	Локальный ip или имя хоста (Необязательный параметр)
proto < Прото кол >	Протокол работы интерфейса. По умолчанию udp. Должен совпадать с протоколом на удаленном конце туннеля
bind	Команда связывает локальный адрес и порт
remote <ip или имя хоста>	Удаленный ip или имя хоста, к которому будет происходить подключение (Обязательный параметр)

Примечание: В connection-блоке может быть несколько remote. Для удаления конкретного remote необходимо выполнить команду:

```
no remote <1P>
```

Например:

```
adm@DionisNX(config — if—conn)# no remote 192.168.10.1
adm@DionisNX(config —if—conn)
```

Прочие настройки.

Дополнительные команды для настройки vpn-интерфейса:

команда	параметр
tls-client	Включить TLS и быть клиентом во время handshake. Эта команда подразумевает обязательный ввод команд <ca>, <cert>, <key>
ca <Корневой сертификат>	Предварительно импортированный корневой сертификат, ca-файл должен быть такой же как на сервере
cert cСертификат клиента>	Предварительно импортированный сертификат клиента
key <Ключ клиента>	Предварительно импортированный ключ сертификата клиента
tls-auth <Дополнительный ключ>	Предварительно импортированный tls-auth ключ. Данная команда добавляет дополнительный слой аутентификации, tls-auth-файл должен быть такой же как на сервере (Необязательный параметр. Используется совместно с <tls-client>)
secret <Pre-shared ключ>	Предварительно импортированный pre-shared ключ в режиме работы туннеля с pre-shared ключом защиты. Файл ключа должен быть одинаковым на обоих концах туннеля
ifconfig < l IP: r_IP>	Приватный адрес локального и удаленного конца туннеля. (Обязательная команда для всех режимов работы интерфейса кроме режима клиента openvpn. В данном режиме команда не используется)
cipher <Алгоритм>	Алгоритм шифрования.
auth <Алгоритм>	Алгоритм Аутентификации.
ping <Интервал в секундах>	ping удаленного конца туннеля, если нет передачи пакетов в течение промежутка времени, большего чем указанный интервал
ping-restart <Интервал в секундах>	Перезагрузка подключения к удаленному концу туннеля, если от него не приходило пакетов в течение промежутка времени, большего чем указанный интервал
pull	Обязательная команда при работе интерфейса в режиме клиента openvpn
ns-cert-type <nsCertType>	Требовать <nsCertType> в поле nsCertType сертификата соседа
tls-cipher <Alg>	Алгоритм tls-шифра. Используется для повышения уровня безопасности контроля канала управления (TLS used only as a control channel).

Ниже приводятся примеры конфигурации интерфейса для различных режимов работы. (В данных примерах настраиваемый интерфейс является локальным концом туннеля).

1. Простой туннель без защиты. В данном режиме необходимо указать имя удаленного хоста или его ip-адрес, а так же локальный и удаленный ip-адреса туннеля. Пример настройки интерфейса:

```
adm@DionisNX(config —if—vpn1)# connection block1
adm@DionisNX(config —if—vpn1 —block1)
adm@DionisNX(config —if—vpn1 —block1)# remote 192.168.33.232
adm@DionisNX(config — if—vpn1 — block1)# exit
adm@DionisNX(config —if—vpn1)# ifconfig 10.8.0.1:10.8.0.2
```

Здесь 192.168.33.232 - это адрес удаленного конца. 10.8.0.1 - локальный адрес туннеля, 10.8.0.2 - удаленный адрес туннеля. Удаленный конец туннеля необходимо также настроить для подключения к vpn-туннелю, указав remote и ifconfig. ifconfig на удаленном конце в данном случае будет 10.8.0.2:10.8.0.1

2. Туннель с pre-shared ключом защиты. Данный режим повторяет настройку простого туннеля без защиты. Кроме этого, необходимо на обоих концах туннеля указать pre-shared ключ.

```
adm@DionisNX(config —if—vpn1)# secret key.pern
```

Ключ должен быть указан как на локальном, так и на удаленном конце туннеля, храниться в секрете и передаваться по защищенному каналу.

3. Туннель с TLS - аутентификацией. Данный режим повторяет настройку простого туннеля без защиты. Кроме того, необходимо выполнить следующее: На локальном конце туннеля выполнить команду <tls-client> а также указать корневой сертификат, и ключ локального конца.

```
adm@DionisNX(config —if—vpn1)# tls—client
adm@DionisNX(config —if—vpn1)# ca ca.crt
adm@DionisNX(config — if—vpn1)# cert client.crt
adm@DionisNX(config — if—vpnl)# key client.key
```

В этом случае удаленный конец туннеля будет выступать в качестве tls-сервера и должен быть настроен соответствующим образом.

4. Работа в режиме клиента openvpn для подключения к мульти-клиент-серверу openvpn. Для подключения к мульти-клиент-серверу openvpn необходимо выполнить следующие команды:

```
adm@DionisNX(config — if—vpn1)# connection conn
adm@DionisNX(config —if—vpn1—conn)# remote 192.168.33.232
adm@DionisNX(config —if—vpn1—conn)# exit
adm@DionisNX(config —if—vpn1)# tls—client
adm@DionisNX(config — if—vpn1)# pull
adm@DionisNX(config — if—vpn1)# ca ca.crt
adm@DionisNX(config — if—vpn1)# cert client.crt
adm@DionisNX(config —if—vpn1)# key client.key
```

20.1 SVPN-интерфейс

Для создания svpn-интерфейса используется команда: interface svpn <номер> из режима configure. При этом номер интерфейса может начинаться с 0.

```
adm@DionisNX(config)# interface svpn 0
adm@DionisNX(config — if— svpnO)#
```

Режимы работы интерфейса svpn:

1. Сервер для туннеля с TLS-аутентификацией,
2. Мульти-клиент-сервер.

Команды для настройки svpn-интерфейса

команда	параметр
proto <Протокол>	Протокол работы интерфейса. По умолчанию udp. Должен совпадать с протоколом на удаленном конце туннеля
port <Номер порта>	Номер порта. По умолчанию 1194
local <ip или имя хоста>	Локальный ip или имя хоста (Необязательный параметр)
bind	Команда связывает локальный адрес и порт
server <ip сети:маска сети>	IP-адрес и маска создаваемой частной сети.(Только для режима работы Мульти-клиент-сервер)
remote <ip или имя хоста>	Удаленный ip или имя хоста, к которому будет происходить подключение (Режим серве- ра с TLS-аутентификацией)
ca <Корневой сертификата	Предварительно импортированный корневой сертификат
cert cСертификат сервера>	Предварительно импортированный сертификат сервера
key <Ключ сервера >	Предварительно импортированный ключ сертификата сервера
dh <Ключ Диффи-Хеллмана>	Предварительно импортированный ключ Диффи-Хеллмана
tls-auth <Дополнительный ключ>	Предварительно импортированный tls-auth-ключ. Данная команда добавляет дополнительный слой аутентификации, tls-auth-файл должен быть такой же, как на клиенте (Необязательный параметр)
ifconfig <l_IP:r_IP>	Приватный адрес локального и удаленного конца туннеля. (Используется только для режима сервера в туннеле с TLS-аутентификацией)
cipher <Алгоритм>	Алгоритм шифрования.

команда	параметр
auth <Алгоритм>	Алгоритм Аутентификации.
ping cИнтервал в секундах>	ping удаленного конца, если нет передачи пакетов в течение промежутка времени, большего чем указанный интервал
ping-restart cИнтервал в секундах>	Перезагрузка подключения к удаленному концу туннеля, если от него не приходило пакетов в течение промежутка времени, большего чем указанный интервал
push ping cИнтервал в секундах>	Установка значения ping для подключаемых клиентов
push ping-restart cИнтервал в секундах>	Установка значения ping-restart для подключаемых клиентов
push route cip сети:маска сети>	Передача клиенту маршрутов, чтобы позволить ему связаться с другими частными подсетями
ns-cert-type <nsCertType>	Требовать <nsCertType> в поле nsCertType сертификата соседа
tls-cipher cAlg>	Алгоритм tls-шифра. Используется для повышения уровня безопасности контроля канала управления (TLS used only as a control channel).
client-to-client	Команда позволяет подключенным клиентам видеть друг друга
duplicate-cn	Команда позволяет подключаться нескольким клиентам с одинаковым common name в поле сертификата
client-net cCommon Name>	Добавление подсети клиента с Common Name в поле сертификата

Примечание:

Команда

adm@DionisNX(config — if—svpn0)# server 10.8.0.0/24

означает, что интерфейсу svpn0 будет назначен адрес 10.8.0.1, а подключаемым клиентам адреса с 10.8.0.4 по 10.8.0.251

Ниже приводятся примеры конфигурации интерфейса для различных режимов работы. (В данных примерах настраиваемый интерфейс является локальным концом туннеля).

1. Сервер для туннеля с TLS-аутентификацией.

В данном режиме необходимо указать имя удаленного хоста или его ip-адрес, а так же локальный и удаленный ip туннеля. Пример настройки интерфейса:

adm@DionisNX(config — if—svpn0)# remote 192.168.33.232
adm@DionisNX(config —if—svpn0)# ifconfig 10.8.0.1:10.8.0.2
adm@DionisNX(config —if—svpn0)# ca ca.crt
adm@DionisNX(config — if—svpn0)# cert server.crt

adm@DionisNX(config — if— svpnO)# key server.key
adm@DionisNX(config —if—svpnO)# dh dhl024.key

2. Мульти-клиент-сервер

adm@DionisNX(config — if— svpn1)# server 10.8.0.0/24
adm@DionisNX(config — if— svpn1)# ca ca.crt
adm@DionisNX(config — if— svpn1)# cert server.crt
adm@DionisNX(config — if— svpn1)# key server.key
adm@DionisNX(config — if— svpn1)# dh dhl024.key
adm@DionisNX(config — if— svpn1)# ping 10
adm@DionisNX(config — if— svpn1)# ping —restart 120
adm@DionisNX(config —if—svpn1)# push ping 10
adm@DionisNX(config — if— svpn1)# push ping —restart 60

Включение нескольких машин на стороне клиента.

Допустим, что локальная сеть клиента использует адреса 192.168.4.0/24 и что VPN-клиент использует сертификат с common name = Users.

Для включения этой сети на сервере необходимо выполнить следующие команды:

adm@DionisNX(config — if— svpn1)# client—net Users
adm@DionisNX(config —if—svpn1 —Users)# iroute 192.168.4.0/24

Если требуется, чтобы другие клиенты могли видеть данную подсеть, необходимо также выполнить команды:

adm@DionisNX(config — if— svpn1)# client—to—client
adm@DionisNX(config —if—svpn1)# push route 192.168.4.0/24

Если требуется назначить конкретный ip-адрес клиенту, необходимо выполнить следующие команды (в данном примере VPN-клиент использует сертификат с common name = Users. Требуется назначить ip 10.8.0.113):

adm@DionisNX(config — if— svpn1)# client—net Users
adm@DionisNX(config — if— svpn1 — Users)# ifconfig —push 10.8.0.113/24

25. Экспорт статистики по Netflow

Экспорт статистики по протоколу Netflow предоставляет возможность анализа сетевого трафика на уровне сеансов, делая запись о каждой транзакции TCP/IP. Dionis-NX может выступать в роли сенсора и передавать информацию на коллектор для выбранных интерфейсов. Поддерживаемые версии протоколов: 3, 5, 9.

Для того чтобы активировать экспорт статистики, необходимо для всех выбранных интерфейсов включить flow cache с помощью команды ip flow:

```
DionisNX(config)# interface ethernet 0
DionisNX(config — if—ethernet0)# ip flow
```

Затем, нужно войти в режим конфигурации ip flow-export и настроить параметры передачи по Netflow:

```
DionisNX(config)# ip flow—export
DionisNX(config — ip—flow—export)#
```

Ниже перечислены все параметры конфигурации flow-export:

destination <ip> <port>	Указать адрес коллектора
no ip destination	Удалить адреса коллектора
version <31519>	Задать версию протокола Netflow
no version	Версия протокола Netflow по умолчанию
max-flows <число>	Максимальное число потоков
no max-flows	Значение по умолчанию 8192
hoplimit <ttl>	Значение TTL
no hoplimit	По умолчанию 1
level <full | proto| ip>	Уровень информации
no level	По умолчанию full
timeout <general|tcp|tcp.rst|tcp.fin|udp|maxlife|> <интервал>	задать временной интервал
[no] timeout <general|tcp|tcp.rst|tcp.fin|udp|malife>	установить значение по умолчанию для временного интервала
enable	Включить экспорт
disable	Выключить экспорт

Основными настройками являются:

- destination;
- version.

После настройки, необходимо активировать экспорт командой enable, например:

```
DionisNX(config)# ip flow—export
DionisNX(config — ip—flow—export)# destination 192.168.0.254 8854
DionisNX(config — ip—flow—export)# version 9
DionisNX(config — ip—flow—export)# enable
```

Для диагностики работы flow кэша (кэша статистики соединений), можно воспользоваться командой show interface:

```
DionisNX# show interface ethernet 0 flow dump
```

26. Служба NTP

В Dionis-NX реализована служба синхронизации времени. Данная служба может получать информацию о точном времени от других NTP-серверов, а также может являться NTP-сервером для других узлов.

Чтобы войти в режим настройки службы NTP, нужно выполнить команду в режиме конфигурации:

(config)# service ntp

Далее нужно добавить имена (IP-адреса)серверов времени, с которыми будет осуществляться синхронизация. Пример:

(config—service—ntp)# server 1.1.1.1 (config—
service—ntp)# server myntpserver (config—
service—ntp)# servers 0.ru.pool.ntp.org

Команды «server» и «servers» отличаются только тем, что если FQDN-имя, указанное командой «servers», разрешается в несколько IP-адресов пула NTP-серверов, то будут делаться попытки осуществлять синхронизацию со всеми этими серверами. В случае команды «server» - только с первым IP-адресом, в который разрешилось FQDN.

Список серверов является упорядоченным. Серверы в начале списка имеют больший приоритет. Список можно редактировать с помощью команд с числовыми префиксами и команд «по». Пример:

(config—service—ntp)# do show
1 server 1.1.1.1
2 server myntpserver
3 servers 0.ru.pool.ntp.org
(config—service—ntp)# no 2
(config—service—ntp)# do show
1 server 1.1.1.1
2 servers 0.ru.pool.ntp.org (config—service—
ntp)# 1 servers 1.ru.pool.ntp.org (config—
service—ntp)# do show
1 servers 1. ru.pool.ntp.org
2 server 1.1.1.1
3 servers 0.ru.pool.ntp.org
(config—service—ntp)# no all
(config—service—ntp)# do show

По умолчанию служба NTP только синхронизирует время от других серверов и сама не является сервером. Чтобы служба NTP стала сервером, необходимо объявить адрес(а), на котором откроется слушающий сокет. Это делается с помощью команды «listen». Допустимы множественные команды «listen».

Например, следующая команда предписывает принимать NTP-запросы на всех интерфейсах:

(config—service—ntp)# listen 0.0.0.0

Или на некоторых интерфейсах:

(config—service—ntp)# listen 10.1.1.1
(config—service—ntp)# listen 10.1.2.1

«Слушающие» адреса могут быть удалены командой:

(config—service—ntp)# no listen <ip>

Для работы с IPv6 адресами используйте команды: listen6 и no listen6.

После настройки службы NTP её необходимо активировать командой:

(config—service—ntp)# enable

По умолчанию, при выполнении данной команды служба NTP пытается синхронизироваться с NTP-серверами немедленно. Это может вызывать длительные задержки. Если эти задержки недопустимы, то необходимо указать опцию:

(config—service—ntp)# sync off

В этом случае команда «enable» завершится немедленно, и синхронизация времени будет происходить в фоновом режиме.

Чтобы вернуть поведение по умолчанию, необходима опция:

(config—service—ntp)# sync on

Чтобы остановить службу, нужно выполнить команду:

(config—service—ntp)# disable

Если служба NTP находится в активированном состоянии, и при этом осуществляется редактирование её настроек, то для того чтобы они вступили в силу, необходимо перезапустить службу:

(config—service—ntp)# disable
(config—service—ntp)# enable

Чтобы посмотреть журнал службы NTP, нужно выполнить одну из команд привилегированного режима:

show service ntp log
show service ntp log <n>
show service ntp log all
show service ntp log follow

Команды выводят, соответственно: 25 последних строк журнала, <n> последних строк журнала, весь журнал, последние строки журнала и последующие при их появлении (режим слежения).

Чтобы остановить службу и удалить все настройки, нужно выполнить команду режима конфигурации:

(config)# no service ntp

27. Служба DNS

В Dionis-NX реализована возможность настройки службы DNS, обеспечивающая функции по разрешению DNS-запросов.

Для понимания команд настройки необходимо ввести ряд понятий, непосредственно используемых при настройке:

- view (вид): виды позволяют использовать различные настройки сервера при общении с различными наборами узлов; по сути они определяют новый экземпляр сервиса, хотя физически в системе выполняется один процесс сервиса;
- master zone (мастер-зона): DNS-зона, для которой настраиваемый сервер является мастер-сервером, т.е. является первичным уполномоченным сервером;
- slave zone (слэйв-зона): DNS-зона, для которой настраиваемый сервер является вторичным уполномоченным сервером; содержимое зоны считывается от одного из серверов, указанных в опции masters данной слэйв-зоны;
- root zone (корневая зона): описание корневых DNS-серверов; самый новый список корневых серверов можно найти на ftp: //ftp. rs . internic . net/domain/named .root;
- forward zone (форвард-зона): позволяет перенаправить DNS-запросы по зоне другим серверам.

Различают два типа DNS-запросов: рекурсивные и итеративные:

- при рекурсивном запросе сервер имен должен найти информацию самостоятельно. То есть при получении рекурсивного запроса сервер имен при отсутствии у него ответа на запрос должен сам обратиться к помощи других серверов имен, например к корневым серверам (данный запрос будет итеративным). Они сами не дадут ответа, но зато направят на другие DNS-серверы. Сервер имен будет проверять все предоставленные ему ссылки, пока не обнаружит необходимую информацию.
- при итеративном запросе сервер имен должен сразу предоставить ответ, не обращаясь к другим DNS-серверам. Если же данный сервер не может предоставить запрошенную информацию, то он возвратит ссылку на другой сервер имен, который, вероятно, может дать ответ на запрошенную информацию.

Основные концепции сервиса DNS - это иерархичность и наследуемость конфигурационных пространств. Иерархия следующая:

- service dns - сервис DNS (уровень A)

  - view <view-name> - виды (уровень B)
    * zone <master|slave|forward|.> - зоны (уровень C)

Самый верхний уровень конфигурации сервиса - это команды уровня service dns.

Часть этих команд есть также на нижележащих уровнях.

Если некоторая опция будет задана на самом верхнем уровне (A), то она определится и для всех нижележащих уровней. Если же значение этой опции будет изменено на каком-либо нижележащем уровне (например. B), то только на этом уровне значение данной опции будет отличаться от всех остальных.

Замечание по формату описания параметров: запись {<NAME>,3} - означает список из максимум трех параметров типа NAME.

Чтобы войти в режим настройки службы DNS, нужно выполнить команду в режиме конфигурации:

(config)# service dns

27.1 Контроль доступа

acl <NAME> <IPLIST>

Команда acl задает именованный набор IP-адресов.

Далее созданный ACL можно использовать в командах, где параметр имеет тип IPLIST.

NAME - имя набора адресов

IPLIST - набор IP-адресов, имеет следующий формат:

<any | none | localips | localnets | {NAME,8} | {[!]A.B.C.D[/MASK],8} >

- any,none,localips,localnets - имена встроенных наборов:
  - any - любые адреса;
  - none - никакие адреса;
  - localips - IP-адреса, присвоенные интерфейсам системы;
  - localnets - IP-адреса сетей, обслуживаемых интерфейсами системы;
- NAME - это имя другого ACL, ранее созданного;
- [!] A.B.C.D[/MASK]:
  - если в начальной позиции не указан знак »!», то это IP-адрес узла или сети(если указана маска MASK);
  - если в начальной позиции указан знак »!», то это любые адреса,кроме указанных после знака »!».

Примеры:

(config—service—dns)# acl al 1.2.3.0/24
(config—service—dns)# acl a2 al 1.2.3.1
(config—service—dns)# acl a3 a2 5.5.5.5
110.0.0.0/24

allow query <IPLIST>

Определяет, каким именно узлам разрешено выполнять ДНС-запросы через настраиваемый сервер.

По умолчанию: все.

allow query-on <IPLIST>

Определяет, на каких внутренних интерфейсах сервера разрешено принимать ДНС-запросы. По умолчанию: все.

allow query-cache <IPLIST>

Определяет, каким именно узлам разрешено получать ответы на запросы из кэша ДНС настраиваемого сервера.

По умолчанию: все.

allow notify <IPLIST>

Определяет, каким узлам ,помимо первичных, разрешено уведомлять настраиваемый сервер (если он является для зоны вторичным) об изменениях зоны.

По умолчанию: запрещено всем, кроме мастер-серверов зоны.

allow recursion <IPLIST>

Указывает, каким узлам разрешено выполнять рекурсивные запросы через данный сервер.

Блокирование рекурсивных запросов для узла не предотвращает получение этим узлом данных, находящихся в кэше ДНС.

По умолчанию: все.

allow transfer <IPLIST>

Указывает, каким узлам разрешено получать зоны от настраиваемого сервера.

По умолчанию: все.

27.2 Виды

Виды позволяют использовать различные настройки сервера при общении с различными наборами узлов.

Для конфигурации зон необходимо войти в режим вида, даже если функционал видов при настройке не требуется.

Вид - это отдельный экземпляр сервера ДНС со своими настройками и зонами.

На все виды распространяются глобальные опции сервиса.

Чтобы войти в режим настройки вида DNS, нужно выполнить команду в режиме конфигурации службы:

[N] view <VNAME>

Создает вид с именем VNAME и вставляет его в позицию N в списке видов.

Чем меньше N, тем вид приоритетнее, с точки зрения попадания запросов в него.

Входящий DNS-запрос анализируется на предмет соответствия указанным правилам попадания в вид.

Анализ попадания в вид происходит начиная от вида с номером 1 и далее по порядку, пока не будет достигнут вид с максимальным номером.

Если запрос попал в какой-либо вид, то он будет обслуживаться этим видом; далее поиск вида, в который может попасть запрос, прекращается, даже если далее будут виды, в которые данный запрос может попасть.

Пример:

(config—service—dns)# view vname

Рассмотрим команды вида, позволяющие указать, какие запросы должны в него попадать.

match clients <IPLIST>

Опция указывает, что данный вид будет обслуживать только тех клиентов, IP-адреса которых входят в IPLIST.

match destinations <IPLIST>

Опция указывает, что данный вид будет обслуживать только тех клиентов, которые обращаются к ДНС-серверу по IP-адресам, входящим в IPLIST.

match recursive-only <yes | no>

Если значение yes, то данный вид будет обслуживать только рекурсивные запросы, иначе - все запросы.

По умолчанию: match recursive-only no.

auto-local-zones

Создает мастер-зоны O.in-addr.arpa., 127.in-addr.arpa., 255.in-addr.arpa. и localhost. с параметрами по умолчанию.

Эта команда значительно упрощает конфигурацию службы.

В случае ее задания будет невозможно определить зоны с вышеуказанными именами.

Примеры

(config—dns—vname)# match clients 1.2.3.0/24 (config—dns—vname)# match destinations 192.168.0.1 192.168.0.2 (config—dns—vname)# match recursive—only yes

27.3 Зоны

DNS-записи, т.н. ресурсные записи, хранятся в зонах. Служба поддерживает 4 типа DNS зон:

- мастер-зона: зона, для которой настраиваемый сервер является первичным уполномоченным сервером.
- слэйв-зона: зона, для которой настраиваемый сервер является вторичным уполномоченным сервером;
- корневая зона: описание корневых ДНС-серверов; в это описание можно добавлять любые другие записи, кроме soa-записи.
- зона ретрансляции: позволяет перенаправить DNS-запросы по зоне другим серверам.

Создание ресурсных записей возможно только для мастер-зоны и корневой зоны. Причем для корневой зоны не поддерживается ресурсная запись SOA.

Имена зон должны оканчиваться точкой.

Рассмотрим далее команды создания зон, а также команды, специфичные именно для зон данного типа.

Создание любой зоны возможно только в режиме конфигурации вида. Допустим, создан вид с именем vname.

При создании ресурсных записей используются три основных типа данных: IP-адрес, число, имя домена.

Первые два типа данных не нуждаются в пояснении. Особое внимание следует обратить на третий тип данных - имя домена. Можно задать имя домена в двух формах:

- имя оканчивается точкой: это абсолютное имя, полностью задающее домен и зону, в которой он находится;
- имя не оканчивается точкой: это относительное имя домена; такой домен рассматривается как домен зоны, в которой он описан(через ресурсную запись зоны).

Чтобы получить полное имя домена из относительного, нужно справа от относительного имени приписать имя зоны, в которой домен описан (без точки), и завершить полное имя точкой.

Пользуясь приведенным правилом, рассмотрим примеры:

- для зоны «zeta.» ресурсная запись a 1.1.1.1 domain ns задает IP-адрес 1.1.1.1 для домена «ns.zeta.»;
- для корневой зоны ».» ресурсная запись a 2.2.2.2 domain ns задает адрес 2.2.2.2 для домена «ns.».

zone master <ZNAME>

Создает мастер-зону с именем ZNAME и входит в режим конфигурации зоны.

В мастер-зоне можно задавать любые ресурсные записи. Команды создания ресурсных записей рассмотрены ниже.

update [PAS]

Включить динамическое обновление A- и PTR- записей для зоны. Параметр PAS - необязательный пароль, который можно использовать для организации удаленного обновления. Более подробно о настройке динамического обновления см.**Динамическое обновление зон .**

zone slave <ZNAME>

Создает слэйв-зону с именем ZNAME и входит в режим конфигурации зоны.

Ресурсные данные для зоны будут автоматически получаться с мастер-серверов, когда зонные данные изменяются (режим нотификации) и когда истекает TTL записи БОА-зоны.

masters {<IP[:port]>,5}

Позволяет задать максимум 5 серверов, являющихся первичными для данной зоны.

Зонную информацию служба будет получать от этих мастер-серверов.

Пример:

(config—dns—vname—slv—zeta.)# masters 1.2.3.4 2.3.4.5

zone forward <ZNAME>

Создает зону ретрансляции с именем ZNAME и входит в режим конфигурации зоны.

Эта зона, все запросы по которой сразу же обслуживаются указанными в данной зоне ретрансляторами, т.е. другими серверами имен.

Единственная команда зоны - это forwarders. Она уже рассмотрена в разделе Другие настройки.

Данная команда определяет список ретрансляторов, т.е. серверов, которым будут переназначаться запросы DNS.

Переназначенный ретранслятору запрос является рекурсивным, т.е. ожидается, что ретранслятор вернет окончательный ответ.

По умолчанию в службе DNS разрешена рекурсия, т.е запросы службы к другим северам имен являются итеративными, т.е. служба DNS сама будет производить поиск окончательного ответа.

Данная команда может быть выполнена в любом типе зоны, кроме корневой, а также на глобальном уровне и уровне видов.

Алгоритм работы службы различается в зависимости от того, на какой тип зоны распространяется данная команда:

- если запрос попадает в авторитетную зону (мастер- или слэйв-) настроенную на ретрансляцию:
  - сначала ответ ищется в кэше или данных зоны;
  - если ответ не найден - посылается рекурсивный запрос ретранслятору;
  - если ответ от ретранслятора не пришёл - сервером начинается самостоятельный итеративный поиск ответа;
- если запрос попадает в зону ретрансляции, то он сразу же обслуживается указанными в данной зоне ретрансляторами.

Список ретрансляторов может быть пустым. Это полезно,если набор ретрансляторов установлен для всех зон глобально, но для одной из зон не нужно использовать ретрансляторы.

Пример:

(config—dns—vname—fwd—zeta.)# forwarders 1.2.3.4 2.3.4.5 only

zone .

Позволяет задать корневую зону.

auto [[[daily | weekly | monthly] [SRV]] | static]

Команда автоматического обновления корневой зоны.

Параметры: [5]

5 daily | weekly | monthly - частота (ежедневно, еженедельно или ежемесячно) загрузки зонных данных для корневой зоны из внешней сети по URL: ftp://ftp.rs.internic.net/domain/db.cache;

- SRV - имя домена ftp-сервера или его IP-адрес; путь для загрузки зонных данных при использовании этого параметра будет выглядеть следующим образом: ftp: //SRV/domain/db. cache;
- static - используется корневая зона по-умолчанию, датированная июнем 2011 года, взятая с вышеуказанного URL. Периодического обновления в данном случае не происходит.

Зонный файл для корневой зоны может быть получен из внешней сети по URL: ftp://ftp. rs . internic. net/domain/db. cache и далее обновлен с указанной периодичностью: ежедневно, еженедельно (по умолчанию) или ежемесячно.

Команды создания RR-записей игнорируются в случае наличия команды auto, кроме auto static. В последнем случае к данным корневой зоны по умолчанию добавляются любые дополнительные ресурсные записи, кроме БОА-записи.

Параметры команды по умолчанию: auto weekly ftp.rs.internic.net

27.3.1 Команды создания ресурсных записей

Команды создания ресурсных записей имеют смысл только для мастер-зоны и корневой зоны (в режиме auto static).

ttl <N>

Команда указывает время жизни по умолчанию для всех записей зоны.

По умолчанию: 86400

soa [master <MNAME> admin <EMAIL> ttl <TTL> refresh <NRF> retry <NRT> expire <EXP> negttl <NEGTTL>]

Команда создает заголовочную SOA-запись.

Необязательные параметры:

- MNAME - имя мастер-сервера зоны; по умолчанию - имя зоны;
- EMAIL - почтовый адрес администратора зоны; по умолчанию - root@<имя зоны>;
- TTL - время жизни записи; если не указано, используется значение ttl для зоны;
- NRF - период обновления зоны вторичным сервером; по умолчанию - 21600 сек;
- NRT - период повторной попытки обновления зоны вторичным сервером; по умолчанию - 1800 сек;
- EXP - интервал устаревания данных зоны для вторичного сервера; по его истечении данные, содержащиеся в зоне, не будут использоваться для ответов на запросы; по умолчанию - 1209600 сек;
- NEGTTL - время жизни отрицательных (неуспешных) ответов в кэше; по умолчанию - 1000 сек.

a <IP> [domain <NAME>] [ttl <N>]

Команда создает адресную A-запись.

Обязательные параметры:

- IP-адрес.

Необязательные параметры:

- имя домена NAME; если не указан,используется имя текущей зоны;
- время жизни записи N; если не указано, используется значение ttl для зоны.

aaaa <IP6> [domain <NAME>] [ttl <N>]

Команда создает адресную AAAA-запись.

Обязательные параметры:

- IPv6-адрес.

Необязательные параметры:

- имя домена NAME; если не указан, используется имя текущей зоны;
- время жизни записи N; если не указано, используется значение ttl для зоны.

cname <ANAME> <CNAME> [ttl <N>]

Команда создает алиасную CNAME-запись.

Обязательные параметры:

- имя алиаса ANAME;
- каноническое имя CNAME для алиаса ANAME.

Необязательные параметры:

- время жизни записи N; если не указано, используется значение ttl для зоны; **mx <NAME> <prio> [domain <NAME>] [ttl <N>]**

Команда создает почтовую MX-запись.

Обязательные параметры: [6]

6 имя почтового ретранслятора NAME;

- приоритет почтового ретранслятора.

Необязательные параметры:

- имя домена NAME; если не указано, используется имя текущей зоны;
- время жизни записи N; если не указано, используется значение ttl для зоны.

ns <NSNAME> [domain <NAME>] [ttl <N>]

Команда создает NS-запись.

Обязательные параметры:

- NSNAME - имя сервера имен.

Необязательные параметры:

- имя домена NAME; если не указано, используется имя текущей зоны;
- время жизни записи N; если не указано, используется значение ttl для зоны.

ptr <ARPANAME> <FULLNAME> [ttl <N>]

Команда создает обратную адресную PTR-запись.

Обязательные параметры:

- ARPANAME - поддомен текущей обратной зоны;
- FULLNAME - каноническое(полное) имя домена для ARPANAME.

Необязательные параметры:

- время жизни записи N; если не указано, используется значение ttl для зоны.

27.4 Другие настройки

Данные настройки задаются на разных уровнях службы. Для каждой команды область, где может быть вызвана команда, указана в строке Область определения.

forwarders [{<IP[:PORT]>,3}] [first | only]

Определяет список ретрансляторов, т.е. серверов, которым мы переназначаем запросы ДНС.

Переназначенный ретранслятору запрос является рекурсивным, т.е. ожидается, что ретранслятор вернет окончательный ответ.

По умолчанию,если это не изменено настройками сервиса,запросы серверам, не являющимся ретрансляторами, будут итеративными, т.е. такой сервер ДНС сам будет производить поиск окончательного ответа.

Данная опция может быть в любом типе зоны.

Если запрос попадает в авторитетную зону (мастер или слэйв) настроенную на ретрансляцию: [7]

7 сначала ответ ищется в кэше или данных зоны;
- если ответ не найден - посылается рекурсивный запрос ретранслятору;
- если ответ от ретранслятора не пришёл - сервером начинается самостоятельный итеративный поиск ответа.

Существует специальный тип зоны, форвард-зона или зона ретрансляции, все запросы по которой сразу же обслуживаются указанными в данной зоне ретрансляторами.

Список ретрансляторов может быть пустым. Это полезно, если набор ретрансляторов установлен для всех зон глобально, но для одной из зон использовать ретрансляторы не нужно.

Параметры:

- IP: port - адрес ретранслятора и порт(необязательно), на который следует пересылать запрос;
- first: если ответа в данных зоны нет, следует пересылать запрос ретрансляторам и, если ответа нет, попытаться самостоятельно разрешить запрос (по умолчанию);
- only: если ответа в данных зоны нет, то для разрешения запроса следует использовать только ретрансляторы.

Область определения: служба, вид, зона ретрансляции.

notify <yes | no | master-only | explicit> [also-notify {<IP:PORT>,3}]

Определяет, нужно ли слать нотификацию (сообщение DNS NOTIFY) вторичным серверам зоны, для которой изменился идентификатор (поменялся ее SOA ID).

Вторичные сервера зоны выбираются из NS-записей зоны.

Дополнительно, параметром also-notify можно указать адреса и порты дополнительных серверов, которым следует слать нотификации.

В случае указания опции explicit, нотификации не шлются никому, кроме списка серверов, указанных параметром also-notify.

Область определения: служба,вид,мастер-зона.

По умолчанию: notify yes.

query-source [IP] [PORT_START [PORT_END]]

Определяет адрес и порт сервера для посылки запросов другим DNS-серверам.

Эта опция используется, если DNS-сервер должен работать с определённым локальным сетевым интерфейсом для отправки запросов, в случае, например, если один из основных DNS-серверов опознает лишь один из его многочисленных адресов.

Указанный IP-адрес будет использован и для TCP-,и для UDP-запросов.

Указанный порт(порты) будет использован только для UDP-запросов, а для TCP будет выбран случайный непривилегированный порт (>1024).

Предупреждение: небезопасно назначать фиксированный порт, т.к. злоумышленник может угадать 16-битный DNS Transaction ID и замусорить кэш сервера своими неадекватными ответами. В случае же случайного порта - ему необходимо угадать два 16-битных числа (порт и id транзакции), что гораздо сложнее.

Область определения: служба.

По умолчанию: query-source 0.0.0.0 и случайный непривилегированный порт (номер порта больше 1000)

Для работы с IPv6 адресами используйте команды: query-source6 и no query-source6.

listen [PORT] [IPLIST]

Установка номера порта и IP-адресов, на которых будет слушать ДНС-сервер.

Область определения: служба.

По умолчанию: listen 53 localips.

Для работы с IPv6 адресами используйте команды: listen6 и no listen6.

recursion <yes | no >

Нужно ли обслуживать рекурсивные запросы?

Область определения: служба,вид.

По умолчанию: recursion yes.

27.5 Динамическое обновление зон

Для мастер-зоны можно использовать динамическое обновление. Динамические обновления позволяют локальной или удаленной службе DHCP обновлять A- и PTR-записи в мастер-зоне. Динамическое обновление зоны может быть двух типов:

- локальное: осуществляется в связке со службой DHCP, работающей на той же машине, что и настраиваемая DNS-служба;
- удаленное: осуществляется любым удаленной DHCP-службой Dionis-NX, отдельным DHCP-сервером или другим удаленным ПО, которое может осуществлять динамические DNS-обновления.

В качестве удаленного DHCP-сервера наиболее совместимым с Dionis-NX службами является сервер ISC DHCP.

В обновляемых зонах могут быть различные статические записи.

Если все настроено правильно, то при выдаче службой DHCP IP-адреса из подсети, которой соответствует настроенная на обновление обратная зона в службе DNS, и доменное имя данной подсети соответствует настроенной прямой зоне, будет происходить автоматическое обновление A- и PTR- ресурсных записей.

27.5.1 Локальное динамическое обновление

Для настройки локального автоматического обновления необходимо:

- сначала настроить DNS-службу:
  - настроить прямую и обратную зоны, подлежащие авто-обновлению;
  - если есть опция allow query для зоны, подлежащей обновлению, среди аргументов этой команды должен быть localhost;

- предусмотреть, чтобы динамические обновления попали в нужный вид (тот, в котором содержится обновляемая зона):
  * в виде, в котором находится обновляемая зона, среди аргументов команды match clients должен быть localhost;
  * в видах, которые расположены выше вида, в котором находится обновляемая зона, среди аргументов команды match clients не должно быть localhost;
- добавить в обновляемую зону команду update;

• затем настроить DHCP-службу:

- настроить интервал раздаваемых адресов (range) таким образом, чтобы он включал в себя подсеть обновляемой обратной зоны, либо был равен этой подсети.
- указать domain-name для подсети, либо глобально;
- запустить или перезапустить DHCP-службу.

Пример:

```
#отрывок DNS—конфигурации:
view vl
  match clients localhost 192.168.33.0/24
  zone master 33.168.192.in—addr.arpa.
  update
   soa master raul.cuba.int.
   ns raul.cuba.int.
   ptr 254 raul.cuba.int.

  zone master cuba.int.
   update
   soa master raul
   a 192.168.33.254 domain raul
   ns raul

view default


#отрывок DHCP—конфигурации:
domain—name cuba.int
subnet 192.168.33.0/24
range 192.168.33.10 192.168.33.200
```

27.5.2 Динамическое обновление DNS Dionis-NX удаленной DHCP-службой

Для настройки удаленного автоматического обновления DNS удаленной DHCP-службой Dionis-NX необходимо: [8]

8 сначала настроить локальную DNS-службу:

- настроить прямую и обратную зоны, подлежащие авто-обновлению;
- если есть опция allow query для зоны, подлежащей обновлению, то в список IP-адресов этой команды должен входить адрес удаленной DHCP-службы;
- предусмотреть,чтобы динамические обновления попали в нужный вид (тот, в котором содержится обновляемая зона):
  * в виде, в котором находится обновляемая зона, в списке IP-адресов команды match clients должен быть адрес удаленной DHCP-службы;
  * в видах, которые расположены выше вида, в котором находится обновляемая зона, в списке IP-адресов команды match clients не должен быть адрес удаленной DHCP-службы;
- добавить в обновляемую прямую и обратную зоны команду update PAS, где PAS - некоторая строка-пароль, которые могут быть разными для этих зон.

• затем настроить удаленную DHCP-службу:

- настроить интервал раздаваемых адресов (гапде)таким образом, чтобы он включал в себя подсеть обновляемой обратной зоны, либо был равен этой подсети;
- указать domain-name для подсети, либо глобально;
- добавить команду update <DNSIP> <FZONE> <PAS>, где DNSIP - IP-адрес DNS-сервера, FZONE - имя прямой зоны, PAS - тот же пароль, что указан в опциях команды update прямой зоны в DNS-службе;
- добавить команду update <DNSIP> <RZONE> <PAS>, где DNSIP - IP-адрес DNS-сервера, RZONE - имя обратной зоны, PAS - тот же пароль, что указан в опциях команды update обратной зоны в DNS-службе;
- запустить или перезапустить DHCP-службу.

Пример:

```
 #отрывок DNS—конфигурации:
view vl
   match clients localhost 192.168.33.0/24
   zone master 33.168.192.in—addr.arpa.
   update pas
    soa master raul.cuba.int.
    ns raul.cuba.int.
    ptr 254 raul.cuba.int.

   zone master cuba.int.
    update pas
    soa master raul
    a 192.168.33.254 domain raul
    ns raul

 view default


 #отрывок DHCP—конфигурации удаленной DHCP—службы:
domain—name cuba.int
update 192.168.33.254 cuba.int. pas
```

update 192.168.33.254 33.168.192.in—addr.arpa. pas
subnet 192.168.33.0/24
range 192.168.33.10 192.168.33.200

27.5.3 Динамическое обновление удаленным DHCP-сервером

Для настройки удаленного автоматического обновления DNS удаленным DHCP-сервером, либо другим ПО, которое может осуществлять удаленные DNS-обновления необходимо:

- сначала настроить локальную DNS-службу: осуществляется аналогично настройке удаленного автоматического обновления DNS удаленной DHCP-службой Dionis-NX;
- после выполнения команды update PAS на экран будет выведена информация об общем ключе и его параметрах, например: key(name:dnskey,secret:ABC123);
- использовать эту информацию для настройки ISC DHCP-сервера.

Пример:

#1. отрывок DNS—конфигурации:

```
view vl
   match clients localhost 192.168.33.0/24
   zone master 33.168.192.in—addr.arpa.
   update pas
    soa master raul.cuba.int.
    ns raul.cuba.int.
    ptr 254 raul.cuba.int.

   zone master cuba.int.
    update pas
    soa master raul
    a 192.168.33.254 domain raul
    ns raul

  view default
```

#2. После выполнения команды update pas на экран будет выведено:

* для прямой зоны: To update A RRs by remote dhcp servers use key (name:pas,key:cGFz)
* для обратной зоны: To update PTR RRs by remote dhcp servers use key (name:pas,key:cGFz)

#3. В конфигурацию удаленного ISC DHCP сервера добавьте следующие строки:

```
ddns—update—style interim;

key pas {
  algorithm hmac—md5;
```

```
    secret "cGFz";
};

zone cuba.int. {
    primary 192.168.33.254;
    key pas;
};
zone 33.168.192.in—addr.arpa. {
    primary 192.168.33.254;
    key pas;
};
```

27.6 Ограничения ресурсов службы

Значения указываются в байтах,

limit journal-size <N>

Устанавливает максимальный размер отдельного файла журнала.

Файл журнала автоматически создается для динамически обновляемой зоны (см. команду update в конфигурировании мастер-зоны) и содержит информацию об обновлении зоны в бинарном формате.

Информация из журнала автоматически переносится в зонный файл, а журнал удаляется в следующих случаях:

- автоматически примерно каждые 15 минут;
- по команде перезагрузки зон: «service dns relod zones <all | dynamio»;
- по команде запуска сервиса: enable.

При достижении указанного максимального размера файла самые старые транзакции журнала удаляются автоматически.

Умолчание: неограниченно.

limit recursive-clients <N>

Максимальное количество одновременно выполняемых рекурсивных запросов, поступивших от клиентов.

Для справки: каждый рекурсивный запрос потребляет примерно 20Кб памяти.

Умолчание: 1000.

limit tcp-clients <N>

Максимальное количество TCP-соединений, поддерживаемых сервером в каждый момент времени.

Умолчание: 100.

limit cache-size <N>

Максимальный объем памяти, отводимой под кэш DNS-сервера для каждого вида, в байтах.

Когда объем данных кэша достигает этого предела, DNS-сервер принудительно удаляет записи из кэша.

Умолчание: неограниченно.

limit tcp-listen-queue <N>

Устанавливает число (не меньше 3) TCP-соединений в ядре системы, ожидающих передачи данных.

Умолчание: 3.

limit max-cache-ttl<N>

Устанавливает TTL хранения ответов в кэше.

Умолчание: 7 дней,

limit max-ncache-ttl<N>

Устанавливает TTL хранения отрицательных ответов в кэше.

Умолчание: 3 часа,

filter-aaaa

Удалять все AAAA-записи из ответа, предназначенного для 1Ру4-клиента (запрос в DNS-службу пришел с 1Ру4-адреса).

27.7 Журналы

log <TYPE> <LEVEL>

Определяет тип и уровень журналирования.

Параметр TYPE задает тип информации для журналирования: [9]

---

9 all-first : любой тип информации; логически расположен в начале списка log-команд;
- all-last : любой тип информации; логически расположен в конце списка log-команд;
- other : остальная информация, не входящая в следующие типы;
- config : информация, связанная с внутренним конфигурационным файлом сервера;
- queries : информация, связанная с DNS-запросами серверу;
- transfer : информация, связанная с передачей зоны;
- update : информация, связанная с обновлением зон.

Параметр LEVEL задает уровень подробности журналируемой информации:

- попе : ведение журнала отключено;
- critical : вести журнал только критических ошибок;
- error : вести журнал обычных ошибок и более серьезных;
- warning : вести журнал предупреждений и более серьезных событий;
- notice : вести журнал замечаний и более серьезных событий;

- info : вести журнал информационных сообщений и более серьезных событий;
- debug <N> : вести журнал отладочных сообщений и более серьезных событий; N - уровень отладки от 1 до 3х.

Пояснение: типы all-last/first нужны для удобства.

- тип all-first: допустим, задано журналирование ВСЕЙ информации с уровнем ERROR с помощью all-first, а затем необходимо указать журналирование только UPDATE-информации с уровнем INFO, при этом остальную ифнормацию оставить на уровне ERROR.
- тип all-last: допустим, задано множество команд журналирования - для каждого типа информации свой уровень; а затем стало необходимым для всех типов информации задать один уровень, при этом сохранив старые настройки (для каждого типа свой уровень); в этом случае следует использовать all-last.

27.8 Диагностика

Рассмотрим команду режима enable для диагностики работы службы DNS.

nslookup <DOMAIN> [TYPE] [rr <RRTYPE>] server <SERVER>

Команда осуществляет DNS-запрос.

Параметры:

- DOMAIN : IP-адрес, либо абсолютное доменное имя хоста, т.к. команда не использует ip resolver search/domain-name;
- TYPE : рекурсивный (recursive) или нерекурсивный (non-recursive) запрос; по умолчанию - рекурсивный;
- RRTYPE : тип ресурсной записи (mx,ptr,cname,a,soa,ns); по умолчанию - все типы ресурсных записей;
- SERVER : сервер,на который посылать запрос; по умолчанию - серверы, указанные в ip resolver nameserver;
- PORT : порт, на который посылать запрос; по умолчанию - 53.

Формат вывода команды (рассмотрим пример):

```
;; — >>HEADER<<— opcode: QUERY, status: NOERROR, id: 25210
;; flags: qr rd ra; QUERY: 1, ANSWER: 8, AUTHORITY: 2, ADDITIONAL: 1

;; QUESTION SECTION:
;ya.ru.                  IN   A

;; ANSWER SECTION:
ya.ru.            3259 IN A 87.250.250.203
```

```
;; AUTHORITY SECTION:
ya.ru.          3233    IN  NS    ns5.yandex.ru.


;; ADDITIONAL SECTION:
nsl.yandex.ru. 124 IN AAAA 2a02:6b8::l


;; Query time: 1 msec
;; SERVER: 192.168.33.254#53(192.168.33.254)
;; WHEN: Fri Dec 2 15:59:48 2011
;; MSG SIZE rcvd: 222
```

Рассмотрим более подробно,что означают эти данные в выводе:

- **opcode** : тип операции (QUERY - запрос);
- **status** : статус операции:
  - NO ERROR - нет ошибок;
  - SERVER FAILURE - ошибка сервера;
  - NXDOMAIN - ошибка в имени, нет такого имени;
  - NOT IMPLEMENTED - реализация отсутствует;
  - REFUSED - отказ;
- **id** : 16-битный номер ДНС-транзакции, нужный для связи запросов и ответов;
- **flags** : дополнительные сведения об ответе, могут быть следующими:
  - qr - сообщение является ответом, а не запросом; присутствует всегда;
  - aa - авторитетный ответ;
  - rd - послано требование обслужить наш запрос рекурсивно;
  - ra - требование рекурсии удовлетворено, т.к. рекурсивные запросы разрешены на сервере имен.
- **Число записей:**
  - QUERY : число записей в разделе запроса;
  - ANSWER : число записей в разделе ответа;
  - AUTHORITY : число записей в разделе авторитета;
  - ADDITIONAL : число записей в дополнительном разделе;
- **Записи:**
  - QUESTION SECTION : раздел запроса, содержит записи, которые необходимо получить от сервера;
  - ANSWER SECTION : раздел ответа, содержит записи, полученные в ответе;
  - AUTHORITY SECTION : раздел авторитета, содержит имена серверов, авторитетных для запрашиваемого домена;

- ADDITIONAL SECTION : дополнительный раздел, содержит дополнительные записи, по смыслу связанные с ответной информацией; например A-записи для возвращаемых серверов имен;

• Дополнительная информация о запросе:

- Query time : через сколько времени после отправки запроса удалённый сервер имен вернул ответ;
- SERVER : адрес и порт сервера имен, через который отправлен запрос;
- WHEN : дата и время получения ответа;
- MSG SIZE : размер сообщения 3anpoca(sent) и/или ответа(гсус!) в байтах.

27.6 Работа со службой

В этом подразделе рассматриваются команды режима enable для работы со службой DNS.

27.9.1 Команды управления сервисом

service dns reload

Эта команда может быть полезна если конфигурация службы не поменялась, однако вы хотите чтобы служба заново считала зонные файлы, например потому, что вы поменяли их командой режима enable (например, командой service dns remove dynamic-rrs).

27.9.2 Команды просмотра данных

show service dns cache <only | zones | all> [VIEW] [domain <NAME>]

Показать данные кэша ДНС для вида VIEW по доменному имени NAME.

Если указан параметр NAME - будут показаны только те записи, в которых присутствует домен NAME, иначе - все записи.

Кэш показывается в формате зонных файлов.

Если вид не задан - показываются данные для всех видов.

- only: показать данные кэша;
- zones: показать данные зон, для которых сервер является авторитетным;
- all: показать данные кэша и зон.

show service dns dynamic-rrs [VIEW]

Показывает динамические записи (прямой и обратной зон) вида VIEW.

show service dns log <all | queries | config | transfer | update | other> [all | number <N> | archive <N>] [archive <N>] [follow]

Показывает записи журналов. Типы журналов:

- queries : журналы ДНС-запросов;
- config : журналы работы с внутренним конфигурационным файлом;
- transfer : журналы передачи зон;
- update : журналы обновления;
- other : журналы другой информации, не вошедший в предыдущие типы;
- all : все журналы.

Число записей:

- all : все записи;
- number N : N записей;

- archive N : записи архива журналов под номером N.

Порядок отображения:

- follow: показывать записи журналов по мере их поступления,

show service dns statistic

Показывает статистику ДНС (число запросов, обновлений, данные по сокетам, соединениям и т.д.).

show service dns status

Показывает текущий статус сервиса ДНС (число потоков, число зон, число клиентов в текущий момент и т.д.).

show service dns zones <all | static | dynamio [VIEW [ZONE]]

Показывает зонные данные статических или динамических зон.

В качестве параметров можно указать вид и зону.

27.9.3 Команды удаления данных

service dns remove cache [view VIEW] [name NAME [all]]

Удаляет данные для имени NAME из кэша ДНС. Если указан параметр all - будут удалены все записи, в которых onwer имеет вид *.NAME, т.е. все поддомены домена NAME.

Если имя NAME не задано - удаляет весь кэш.

Если задан вид - удаляет только из этого вида.

service dns remove dynamic-rrs <VIEW> <FZONE> <RZONE> <NAME> <IP> [force]

Удаляет динамическую запись из прямой зоны FZONE и обратной зоны RZONE вида VIEW.

Запись определяется доменным именем NAME (полным или относительным) и IP-адресом IP.

Если зона статическая - запись удалена не будет, только если не будет указан параметр force. В этом случае запись, если она будет найдена, будет удалена.

28. Служба DHCP

Dionis-NX имеет службу DHCP, реализующую серверную часть протокола DHCP (Dynamic Host Configuration Protocol — протокол динамической конфигурации узла).

Протокол DHCP - это сетевой протокол прикладного уровня модели OSI, позволяющий компьютерам автоматически получать IP-адрес и другие параметры, необходимые для работы в сети TCP/IP. Данный протокол работает по модели «клиент-сервер». Для автоматической конфигурации клиент на этапе конфигурации сетевого устройства обращается к серверу DHCP и получает от него нужные параметры.

Основная концепция службы DHCP напоминает концепцию службы DNS - это иерархичность, вложенность и наследуемость конфигурационных пространств.

Иерархия следующая:

- service dhcp - служба DHCP (уровень А)
- subnet/host - подсеть/хост (уровень Б)

Самый верхний уровень конфигурации службы - это команды уровня service dhcp.

Многие из этих команд есть также на нижележащем уровне Б - в описании подсети или хоста.

Если на верхнем уровне (А) задана некоторая опция, то она определяется и для нижележащего уровня (Б). Далее можно изменить на значение данной опции на уровне Б и оно уже будет отличаться от все остальных.

Замечание по формату описания параметров: запись {<NAME>,3} - означает список максимум из трех параметров типа NAME

Чтобы настроить службу DHCP, следует войти в режим ее конфигурации:

service dhcp

Команда осуществляет вход в конфигурацию DHCP-службы.

28.1 Общие настройки службы

listen <ethernet | bond> <N>

Слушать запросы DHCP на Ethernet/Bond интерфейсе N.

Для задания нескольких интерфейсов нужно использовать данную команду для каждого интерфейса.

По умолчанию: слушать на всех широковещательных Ethernet-интерфейсах.

local-port <1M>

Порт, на котором слушать DHCP-запросы.

По умолчанию: 67

local-address <IP>

IP-адрес, на котором слушать не широковещательные DHCP-запросы на порт 67 (если иное не указано командой local-port).

Широковещательные запросы приниматься не будут.

Эта команда может быть полезна, если в связке со службой DHCP используется (на другом узле) служба DHCP-RELAY.

max-lease-time <N>

Максимальный срок аренды адреса (сек.).

Клиент в запросе может прислать желаемое максимальное значение срока аренды адреса.

Опцию можно указывать глобально, в настройках хоста и подсети.

При указании данной опции, клиент не может получить аренду адреса на большее время, чем указано в данной опции.

По умолчанию: 86400

min-lease-time <N>

Минимальный срок аренды адреса (сек.).

Клиент в запросе может прислать желаемое минимальное значение срока аренды адреса.

Опцию можно указывать глобально, в настройках хоста и подсети.

При указании данной опции, клиент не может получить аренду адреса на меньшее время, чем указано в данной опции.

По умолчанию: 300

default-lease-time <N>

срок аренды адреса по умолчанию.

Опцию можно указывать глобально, в настройках хоста и подсети.

Этот срок аренды адреса назначается клиенту, если он не прислал в запросе желаемое максимальное значение.

По умолчанию: 43200

respond-delay <N>

Задает число секунд (от 0 до 255), в течение которых служба будет ждать, прежде чем ответить на запрос клиента.

Эта опция позволяет сделать службу DHCP дублирующей/запасной для основного DHCP-сервера на другом компьютере. Когда основной DHCP-сервер не отозвался, дублирующая служба DHCP будет отзываться после определенного количества запросов клиента, указанного параметром N как продолжительность ожидания перед ответом клиенту. Она может понадобиться для организации дублирующего DHCP-сервера на основе данной службы: если основной сервер DHCP не ответил в течении указанного числа секунд, то клиенту ответит дублирующая служба DHCP Dionis-NX.

По умолчанию: 0

send-hostname <on|off>

Это команда нужна, если в сети есть BOOTP-клиенты (бездисковые рабочие станции), которым необходимо помимо фиксированного адреса присваивать имя хоста.

Включить (on) или отключить (off).

Если опция включена, то для каждого объявления хоста, находящегося в зоне действия опции, имя, использованное в объявлении host (см. ниже подраздел "Настройка статического назначения") передается клиенту в качестве его имени.

Опцию можно указывать глобально и в настройках хоста.

По умолчанию: отключено.

ddns-ttl <1M>

Устанавливает умалчиваемое значение для TTL-динамических записей. Может понадобиться, если будут использоваться динамические DNS-обновления.

По умолчанию: определяется клиентом.

28.2 Настройка статического назначения

Обычно фиксированные адреса назначаются важным хостам сети,например серверам.

Рабочим станциям обычно назначают адрес динамически(см. подраздел "Настройка динамического назначения").

host <NAME>

Вход в режим статического назначения IP-адреса для клиентов.

NAME - имя, идентифицирующее хост; используется только при включённой опции send-hostname: в этом случае оно передается клиенту в качестве его имени.

Переопределяется опцией host-name.

Имя хоста несущественно,если не включена(см.выше) опция send-hostname.

Следует также настроить признаки, по которым служба будет определять, принадлежит ли входящий DHCP-запрос данной host-конфигурации.

mac <MAC>

MAC - MAC-адрес клиента.

Наиболее часто используется привязка хост-декларации к MAC-адресу клиента.

client-id <NAME>

Помимо привязки к MAC-адресу, можно выполнить привязку хост-декларации к идентификатору клиента, который может передаваться в запросе (это т.н. dhcp-client-identifier DHCP-запроса).

NAME - идентификатор клиента.

В результате для поиска host-декларации, соответствующей клиенту,приславшему запрос, происходит следующее:

сначала ищется client-id, совпадающий с dhcp-client-identifier, который прислан клиентом;
если нужный client-id не находится, то ищется совпадение MAC-адреса в команде mac MAC-адресу клиента.

ip <IP>

Задает статический IP-адрес клиента.

28.3 Настройка динамического назначения

Для настройки динамического назначения адресов необходимо описать все сети, обслуживаемые интерфейсами системы, которые будут обслуживать DHCP-запросы.

У этих интерфейсов должны быть назначены IP-адреса с нужными масками сети.

subnet <IP/MASK>

Вход в режим config-service-dhcp-subnet-IP/MASK: динамическое назначение IP-адресов для клиентов.

range <IP_START> [IP_END]

Задает диапазон в сети SUBNET, адреса в пределах которого могут быть наначены клиентам.

Все IP-адреса в диапазоне должны принадлежать той подсети, к описанию которой относится секция range.

Если IP_END не указан, то диапазон состоит из одного адреса.

Пример:

(config-service-dhcp-subnet-192.168.1.0/24)# range 192.168.1.10 192.168.1.210

Этой командой мы предписываем службе назначать клиентам IP-адреса из указанного диапазона (всего 201 адрес).

28.4 Сетевые DHCP-опции

Помимо назначения клиентам IP-адресов, служба может передавать им другую конфигурационную сетевую информацию.

Эта информация называется сетевые DHCP-опции. Они могут быть указаны на любом из уровней службы: как глобально, так и в хост- или сетевой декларациях.

broadcast-address <IP>

Адрес для широковещательных запросов,

domain-name <NAME>

Имя домена для разрешения имен через DNS.

domain-search <NAME>

Имя домена для разрешения имен через DNS.

Можно указать несколько доменов,

gateway <1P>

Адрес шлюза. Можно задавать несколько шлюзов,

subnet-mask <1P>

Маска подсети.

Если не указана, значение маски берется из описания subnet, в которую попадает запрос.

[N] name-server <IP>

IP-адрес сервера имен с приоритетом N.

[N] ntp-server <IP>

IP-адрес сервера времени с приоритетом N.

[N] wins-server <IP>

IP-адрес WINS (NetBios) сервера с приоритетом N.

[N] smtp-server <1P>

IP-адрес сервера SMTP сервера с приоритетом N.

28.5 Пользовательские DHCP-опции

Существует возможность создать пользовательские DHCP-опции, которые будут передаваться указанным клиентам так же, как и стандартные DHCP-опции.

Перед использованием пользовательской опции ее необходимо определить.

user-option-def <NAME> <CODE> <TYPE>

Данная команда определяет новую опцию с именем NAME, имеющую код CODE (от 128 до 254) и тип TYPE.

Код опции CODE принимает значения из интервала 128-254, т.к. все коды меньше 128 зарезервированы под стандартные DHCP-опции. На самом деле интервал от 128 до 224 так же зарезервирован под стандартные DHCP-опции, однако это произошло гораздо позже опубликования стандарта DHCP-протокола в соответствующем документе RFC и не все клиенты могу поддерживать данный стандарт и вполне могут использовать интервал 128-224 как пользовательский интервал опций. Поэтому, если это возможно, рекомендуется использовать интервал кодов 224-254. Интервал 128-224 оставлен для совместимости.

Тип TYPE определяет тип значений опции и может быть следующим:

- bool - задает булевый тип значения опции: on, off
- string - задает строковый тип значения опции: любая текстовая строка
- bytes - задает бинарный тип значения опции: последовательность байт длиной до 128, байты разделены символом »:».
- uint32 - задает целочисленный тип значения опции: любое 32-битное число
- ip - задает тип значения опции в виде IP-адреса.

user-option <NAME> <VAL>

Задает значение VAL ранее определенной опции NAME. Определение опции делается командой user-option-def.

28.6 Связь с DNS (динамическое обновление)

Служба может работать в связке с DNS службой по части динамического обновления DNS-записей.

Данная возможность описана в главе "Служба DNS" в пункте "Динамическое обновление зон".

28.7 Работа со службой

28.7.1 Команды просмотра данных

show service dhcp log [all | N] [archive N] [follow]

Показывает журнал сервиса.

Параметры:

- N - число записей
- all - показать все записи
- follow - просмотр журналов по мере появления
- archive - просмотр старых журналов

По умолчанию: show service dhcp log 25

show service dhcp status

Показывает текущий статус службы DHCP (корректность внутренней конфигурации, корректность базы данных зон, состояние сервиса).

show service dhcp lease [all | active | free | IP]

Показывает данные по выданным адресам: всем, действующим, свободным или по указанному.

Формат выходных данных:

HOST:<HOST> (<STATUS>)
IP: <IP> MAC: <MAC> <DATE_S> — <DATE_E>

Значения полей:

HOST - имя хоста клиента,присланное им в OHCP-запросе(может отсутствовать, т.к. протокол DHCP не требует передавать его)

STATUS - статус адресной информации (free - свободная, active - занятая)

IP - арендованный IP-адрес

MAC - MAC-адрес клиента

DATE_S - время начала срока аренды адреса (формат YYYY/MM/DD/HH:MM:SS)

DATE_E - время конца срока аренды адреса (формат YYYY/MM/DD/HH:MM:SS)

По умолчанию: show service dhcp lease all.

28.7.2 Команды удаления данных

service dhcp remove lease [IP | DOMAIN]

Прекратить аренду адреса по указанному IP-адресу узла или его имени DOMAIN.

Если ничего не указано - прекращается аренда всех адресов.

28.8 Примеры

Рассмотрим пример:

- пусть в DHCP описано несколько сетей и хостов, и среди них имеется сеть 1.2.3.0/24 и хост zeta.
- если необходимо для хоста с MAC-адресом 22:22:22:33:33:33 назначить север имен 2.2.2.2, для подсети 1.2.3.0/24 назначить сервер имен 3.3.3.3, а для всех остальных - сервер имен 5.5.5.5, следует выполнить следующие команды:

```
(config—service—dhcp)# name—server 5.5.5.5 (config—
service—dhcp)# host zeta (config—service—dhcp—host—zeta)#
mac 22:22:22:33:33:33
(config—service—dhcp—host—zeta)# name—server 2.2.2.2
(config—service—dhcp—host—zeta)# subnet 1.2.3.0/24 (config—
service—dhcp—subnet—1.2.3.0/24)# name—server 3.3.3.3
```

29. Служба DHCP6

Dionis-NX имеет службу DHCP6, реализующую серверную часть протокола DHCPv6 (Dynamic Host Configuration Protocol v6 — протокол динамической конфигурации узла версии 6).

Для базовой информации о службе DHCP6 и протоколе см п. 28.

Помимо общей информации о DHCP, протокол DHCPv6 использует понятия префикс сети и длина префикса сети. Префикс сети - это часть IPv6-адреса, которая задает адрес сети. Длина префикса сети - обозначает, какая часть IPv6 адреса является адресом сети, причем отсчет сетевого адреса начинается слева направо. Длина префикса может быть от 0 до 128 (максимальная длина в битах IPv6 адреса).

Например FF00::1/16 - задает префикс(сеть) FF00 и 112-битное пространство адресов внутри него.

Чтобы настроить службу DHCP, следует войти в режим ее конфигурации:

service dhcp6

Команда осуществляет вход в конфигурацию DHCP-службы.

29.1 Общие настройки службы

listen <ethernet | bond> <N>

см п. (28).

max-lease-time <N>

см п. 28.

min-lease-time <N>

см п. 28.

default-lease-time <N>

см п. 28.

respond-delay <N>

см п. 28.

send-hostname <on|off>

см п. 28.

ddns-ttl <N>

см п. 28.

29.2 Настройка статического назначения

Обычно фиксированные адреса назначаются важным узлам сети, например серверам.

Рабочим станциям обычно назначают адрес динамически(см. подраздел "Настройка динамического назначения").

host <NAME>

Вход в режим статического назначения IP-адреса для клиентов.

см п. 28.

host-id <DUID>

Задает привязку хост-декларации к идентификатору клиента.

Клиент использует уникальный идентификатор DHCP (DUID), чтобы получить IP-адрес и другие настройки от сервера DHCPv6. Фактическая длина DUID зависит от его типа. Первые 16 битов DUID содержат тип DUID, которых всего три. Значение оставшихся битов зависит от типа. Три типа, идентифицированные в RFC 3315:

- адрес уровня ссылки плюс время
- присвоенный поставщиками уникальный идентификатор
- адрес уровня ссылки

ip <1P>

Задает статический 1Ру6-адрес клиента.

Если клиент узел Dionis-NX, то данный адрес получит на клиенте длину префикса 64.

29.3 Настройка динамического назначения

Для настройки динамического назначения адресов необходимо описать сеть, внутри которой задает интервал адресов, подлежащих раздаче клиентам.

subnet <IP/MASK>

Вход в режим динамического назначение IP-адресов для клиентов,

range < <IP_START> <IP_END> | <SUBNET> [temporary] >

Задает диапазон адресов для назначения клиентам.

Существует два варианта задания диапазона:

- начальный и конечный адрес
- подсеть адресов; если задан параметр temporary - подсеть может использоваться для раздачи временных адресов (Temporary Addresses, опция 1A_TA)

Если клиент узел Dionis-NX, то данный адрес получит на клиенте длину префикса 64.

29.4 Делегирование префиксов

В DHCPv6 существует понятие Делегирующий роутер, раздающий префиксы и Запрашивающий роутер, который их получает и затем их использует для назначения внутри них адресов для своих клиентов.

В секции host можно настроить делегирование префиксов следующей командой:

prefix <PREFIX>

В данной команде PREFIX задает префикс в формате IPV6/MASK.

В секциях subnet можно настроить делегирование префиксов следующей командой:

prefix <IP_START> <IP_END> <PLEN>

В данной команде задается начальный и конечный адрес внутри сети subnet, а также длина префикса (PLEN) для данного интервала адресов.

29.5 Сетевые DHCP-опции

Помимо назначения клиентам IP-адресов, служба может передавать им другую конфигурационную сетевую информацию.

Эта информация называется сетевые DHCP-опции. Они могут быть указаны на любом из уровней службы: как глобально, так и в хост- или сетевой декларациях.

ia-prefix <LEN>

Задает длину префикса в процессе делеигрования префиксов службой. Данная опция передается в OHCPб-подопции IAPREFIX опции IA_PD.

Обычно используется службой автоматически и не требует задания администратором.

fqdn <NAME>

Задает FQDN клиента в процессе динамического DNS-обновления зон.

Обычно используется службой автоматически и не требует задания администратором,

info-refresh-time <VAL>

Сообщает клиенту, если он этого затребует, как долго ждать следующего обновления информации через Information-request сообщения.

domain-search <NAME>

Имя домена для разрешения имен через DNS.

Можно указать несколько доменов.

[N] name-server <1P>

IP-адрес сервера имен с приоритетом N.

[N] ntp-server <1P>

IP-адрес сервера времени с приоритетом N.

29.6 Пользовательские DHCP-опции

Существует возможность создать пользовательские DHCP-опции, которые будут передаваться указанным клиентам так же, как и стандартные DHCP-опции.

Подробности см в п. 28.

29.7 Связь с DNS (динамическое обновление)

Служба может работать в связке с DNS службой по части динамического обновления DNS-записей.

Данная возможность описана в главе "Служба DNS" в пункте "Динамическое обновление зон".

29.8 Работа со службой

Команды аналогичны службе DHCP, за исключением того, что в имени команды нужно указать имя службы dhcp6, например: show service dhcp6 log.

Подробности см п. (28).

29.9 Примеры

Рассмотрим пример настройки DHCP6:

```
listen ethernet 0
subnet fd55:1: :/64
range fd55:1:: 10 fd55:1:: 100
subnet fd77:111: :/64
1 name—server fd77: 111:: 1
prefix fd77:111:: fd77:lll:: 64
range fd77:lll: :/64
enable
```

30. Служба DHCPRELAY

Dionis-NX имеет службу DHCPRELAY, которая является ретранслятором DHCP- сообщений на указанные сервера.

Для запуска службы необходимо:

- указать хотя бы один IP-адрес сервера, на который будут перенаправляться DHCP-запросы, командой server;
- указать интерфейс для взаимодействия с DHCP-серверами командой listen-server.

Для настройки службы следует войти в ее конфигурацию:

service dhcprelay

Осуществляет вход в режим конфигурации службы DHCPRELAY.

30.1 Основные настройки

listen-server <1P>

Задает интерфейс, с которого будут перенаправляться DHCP-запросы на сервера DHCP. Кроме того, этот интерфейс будет действовать так же, как и интерфейсы указанные командой listen, т.е. будет принимать DHCP-запросы/ответы как от клиентов, так и от серверов. Поэтому, если реально присутствует всего один интерфейс, достаточно указать его только командой listen-server.

listen <IFACE>

Задать интерфейс, на котором следует ожидать запросы/ответы DHCP.

По умолчанию: все широковещательные интерфейсы.

30.2 Дополнительные настройки

send-relay-options

Служба будет добавлять к DHCP-запросу свой идентификатор, который состоит из Circuit ID (имя аппаратного порта получения запроса) и Remote ID (MAC-адрес интерфейса получения запроса); это т.н. DHCP Опция 82 — опция протокола DHCP, использующаяся для того чтобы проинформировать DHCP-сервер о том, от какого DHCP-ретранслятора и через какой его порт был получен запрос; применяется при решении задачи привязки IP-адреса к порту коммутатора и для защиты от атак с использованием протокола DHCP.

drop-alien-replies

Отбрасывать ответы DHCP-серверов, если они содержат чужую Опцию 82 (идентификаторы в ней не соответствует идентификаторам службы)

port <PORT>

Порт, на котором слушать DHCP-запросы

По умолчанию: 67.

max-packet-size <SZ>

Максимальный размер DHCP-пакета (вместе с Опцией 82).

По умолчанию: 576 байт.

mode <append| replace |forward |discard>

Указывает, что делать с входящими DHCP-пакетами, которые уже имеют внутри себя Опцию 82, т.е. пришли от других релэй-агентов.

Возможные варианты:

- append : добавить свои идентификаторы к Опции 82
- replace : заменить своими идентификаторами уже имеющиеся в Опции 82 (по умолчанию)
- forward : ничего не менять
- discard : отбросить

30.3 Пример

Приведем настройку для следующей сети:

K/lMEHT(ethernet0)<—>DhcpRelay(ethernet1(192.168.0.1)-ethernet2(10.0.0.1))<—>DhcpServer(ethernet3(10.0.0.2))

Интерфейсы ethernet0,ethernet1 - обслуживают клиентскую сеть 192.168.0.0/24.

Интерфейсы ethernet2,ethernet3 - обслуживают сеть сервера 10.0.0.0/24.

Минимальная конфигурация Dhcp Relay:

```
(config—service—dhcprelay)# listen ethernet 1
(config—service—dhcprelay)# listen—server ethernet 2
(config—service—dhcprelay)# server 10.0.0.2 (config—
service—dhcprelay)# enable
```

Минимальная конфигурация Dhcp Server:

```
 (config—service—dhcp)# local—address 10.0.0.2
 (config—service—dhcp—subnet—192.168.0.0/24)# range 192.168.0.10 192.168.0.100
(config—service—dhcp)# enable
```

31. Служба DCHPRELAY6

Dionis-NX имеет службу dhcprelay6, которая является ретранслятором DHCPv6- сообщений на указанные сервера.

Для запуска службы необходимо:

- указать хотя бы один интерфейс командой listen-from;
- указать хотя бы один интерфейс командой listen-to;

Для настройки службы следует войти в ее конфигурацию:

service dhcprelay6

Осуществляет вход в режим конфигурации службы dhcprelay6.

31.1 Основные настройки

listen-from <IFACE> [IP]

Задать интерфейс и его адрес, на котором служба будет ожидать запросы клиентов или других ретрансляторов DHCPv6. Если IP не задан, будет использоваться первый найденный не Link-local адрес интерфейса.

listen-to <IFACE> [IP]

Задать интерфейс и его адрес, на который служба будет перенаправлять запросы клиентов или других ретрансляторов DHCPv6. Если IP не задан, будет использоваться адрес FF02::1:2 (AII_DHCP_Relay_Agents_and_Servers).

31.2 Дополнительные настройки

send-iface-id

Слать Interface-ID опцию, даже если указан только один интерфейс командой listen-from. В случае, если таких интерфейсов несколько, данная опция шлется в любом случае.

max-hops <N>

Максимальное число узлов, через которые может пройти DHCPv6-naKeT, прежде чем будет отброшен службой.

Умолчание: 10.

port <PORT>

Порт, на котором слушать DHCPv6-3anpocbi

По умолчанию: 547.

31.3 Пример

Приведем настройку для следующей сети: K/IMEHT(ethernetO)<—

>DhcpRelay(ethernetl-ethernet2)<—>DhcpServer(ethernet3) Интерфейсы

ethernetO,ethernetl - обслуживают клиентскую сеть.

Интерфейсы ethernet2,ethernet3 - обслуживают сеть сервера.

Минимальная конфигурация Dhcp Relay: (config—service—dhcprelay)# listen—from ethernet 1 (config—service—dhcprelay)# listen—to ethernet 2 (config—service—dhcprelay)# enable

32. Служба PROXY

Dionis-NX имеет службу PROXY, которая является прокси-сервером для протоколов FTP и HTTP.

Прокси-сервер - это служба, позволяющая клиентам выполнять косвенные запросы к другим сетевым службам.

Далее под службой будем понимать описываемую прокси-службу Dionis-NX, под УС будем понимать удаленный сервер, ресурс которого желает получить клиент через прокси-службу.

Рассмотрим алгоритм обработки HTTP-запроса клиента через PROXY-службу при задействовании всех ее возможностей:

1. клиент пытается получить ресурс, расположенный, например, на удалённом HTTP-сервере (web-страница, картинки, аудио-, видео-файлы и др.);
2. клиент подключается к службе или запрос клиента перенаправляется маршрутизатором на службу;
3. служба аутентифицирует пользователя, анализирует запрос клиента и проверяет,разрешен ли данный запрос для данного клиента;
4. если запрос разрешён, могут быть сделаны модификации заголовков HTTP-запроса клиента;
5. далее служба ищет запрашиваемый ресурс в своем кэше;
6. если ресурс найден в кэше, делается проверка на свежесть ресурса;
7. если ресурс свеж, он возвращается клиенту;
8. если ресурс стар или не найден в кэше, то служба пытается получить его с удаленного HTTP-сервера, возможно на ограниченной скорости;
9. если ресурс получен с удаленного сервера, то служба проверяет, разрешен ли данный ресурс для возврата клиенту;
10. если ресурс разрешен, то служба проверяет, нужно ли его кэшировать;
11. перед возвратом ресурса его HTTP-за головок также может быть изменен;
12. наконец ресурс возвращается клиенту.

В дальнейшем в разделах данной главы будет указано, к какому пункту данного алгоритма относится раздел.

Таким образом, основные цели службы:

- кэширование данных: может держать копию часто запрашиваемых Интернет-ресурсов в своем кэше и выдавать по запросу, снижая тем самым нагрузку на Интернет-канал и ускоряя получение клиентом запрошенной информации;
- защита локальной сети от внешнего доступа: локальные узлы взаимодействуют с внешними ресурсами только через службу, а внешние узлы не могут обращаться к локальным;
- ограничение доступа из локальной сети к внешней:
  - на удаленные ресурсы определённого типа;
  - для определённых клиентов;
  - на объем трафика извне в сторону определённых клиентов и/или сетей.

32.1 Общие понятия

32.1.1 Режимы работы службы

Существуют два режима работы службы:

- обычный режим: необходимы настройки прокси-сервера в браузере клиента; поддерживает проксирование/кэширование HTTP, FTP протоколов
- прозрачный режим: настраивать браузеры клиентов не нужно; необходима настройка NAT-правил; поддерживает проксирование/кэширование только HTTP протокола; успешность выполнения запроса полностью зависит от наличия заголовка Host в HTTP-запросе клиента.

Более подробно настройка службы в обоих режимах описана далее в подразделе "Общие настройки службы".

32.1.2 Правила и списки доступа

Основные понятия службы - списки контроля доступа(AC1_) и правила доступа. ACL - это именованный набор элементов определённого типа.

Формат ACL следующий: **acl <NAME> <TYPE> <PARAMS>**

Рассмотрим поля команды: [10]

10NAME - это имя ACL;

- TYPE - это тип ACL; определяет тип элементов данного списка ACL; может принимать множество значений, рассмотренных ниже;

- PARAMS - это элементы, входящие в данный список.

Рассмотрм возможные типы ACL:

- src - IP-адреса источника (медленный тип);
- dst - IP-адреса назначения (медленный тип);
- myip - IP-адреса локальных интерфейсов системы;
- srcdomain - доменное имя источника;
- dstdomain - доменное имя назначения;
- srcdom-regex - регулярное выражение для доменного имени источника;
- dstdom-regex - регулярное выражение для доменного имени назначения;
- random - случайная частота события (задается дробным числом типа 1/N);
- time - дата и или время события;
- port - порт назначения;
- myport - локальный порт системы;
- proto - протокол HTTP или FTP;

- method - метод HTTP;
- user-agent - приложение клиента (UserAgent);
- proxy-auth - имя пользователя;
- proxy-auth-all - все пользователи;
- proxy-auth-regex - регулярное выражение для имени пользователя;
- maxconn - максисмальное число прямых TCP-соединений (X-Forwarded-For не учитывают-ся);
- max-user-ip - максимальное число попыток аутентификации с разных IP-адресов для одного и того же пользователя;
- req-mime - MIME-тип запроса клиента;
- rep-mime - MIME-тип ответа пользователя;
- mac - MAC-адрес для клиентов из той же подсети;
- uri - регулярное выржание для URI;
- urn - регулярное выржание для URN.

Под источником и назначением обычно в службе прокси понимается клиент и сервер.

На примерах рассмотрим, как создавать ACL:

```
(config—service—proxy)# acl al src 10.0.0.0/24 10.0.1.0/24
(config—service—proxy)# acl al src 10.0.2.0/24 (config—
service—proxy)# acl a2 src 10.0.3.0/24 (config—service—
proxy)# acl ul uri xxx
```

В результате: ACL al будет состоять из 3х сетей 10.0.0.0/24 10.0.1.0/24 10.0.2.0/24; ACL a2 будет состоять из сети 10.0.3.0/24; ACL ul будет состоять из всех запросов, в URI которых есть слово xxx.

При поиске службой того ACL, в пределы которого попадает запрос клиента или ответ сервера, используется OR-логика: если запросу соответствует хотя бы один из элементов, перечисленных в данном ACL, то считается, что запрос попал в данный ACL, и поиск прекращается. Поэтому, для оптимизации работы службы, при определении параметров ACL лучше задавать первым тот параметр ACL, вероятность попадания в который **максимальная.**

Рассмотрим пример создания правил доступа на основе ранее созданных ACL:

```
(config—service—proxy)# http—access deny al ul
(config—service—proxy)# http—access permit a2
(config—service—proxy)# http—access deny all
```

При поиске службой правила, в который попадает запрос клиента или ответ сервера, использу-ется AND-логика: запрос/ответ должен удовлетворять всем ACL, перечисленным в правиле. По-этому, для оптимизации работы службы, при определении правил на основе ACL лучше задавать первым тот параметр правила, вероятность попадания в который **минимальная.**

В данном примере доступ запрещается для запросов, приходящих из сетей al, только если они пытаются получить ресурс, в URI которого есть слово xxx. Для сетей из a2 разрешается свободный доступ. Для всех остальных сетей доступ запрещён (используется встроенный ACL с именем all).

Помимо правил доступа, существуют и другие команды, использующие ACL. Например: Рас-смотрим пример создания правил доступа на основе ACL:

(config—service—proxy)# max—reply—size al

Наконец, добавим еще одно правило по оптимизации работы службы: правила на основе ACL типа srcdomain, dst, proxy-auth лучше определять как можно раньше в списке.

Важное замечание: для правила доступа любого типа рекомендуется всегда добавлять в конец списка правил данного типа запрещающее или разрешающее (зависит от контекста) правило, направленное на всех (например, http-access deny all).

Существуют ACL особого типа - комплексные ACL:

(config—service—proxy)# acl—any anyl al a2

Данный ACL anyl выполнится, если выполнится хотя бы один из ACL al или a2.

(config—service—proxy)# acl—all all 1 al a2

Данный ACL all 1 выполнится, если выполнятся все из ACL al или a2.

32.1.3 Регулярные выражения

В некоторых командах службы параметры задаются упрощёнными регулярными выражениями (РВ). Тип параметра в этом случае называется REGEX.

РВ - это формальный язык поиска подстрок в тексте, основанный на использовании метасимволов.

По сути РВ - это строка-образец, состоящая из символов (С) и метасимволов (МС) и задающая правило поиска.

МС - это С, который используется для замены других С или их последовательностей, приводя таким образом к символьным шаблонам.

В дальнейшем в этом разделе все символы,указанные в кавычках, должны вводиться в системе без кавычек.

Кратко опишем используемые МС:

- «$» - задает конец строки; например: [0-9]{2}$ - это все строки, оканчивающиеся на двухзначные числа.
- «()» - задает группу символов; например: а([0-9]Ь){2} - это а1Ь4Ь, а5Ь2Ь и т.д., т.е. МС {2} применяется к группе ([0-9]Ь).

Существует возможность использовать МС как С, т.е. МС будет иметь только функцию символа, т.е. отображать сам себя. Это называется экранирование МС.

Символ экранирования (СЭ) - «\\».

При установке параметра типа REGEX, если в значении параметра используется СЭ, необходимо взять все значение параметра в двойные кавычки,например:

(config—service—proxy)# acl al urn "\\.cgi$"

При удалении параметра типа REGEX, если в значении параметра используется СЭ, необходимо взять все значение параметра в двойные кавычки и продублировать СЭ, например:

(config—service—proxy)# no acl al urn "\\\Vcgi$"

32.2 Общая настройка службы

Часть информации о настройке в данном разделе, а именно настройка режимов работы службы, относится п.2 Алгоритма.

Чтобы войти в режим настройки службы, следует выполнить команду:

(config)# service proxy

Необходимо также задать почтовый адрес администратора службы, которая будет указана в веб-страницах, описывающих проблему с выполнением запроса клиента, чтобы он мог, в случае необходимости, узнать причину данной проблемы:

(config—service—proxy)# admin—email ivanov@company.ru

По умолчанию используется адрес admin@company.

Важно: система должна быть способна разрешать имена узлов через DNS. Для этого либо настройте и включите службу DNS и/или укажите в ip resolver адрес(а) серверов имен.

32.2.1 Настройка службы в обычном режиме для HTTP/FTP

Следует настроить порт и адрес(необязательно), на которых служба будет ожидать входящие запросы:

(config—service—proxy)# listen 192.168.0.1 3128

Данная команда настраивает службу в обычном режиме, т.е. клиентам необходимо будет указать в настройках своих Интернет-браузеров адрес и порт службы.

Для проксирования FTP в обычном режиме используйте команду listen-ftp.

32.2.2 Настройка службы в прозрачном режиме для HTTP/FTP

Обычный режим может быть не очень удобным, т.к. необходимо настраивать браузеры клиентов.

Поэтому можно настроить службу в режиме intercept (режим перехвата или прозрачный режим).

Для проксирования FTP в прозрачном режиме используйте команду listen-ftp с опцией intercept.

Рассмотрим пример настройки службы в прозрачном режиме, обслуживающую запросы, приходящие из сети 192.168.1.0/24 на 80-й порт(HTTP) на интерфейс ethernet2 с адресом 192.168.1.254:

```
(config)# ip nat—list proxy
(config — nat—proxy)# exclude in tcp dport 80 dst 192.168.1.254
(config —nat—proxy)# nat tcp dport 80 src 192.168.1.0/24 redirect port 3127
(config)#
(config)# ip access—list dropmyself
(config—acl—dropmyself)# deny tcp dport 3127 dst 192.168.1.254
(config)#
(config)# interface ethernet 2
(config —if—ethernet2)# ip nat—group proxy (config—
if— ethernet2)# ip access—group dropmyself in
(config)#
(config—service—proxy)# listen 192.168.1.254 3127 intercept
```

В данном режиме службы клиентам не нужно ничего настраивать, они автоматически, после запуска службы, будут работать через нее. В данном примере предполагалось, что клиенты обслуживаются маршрутизатором Dionis-NX, на котором и настраивается прокси-служба.

Рассмотрим вышеприведенные команды: [11]

---

11 правило proxy перенаправляет трафик на 80-й порт с адресов сети 192.168.1.0/24 на порт 3127, исключая при этом запросы на 80й порт локальной WEB-службы;

- правило dropmyself отбрасывает весь трафик, приходящий на адрес и порт прокси сервера, объявленные ниже как intercept: это нужно для предотвращения обработки запросов непосредственно на прерывающий сокет службы, т.е. на 192.168.1.254:3127, т.к. это может вызвать петлю перенаправления запросов, когда служба будет бесконечно слать запрос самой себе, думая что она и есть удалённый сервер;
- ip nat-group proxy и ip access-group dropmyself in применяют вышеописанные правила на интерфейсе; вместо ip nat-group нужно использовать ip nat-group-xfrm, если трафик через интерфейс предполагается заворачивать в какой-либо туннель, например DISEC: в этом случае NAT выполняется до «заворачивания» в туннель для отсылаемого через интерфейс трафика и после «разворачивания» из туннеля для принимаемого интерфейсом трафика;
- команда listen непосредственно определяет сокет (адрес и порт) для принятия HTTP-запросов и объявляет его прерывающим сокетом; пары 0.0.0.0:3128 и 127.0.0.1:3128 запрещены.

Важно: в команде listen для режима intercept можно указать опцию no-pmtu-discovery, если у некоторых клиентов иногда возникают проблемы с долгим ожиданием обработки HTTP-запроса. Это может быть связано проблемой доставки ICMP Must fragment сообщения:

(config—service—proxy)# listen 192.168.1.254 3127 intercept no—pmtu—discovery

Вы можете указать неограниченное количество сокетов обоих типов, на которых будут приниматься запросы. Единственное ограничение: один и тот же сокет может быть только одного типа.

32.3 Настройка параметров кэширования

Далее настроим параметры службы, описывающие кэш.

Настройка кэша:

(config—service—proxy)# cache type aufs 32 256

Данная команда задает тип кэша на диске aufs, размер кэша в памяти 256 Мб и на диске 32 Мб.

Другой тип кэша, который может быть использован в данной команде, это ufs. Отличие ufs от aufs в том, что первый обеспечивает синхронные файловые операции над кэшем, а второй - асинхронные, что более эффективно. Рекомендуется всегда использовать aufs.

Настройка механизма замены объектов в кэше на диске и в памяти:
(config—service—proxy)# cache replacement—policy disk hlru (config—
service—proxy)# cache replacement—policy memory hlru

Рекомендуется использовать один и тот же механизм для кэша в памяти и кэша на диске.

Опишем возможные механизмы замены объектов:

- lru : Least Recently Used (сначала заменяются самые старые объекты, к которым давно не было доступа);
- hlru : Heap-based Last Recently Used (аналогично предыдущему,но в новой, более эффективной реализации);
- gdsf : Greedly Dual Size Frequency (в кэше остаются прежде всего популярные маленькие объекты);
- lfuda : Least Frequently Used with Dinamic Aging (в кэше остаются прежде всего популярные объекты,вне зависимости от размера).

Настройка размеров кэшируемых объектов на диске(первые две команды) и в памяти(третья команда):

(config—service—proxy)# cache limit disk max 500
(config—service—proxy)# cache limit disk min 5
(config—service—proxy)# cache limit memory max 8

Размер задается в Кб. Объекты,размер которых выходит за указанные рамки, не будут кэшироваться. Минимальное ограничение для объектов в памяти равно 0 и не может быть изменено.

32.4 Настройка доступа к службе

Данный раздел относится к пп.3 и 9 Алгоритма.

С помощью этих настроек осуществляется контроль над тем, какие узлы могут иметь доступ к сети Интернет и какие данные они могут запрашивать через прокси-службу.

Для начала напомним, как формируется URI (Унифицированный идентификатор ресурса): URI=URL|URN, где | - конкатенация.

Напрмер, URI = ftp://ftp.dlink.ru/pub/ADSL/ , где: URL = ftp://ftp.dlink.ru , URN = /pub/ADSL/

Далее рассмотрим примеры.

Запрет доступа к нежелательным сайтам и фильтрация ответов сервера:

```
 (config—service—proxy)# acl workers src 10.0.0.0/24
(config—service—proxy)# acl bosses src 10.0.1.0/24
(config—service—proxy)# acl mimel rep—mime ^audio
(config—service—proxy)# acl blacklist uri ^httpV/jihad
(config—service—proxy)# acl blacklist uri kommunist
(config—service—proxy)# acl blacklist urn 'Vcgi—bin
(config—service—proxy)# acl blacklist uri
        ^http://bad\\.site\\.numberfO—9]+\\.sym—.\\.tail — *\\.ru$
(config—service—proxy)# acl docsite uri ^httpV/documentation
(config—service—proxy)# http—access deny blacklist workers
(config—service—proxy)# http—access deny docsite bosses
(config—service—proxy)# http—access permit all (config—
service—proxy)# http—reply—access deny mime1
```

Данные правила запрещают работникам (сеть workers) посещать следующие сайты:

- сайты, начинающиеся на http://jihad;
- сайты, в URI которых есть слово kommunist;
- сайты, в URN которых есть /cgi-bin, т.е. запрет доступа к CGI-приложениям серверов;
- сайты, имеющие вид http: //bad . site. number$A.sym-$B.tail-$C.ru, где:
    - $A -это набор цифр от 0 до 9;
    - $B - любой символ;
    - $C - любой набор символов, том числе и пустой.

Кроме того, данные правила запрещают директорам (сеть bosses) посещать сайты, начинающиеся на http://documentation.

Другим лицам (которые не входят в категории bosses и workers) доступ на любые сайты разрешен.

Если доступ разрешен, то используется последнее правило (http-reply-access): после получения ответа от сервера анализируется тип содержимого ответа (поле Content-Type HTTP-заголовка ответа) и если типа содержимого начинается с audio, то содержимое клиенту не возвращается и не кэшируется.

32.5 Настройка фильтрации HTTP-заголовков

Данный раздел относится к п.4 и 11 Алгоритма.

Служба имеет возможность удалять HTTP-загловки и изменять их значения в запросах клиентов и/или ответах сервера.

По умолчанию никакие заголовки не удаляются.

Для исключения заголовка из HTTP-пакета запроса и ответа используются следующие команды: request-header-access deny и reply-header-access denyсоответственно.

Для включения заголовка в HTTP-пакета запроса и ответа используются следующие команды: request-header-access permit и reply-header-access permit соответственно.

Для замены содержимого в ранее исключенных HTTP-заголовках запроса и ответа используются следующие команды: request-header-replace и reply-header-replace соответственно.

Следует обратить внимание, что замена содержимого работает только для заголовков, которые попадают в request/reply-header-access deny правила.

Рассмотрим пример. Скрытие от удаленных серверов названия приложения, с помощью которого клиенты локальной сети работают в сети Интернет:

```
(config—service—proxy)# acl Ian src 10.0.0.0/24
(config—service—proxy)# request—header—access deny User—Agent Ian (config—
service—proxy)# request—header—access permit All Ian (config—service—proxy)#
request—header—replace User—Agent "Fake Browser 1.0"
```

Данные команды заменяют информацию в заголовке User-Agent во всех исходящих из сети 10.0.0.0/24 HTTP-пакетах на Fake Browser 1.0.

Аналогично можно менять заголовки в ответах HTTP-серверов командами reply-header-access/reply-header-replace.

32.6 Настройка выборочного проксирования

Служба имеет возможность предписывать, какие запросы и для каких пользователей следует всегда направлять только через службу proxy, а какие следует передавать напрямую на сервер назначения, минуя службу proxy.

Однако ответы от сервера назначения будут также и далее кэшироваться службой, только если это не запрещано специально (см. раздел "Настройка выборочного кэширования").

Рассмотрим примеры. Запретим проксирование всех запросов на utro.ru и его поддомены, кроме поддомена mail.utro.ru:

```
(config—service—proxy)# acl direct1 dstdomain mail.utro.ru
(config—service—proxy)# acl direct2 dstdomain .utro.ru
(config—service—proxy)# always—direct deny direct1
(config—service—proxy)# always—direct permit direct2
```

Как видно из примера, для указания запросов, перенаправляемых напрямую, используется команда always-direct.

Существует противоположный ей вариант - команда never-direct - предписывает, какие запросы не следует направлять напрямую на сервер назначения.

Рассмотрим пример ее использования:

```
(config—service—proxy)# acl local—servers dstdomain .foo.net
(config—service—proxy)# never—direct deny local—servers
(config—service—proxy)# never-direct allow all
```

Данный пример предписывает все запросы, кроме запросов на поддомены домена foo.net , выполнять через службу.

32.7 Настройка выборочного кэширования

Данный раздел относится к п.10 Алгоритма.

Служба имеет возможность предписывать, какие объекты и для каких пользователей следует кэшировать, а какие нет.

Рассмотрим примеры. Запретим кэширование объектов, расположенных в локальной сети:

```
(config—service—proxy)# acl lan dst 10.0.0.0—10.0.3.0/24
(config—service—proxy)# caching deny lan (config—
service—proxy)# caching permit all
```

Обычно не следует кэшировать объекты, находящиеся на HTTP-серверах локальной сети, т.к. доступ к ним и так достаточно быстрый. Это сохранит место в кэше для других объектов. В данном примере все HTTP-запросы на адреса из четырех сетей (10.0.0.0/24 - 10.0.3.0/24) не будут кэшироваться.

Другой пример. Запретим в LAN кэширование всего, кроме аудиоинформации:

```
(config—service—proxy)# acl lan dst 10.0.0.0—10.0.3.0/24
(config—service—proxy)# acl aud req — mime ^audio
(config—service—proxy)# caching deny lan laud (config—
service—proxy)# caching permit all
```

32.8 Настройка аутентификации

Данный раздел относится к п.3 Алгоритма.

Служба имеет возможность проводить аутентифиуацию пользователей, желающих работать через нее. Это возможно только в обычном режиме работы службы.

В этом случае клиенты, настроив свой браузер на использование прокси, при попытке выхода в Интернет получат приглашение (с названием Dionis NX-PROXY) ввести имя пользователя

и пароль. Служба, одобрив аутентификацию, может с помощью правил доступа задать разное обслуживание для разных пользователей, прошедших аутентификацию.

В службе существует 2 вида аутентифицации:

- локальная
- удаленная (по LDAP протоколу)

Они могу быть в системе одновременно, в этом случае необходимо учитывать следующие особенности:

- при добавлении аутентификации нового типа при включенной службе, необходимо ее перезапустить
- при наличии двух видов аутнтификации в конфигурации для разных Интернет-браузеров (клиентов службы), может использоваться свой тип аутентификации: выбор зависит от браузера и удаленной системы, на приемр может выбраться первая предложенная службой аутентификация, а может и более криптографически сильная из двух.

32.8.1 Локальная аутентификация

Для аутентификации данного типа необходимо добавить в конфигурацию системы имена пользователей и их пароли. Рассмотрим пример.

Сначала включим локальную аутентификацию:

```
(config—service—proxy)# auth local 3
```

Указываем число параллельных процессов аутентификации.

Далее добавим пользователей:

```
(config—service—proxy)# user ivan pas1
(config—service—proxy)# user liza pas2
(config—service—proxy)# user oleg pas3
(config—service—proxy)# user mary pas4
```

Например, у пользователя ivan пароль pas1.

Далее создадим список доступа для пользователей:

```
(config—service—proxy)# acl allusers proxy—auth—all
(config—service—proxy)# acl vipusers proxy—auth "oleg mary"
(config—service—proxy)# acl urll url —regex ^httpV/vk
(config—service—proxy)# http—access permit urll vipusers
(config—service—proxy)# http—access deny urll allusers
(config—service—proxy)# http—access permit allusers
(config—service—proxy)# http—access deny all
```

Данные правила разрешают доступ в Интернет только тем пользователям, которые прошли аутентификацию. Среди аутентифицированных пользователей запрещен доступ к сайтам, начинающимся на http://vk, всем, кроме пользователей oleg и mary (vipusers).

Также vipusers можно задать в следующем виде:

(config—service—proxy)# acl vipusers2 proxy—auth oleg
(config—service—proxy)# acl vipusers2 proxy—auth mary

В данном случае ACL vipusers и vipusers2 полностью идентичны.

32.8.2 Удаленная аутентификация LDAP

В данном случае аутентификация проходит на удаленном сервере аутентификации (далее CA), поддерживающем базу данных пользователей. Этот сервер должен поддерживать LDAP протокол. Например, это может быть Windows Server с поддержкой Active Directory (данная подсистема имеет LDAP-компонент).

Формат команды для включения такой аутентификации:

auth ldap <IP> <USERS_DN> [binddn <BIND_DN> <PAS>] [filter FILTER] [NUM] [credttl SEC] [ssltls]

Параметры команды следующие:

- IP - IP-адрес CA
- USERS_DN - имя LDAP-каталога, где содержится информация о пользователях; например "dc=domain,dc=int"
- BIND_DN - указывает пользователя, от имени которого будет производится поиск в LDAP-каталоге; например "cn = bob,cn = users,dc=domain,dc=int"
- PAS - пароль для BIND_DN пользователя
- NUM - число параллельных процессов аутентификации на локальной системе.
- FILTER - задает имя атрибута по которому искать пользователя в каталоге, например uid или cn; по умолчанию sAMAccountName - для поиска в каталогах Windows AD
- SEC - время жизни сессий аутентифицировавшихся пользователей (в секундах)
- ssltls - шифровать LDAP трафик по SSL/TLS протоколу (необходима поддержка SSL/TLS на CA).

Параметры USERS_DN и BIND_DN можно вводить как в полной форме

("dc=domain,dc=int,cn = user"), так и в краткой.

Формат краткой формы - [U1[.U2[.U3...]]@]D1[.D2[.D3...]]
В данном формате:
- Ul..Un - необязательное имя пользователя, в полной форме становится cn = Ul,cn = U2,...,cn =Un.
- Dl..Dn - необязательное имя домена, в полной форме становится dn = Dl,dn = D2,...,dn =Dn.

Рассмотрим пример:

(config—service—proxy)# auth ldap 10.0.0.1 factor—ts.int binddn admin@factor—ts.int passl23

Далее необходимо настроить правила аутентификации также, как это сделано в локальной аутентификации.

32.9 Настройка контроля пропускной способности сети

Данный раздел относится к п.8 Алгоритма.

Служба может контролировать скорость потока данных от удалённых серверов к пользователям: ограничения накладываются **только** на кэш-промахи, кэш-попадания не ограничиваются по скорости.

Контроль пропускной способности сети действует при помощи пулов задержки.

Для лучшего понимания настройки рассмотрим механизм пулов задержки на основе аналогии с ведрами и кранами.

Существует 4 типа ведер:

- общее ведро: может ограничивать весь трафик (тип agr);
- общее ведро, разделенное на 256 индивидуальных вёдер: может ограничивать весь трафик и трафик по каждому из 256 узлов (тип agr-host24);
- общее ведро, разделенное на 256 сетевых ведер, разделенных на 256 индивидуальных ведер: может ограничивать весь трафика, трафик по каждой из 256 С-сетей и трафик по каждому из 65536 узлов (т.е. 256 узлов из 256 сетей) (тип agr-net24-host16);
- аналогично предыдущему типу, только добавляется контроль трафика по каждому аутентифицированному пользователю (тип agr-net24-host16-user).

Рассмотрим алгоритм работы контроля скорости трафика на основе общего ведра.

Весь входящий внешний трафик заполняет общее ведро. В этом общем ведре есть один большой кран, из которого пользователи службы получают данные удаленных серверов, которые наполняют ведро своим трафиком.

У ведер есть параметры: размер и скорость наполнения. Размеры и скорость указываются в байтах и байтах в секунду соответственно. Возможно использование префиксов kb,mb,gb - для кило,мега и гигабайт (размер умножается на 10^3,10^6 и 10^9 соответственно).

Как только клиенты службы через один большой кран сольют себе весь трафик из ведра (в случае если их общая скорость слива превысит скорость наполнения ведра) и ведро перестанет успевать наполняться, включается общее ограничение (если оно задано) на скорость подачи трафика из крана, которое становится равным скорости наполнения ведра. Если скорость слива ниже скорости наполнения, ведро будет заполняться (до установленного объёма).

Чтобы настроить общее ограничение, следует выполнить команду:

```
(config—service—proxy)# delay—pool pool1 agr 1mbs/100mb
```

Данное ведро имеет размер 100Мб и скорость наполнения 1mbs. Пока оно не пусто - клиенты не ограничены. Как только оно становится пустым, общая скорость для всех клиентов падает до lmbs.

Механизм для остальных ведер аналогичен. Отличие только в размере и скорости наполнения ведер: например, по каждой сети типа С, по каждому узлу, по каждому пользователю.

Рассмотрим пример:

(config—service—proxy)# delay—pool pool2 agr—host24 10mbs/100mb 1mbs/10mb
(config—service—proxy)# delay—pool pool3 agr—net24—host16 10mbs/100mb 1mbs/10mb 100kbs/lmb
(config—service—proxy)# delay—pool pool4 agr—net24—host16—user 10mbs/100mb 1mbs/10mb 100kbs/1mb 50kbs/1mb
(config—service—proxy)# delay—pool pool5 agr—host24 10mbs/100mb —1/—1

Рассмотрим подбробно, какие ведра определены данными командами:

- poo12: общее ведро размером 100Мб и скоростью наполнения 10Мбс, внутри которого есть 256 ведер 1Мбс/10Мб для каждой С-сети;
- poo13: общее ведро размером 100Мб и скоростью наполнения 10Мбс, внутри которого есть 256 ведер 1Мбс/10Мб для каждой С-сети, и в каждом из таких ведер есть 256 ведер, задающих ограничение lOOkbs/lmb на каждый хост;
- poo14: аналогично poo13, только кроме этого задается ведро 50kbs/1mb на каждого пользователя;
- poo15: аналогичен poo12, только ограничение на С-сеть не накладывается (-1 обозначает отсутствие ограничения).

Разрешающие правила (delay-access permit) ограничивают трафик ведром, указанным в правиле, для пользователей, попадающих под действие указанного в правиле ACL. Если такое правило найдено - просмотр правил прекращается и правило применяется.

Запрещающие правила (delay-access deny) не ограничивают трафик ведром, указанным в правиле, для пользователей, попадающих под действие указанного в правиле ACL. Если такое правило найдено - просмотр правил для ведра данного типа прекращается, однако продолжается для ведер других типов.

32.10 Правила проверки объектов в кэше на свежесть

Данный раздел относится к пп.6-8 Алгоритма.

Служба имеет возможность настраивать правила, по которым те или иные объекты кэша будут считаться службой свежими или несвежими.

Свежесть объекта, в свою очередь, определяет, необходимо ли обновлять в кэше объект с удаленных серверов или нет.

Говоря кратко, чем чаще объекты определяются как свежие, тем больше будет процент попадания в кэш и , как следствие, большая скорость обслуживания клиентов.

Далее под ресурсом будем понимать именованную последовательность данных, расположенную на HTTP-сервере и доступную по ее URI.

Под объектом будем понимать размещенный в кэше службы ресурс, полученный с HTTP-сервера.

Перед рассмотрением правил напомним и введем несколько понятий.

Заголовки HTTP, необходимые для понимания правил:

- Date : дата генерации HTTP-ответа сервером;
- Last-Modified : дата последней модификации ресурса;
- Expires : дата предполагаемого истечения срока актуальности ресурса;
- If-Modified-Since : выполнять указанный HTTP-метод если ресурс изменился с указанного момента.

Директивы HTTP-заголовка Cache-Contol, который управляет кэшированием:

- no-cache : сервер не должен использовать кэшированный ответ;
- no-store : ответ на этот запрос не должен кэшироваться;
- max-age : максимальное время хранения объекта в кэше;
- must-revalidate : если кэшированный ответ устарел, то должен быть обновлен, вне зависимости от правил кэш-службы.

Другие термины:

- возраст объекта (ВО) - это разница между Date и Last-Modified объекта; Date обновляется, а Last-Modified может обновиться во время последнего запроса объекта с HTTP-сервера;
- возраст реакции (ВР) - это степень несвежести объекта, т.е. время прошедшее с момента последнего запроса объекта клиентом; иначе говоря, это время с момента последней валидации службой объекта на свежесть: возврата свежего объекта из кэша, получения отсутствующего или обновление несвежего объекта с сервера;
- LM-фактор - это отношение ВР к ВО.

Рассмотрим, как создаются правила проверки на свежесть:

(config—service—proxy)# refresh \\.jpg$ 1000 70% 5000

Все параметры команды refresh обязательны. Разберем их по порядку появления в строке команды:

- \\jpg$ " это РВ, задающее URI объектов, для которых будет применяться данное правило; в данном случае это картинки с расширением jpg;
- 5000 - это максимальный ВР в минутах (ВРМАКС); объект несвеж, если его ВР больше ВР-МАКС, в данном случае 5000 мин;
- 70% - это максимальный LM-фактор (ЛММАКС); объект несвеж, если его LM-фактор больше ЛММАКС, в данном случае 70%;
- 1000 - это минимальный ВР в минутах (ВРМИН); объект свеж, если его ВР меньше ВРМИН, в данном случае 1000 мин.

Правила проверки на свежесть будут работать только для объектов, у которых не указан Expires, если обратное не указано специальной опцией правила (см.ниже).

Чем больше прошло времени с момента последней проверки на свежесть, тем менее свеж объект и тем больше LM-фактор. Когда LM превысит значение, заданное в команде, объект перестанет быть свежим.

Порядок правил важен - как только запрашиваемый объект попадет в одно из правил, поиск правил будет остановлен для данного объекта.

Рассмотрим алгоритм проверки на свежесть в случае, если не заданы дополнительные опции правила:

1. проверка no Expires, если он есть в пакете: если дата прошладо объект не является свежим; если дата еще не наступиладо объект является свежим; выход;
2. если ВР больше чем ВР_МАКС, то объект не является свежим и требуется его обновление с удаленного сервера;
3. если ВР меньше чем ВР_МАКС, то осуществляется проверка по LM-фактору: если объект является свежим - выход;
4. если ВР меньше ВР_МИН, то несвежесть по LM-фактору не убедительна: объект был совсем недавно получен и мог быть создан также совсем недавно, в результате LM-фактор стал высоким; т.о если ВР меньше ВР_МИН, то объект является свежим - выход;
5. иначе, несвежесть по LM-фактору убедительна, объект признается устаревшим и необходимо его обновление с удаленного сервера.

Рассмотрим дополнительные опции правила, которые могут изменить вышеописанный алгоритм:

- override-expire : игнорировать заголовок Expires;
- override-lastmod : игнорировать заголовок Last-Modified;
- reload-into-ims : использовать заголовок If-Modified-Since (только при наличии Last-Modified) вместо директивы no-cache;
- ignore-no-cache : игнорировать директиву no-cache;
- ignore-no-store : игнорировать директиву no-store;
- ignore-reload : игнорировать директивы no-cache и reload;
- ignore-must-revalidate : игнорировать директиву must-revalidate;
- refresh-ims : всегда обновлять объект, вне зависимости от наличия If-Modified-Since в запросе клиента.

Пример использования дополнительных опций:

(config—service—proxy)# refresh \\.jpg$ 10000 70% 20000 reload —into—ims ignore-reload

32.11 Настройка адаптации содержимого

Служба имеет возможность анализировать, блокировать или изменять содержимое проксируемых через нее сообщений. Это называется адаптация содержимого, хотя в некоторых случаях никакого изменения содержимого не производится.

Для настройки адаптации содержимого необходимо указать сервер адаптации:

adapt-srv NAME <icap|ecap> <request| response > <URL:PORT[URN]> [bypass] [routing] [on-overload TYPE] [ipv6] [CONLIM]

Параметры команды:

• NAME - краткое имя-идентификатор задаваемого сервера адаптации

icap|ecap - протокол сервера адаптации (ICAP либо ECAP)

- request|response - задает тип сообщений, обрабатываемых данным сервером - запросы или ответы.

- URL:PORT[URN] - задает имя домена сервера адаптации и его порт, а также URN - необязательное имя ресурса (путь к приложению на сервере адаптации)

- bypass - этот сервер является необязательным, в случае его недоступности, можно использовать следующий сервер в цепочке серверов или в множестве серверов (см.ниже)
- routing - динамическое изменение плана адаптации содержимого текущего обрабатываемого сообщения
- TYPE - (только для icap) задает что действие при достижении максимального числа соединений: block(послать клиенту сообщение об ошибке), bypass (не использовать перегруженный сервер адаптации), wait (ожидать уменьшения числа соединений), force(игнорировать превышение и продолжать использовать указанный сервер адаптации)
- ipv6 - (только для icap) использовать IPv6 для соединения с сервером адаптации

Следующая команда задает именованное множество (с именем NAME) серверов адаптации:

adapt-set NAME {SRV,5}

Множество используется для повышения доступности адаптационного механизма - если один из серверов множества недоступен, сообщение передается следующему в списке серверу.

Следующая команда задает именованную цепочку (с именем NAME) серверов адаптации:

adapt-chain NAME {SRV,5}

Цепочка используется для множественной обработки сообщения разными серверами адаптации, т.е. сначала сообщение обрабатывается 1-м в цепочке сервером, затем результат обработки передается 2-му серверу и т.д.

Можно также задать правила адаптации, описывающие связь между серверами адаптации и ACL:

adapt-access <permit|deny> <NAME> {ACL,5}

Команда задает правило использования сервера адаптации (а также цепочки или множества серверов адаптации) NAME при попадании сообщения в указанные ACL.

Для задания прочих опций адаптации содержимого используйте следующую команду:

adapt <send-username HNAME [encode] | send-client-ip | icap-srvfail-limit N | icap-io-timeout N | icap-connect-timeout N | icap-service-revival-delay N>

Параметры:

send-username - задать имя HTTP-заголовка в котором будет пересылаться имя аутентифицированного пользователя; encode - закодировать имя пользователя алгоритмом BASE-64(send-client-ip - сообщать серверам адаптации IP адрес клиента)

- icap-connect-timeout - как долго ждать установления TCP-соединения с ICAP-сервером
- icap-service-revival-delay - как долго ждать после запроса опций ЮАР, прежде чем послать новый запрос опций ICAP.

32.12 Прочие настройки службы

Как долго ждать выполнения запроса на TCP-соединение (в сек.):

(config—service—proxy)# timeout connect \<VAL\>

По умолчанию: 60 сек.

Как долго ждать ответа на DNS-запрос (в сек.):

(config—service—proxy)# timeout dns \<VAL\>

По умолчанию: 30 сек.

Использовать IPv4 DNS-сервера в первую очередь и только затем IPv6 DNS-сервера: (config—service—proxy)# ipv6 dns4—first

32.13 Настройка журналов службы

Служба имеет систему журналов. Всего их существует 5 типов:

- cache: общий журнал службы; имеет пять уровней подробности журналирования, причем каждый следующий уровень включает сообщения предыдущего:
  - low: : сообщения о критических и фатальных ошибках;
  - middle: : сообщения о важных проблемах службы, предупреждения;
  - high : сообщения о небольших проблемах службы и высоко-уровневые операции службы;
- access : информация о доступе к службе и клиентских запросах;
- store : информация об объектах, сохраняемых в кэше и удаляемых из него;
- useragent : информация о клиентских браузерах (на основе заголовка User-Agent HTTP-запроса);
- referer : информация о доменах, посещаемых клиентами (на основе заголовка Referer HTTP-запроса).

При нормальной работе системы не следует устанавливать уровень cache журнала выше middle, чтобы не замедлять работу системы.

Обычно, если нет проблем в работе службы, достаточно бывает следующих журналов:

(config—service—proxy)# log access
(config—service—proxy)# log cache middle

Также можно отключить журнал доступа, оставив только общий журнал службы(cache).

Для просмотра журналов службы следует выполнить команду:

(config—service—proxy)# do show service proxy log

По умолчанию будет показан общий журнал службы.

Чтобы посмотреть, например, журнал доступа, следует выполнить команду:

(config—service—proxy)# do show service proxy log access

Иногда бывает необходимо вести журналы клиентских запросов, но делать это не для всех клиентов. Для этого можно задать ACL в команде log access. Рассмотрим пример:

(config—service—proxy)# acl vipl src 10.0.0.1 10.0.0.2
(config—service—proxy)# log access acls vipl

Данные правила разрешают вести журнал доступа только для узлов с адресами 10.0.0.1 ,**10.0.0.2**.

Существует возможность скрыть IP-адреса из log access: (config-service-proxy)# acl vipl src 10.0.0.1 10.0.0.2 (config-service-proxy)# log access 24 acls vipl

Данные правила говорят делать операцию AND между IP-адресом клиента и маской 255.255.255.0 (соответствует маске 24), т.е. младшая часть адреса будет обнулена в логе.

По умолчанию в лог access не пишутся детали запросов клиентов. Существует возможность отключить это параметром query-terms команды log access: (config-service-proxy)# log access query-terms

Рассмотрим формат строки журнала доступа на следующем примере:

16/Apr/2012:16:50:32 +0400 555 192.168.33.116 TCP_MISS/200 250 GET

http://notify5.dropbox.com/subscribe? — DIRECT/199.47.216.147 text/plain

Рассмотрим поля строки журнала по порядку слева направо:

- 16/Apr/2012:16:50:32 +0400 - время (UTC) завершения обработки запроса службой;
- 555 - продолжительность (в милисекундах) обработки запроса;
- 192.168.33.116 - адерс клиента;
- TCP_MISS/200 - код результата обработки запроса службой (коды перечислены ниже) и код HTTP-ответа;
- 250 - полный размер в байтах HTTP-пакета (размер HTTP-заголовков и размер, указанный в Content-Length);
- GET -метод HTTP;
- http://notify5.dropbox.com/subscribe? - URIзапроса;
- DIRECT - код узла, обозначающий,что запрос перенаправляется службой напрямую на удаленный сервер; NONE - запрос не перенаправляется никуда;
- 199.47.216.147 - адрес удалённого сервера, в случае если код узла - DIRECT;
- text/plain - тип содержимого HTTP-ответа, берется из Content-Type заголовка HTTP-ответа.

Коды результата обработки запроса службой приведены ниже:

- TCP_HIT : в кэше найдена свежая копия ресура; копия возвращена клиенту;
- TCP_MISS : в кэше не найдена копия ресурса;
- TCP_REFRESH_HIT : в кэше найдена возможно несвежая копия ресурса; послан запрос валидации ресурса на УС; УС вернул ответ «Not Modified»,Т.е. изначально копия была свежа;
- TCP_REF_FAIL_HIT : в кэше найдена возможно несвежая копия ресурса; послан запрос валидации ресурса на УС; УС не вернул ответ или вернул непонятный службе ответ; служба возвратила клиенту возможно несвежую копию ресурса;
- TCP_REFRESH_MISS : в кэше найдена возможно несвежая копия ресурса; послан запрос валидации ресурса на УС; УС вернул возвратил обновленные данные ресурса, т.е. копия в кэше была действительно несвежая;
- TCP_CLIENT_REFRESH_MISS : в кэше найдена копия ресурса; клиентский запрос содержал «Cache-Control: no-cache»; запрос перенаправлен УС, т.е. копия в кэше была принудительно обновлена;
- TCP_IMS_HIT : клиент прислал запрос валидации ресурса, т.к. он его уже имеет, при помощи IfModifiedSince; в кэше найдена более свежая копия; копия возвращена клиенту;
- TCP_SWAPFAIL_MISS : в кэше найдена свежая копия ресура; служба не смогла загрузить копию из своего кэша; послан запрос на УС,как будто это был кэш-промах;
- TCP_NEGATIVE_HIT : отрицательный HTTP-ответ (например «Connection refused» или «404 Not Found») был ранее закэширован; клиенту возвращён закэшированный отрицательный ответ;
- TCP_MEM_H1T : в памяти найдена свежая копия ресурса и возвращена клиенту;
- TCP_DENIED : запрос клиента запрещен из-за правил доступа (http-access или http-reply-access);
- TCP_OFFLINE_HIT : если включен режим offline, любой ресурсы,найденный в кэше возвращается клиенту,без проверки на свежесть;
- NONE - другой результат; используется при разного рода ошибках в запросе, например непонятный URI ресурса.

32.14 Работа со службой

Для просмотра статуса и правильности настройки службы следует выполнить команду:

(config—service—proxy)# do show service proxy status

Строка «Info: config is ok» говорит о том, что все в порядке. Любые строки, начинающиеся с «Error: » говорят об ошибках.

Чтобы удалить объект из кэша службы по его URI, следует выполнить команду:

(config—service—proxy)# do service proxy remove—object http://example.com/pic.jpg

Команда будет выполнена, только если служба включена.

Очистка всего кэша:

(config—service—proxy)# do service proxy remove—object all

Команда будет выполнена, только если служба отключена.

Обновление объекта в кэше:

(config—service—proxy)# do service proxy reload—object http://example.com/pic.jpg

Команда будет выполнена, только если служба включена.

Чтобы посмотреть различную информацию о работе службы:

(config—service—proxy)# do show service proxy info

Существует много типов информации, которую можно посмотреть, используя данную команду. Эта команда описана более подробно в следующем подразделе «Мониторинг службы».

Экспортировать сертификат службы как есть, либо в DER-формате (для импорта в браузеры клиентов):

(config—service—proxy)# do service proxy export FILE [der]

32.15 Мониторинг службы

Служба прокси Dionis-NX имеет мощную систему мониторинга своего состояния. Мониторинг службы осуществляется различными подкомандами команды show service proxy info.

Рассмотрим данные подкоманды.

32.15.1 Общая информация

Общая информация о работе службы:

show service proxy info

Будучи вызвана без параметров, эквивалентна команде:

show service proxy info general

Формат вывода данной команды:

- Squid Object Cache - версия службы;
- Start Time - время запуска службы;
- Current Time - текущее время;
- Connection information for squid - информация о соединениях и клиентах:
  - Number of clients accessing cache - число обслуженных клиентов; клиенты различаются по их IP-адресам;
  - Number of HTTP requests received - число полученных HTTP-запросов;
  - Request failure ratio - соотношение неуспешных запросов (ошибка в TCP соединении, ошибка DNS или аппаратная сетевая ошибка) к успешным;
  - Average HTTP requests per minute since start - среднее число HTTP запросов в минуту;
  - Select loop called - среднее число вызовов select()/poll() и среднее время между ними;
- Cache information for squid - информация о кэше:

- Hits as % of all requests - процент кэш-попаданий во всех запросов за последние 5 и 60 минут;
- Hits as % of bytes sent - процент трафика кэш-попаданий во всем трафике за последние 5 и 60 минут;
- Memory hits as % of hit requests - процент кэш-попаданий в памяти (TCP_MEM_H1T) во всех кэш-попаданиях;
- Disk hits as % of hit requests - процент кэш-попаданий на диске (TCP_H1T) во всех кэш-попаданиях;
- Storage Swap size - размер (Кб) данных в кэше на диске;
- Storage Swap capacity - процент заполнения данными кэша на диске;
- Storage Mem size - размер (Кб) данных в кэше в памяти;
- Storage Mem capacity - процент заполнения данными кэша в памяти;

Median Service Times (seconds) - среднее время выполнения операций (в секундах):

- HTTP Requests (All) - среднее время выполнения запроса;
- Cache Misses - среднее время выполнения запроса при кэш-промахе;
- Cache Hits - среднее время выполнения запроса при кэш-попадании (запросы с результатом TCP_H1T, TCP_MEM_H1T);
- Near Hits - среднее время выполнения запроса при кэш-обновлении (запросы с результатом TCP_REFRESH_HIT);
- Not-Modified Replies - среднее время выполнения запроса If-modified-since (запросы с результатом TCP_IMS_HIT);
- DNS Lookups - среднее время выполнения DNS-разрешения (прямого и обратного);

Resource usage for squid - статистика использования ЦПУ и памяти службой (значения времени - в секундах):

- UP Time - время работы службы в секундах;
- CPU Time - процессорное время,использованное службой;
- CPU Usage - процент использования процессора службой; отношение CPU Time к UP Time;
- CPU Usage, 5 minute avg - аналогично предыдущему, но за последние 5 минут;
- CPU Usage, 60 minute avg - аналогично предыдущему, но за последние 60 минут;
- Process Data Segment Size via sbrk() - размер сегмента данных службы (Кб);
- Maximum Resident Size - максимальный размер памяти, доступный службе;
- Page faults with physical i/o - число ошибок «Отсутствие страницы в памяти»;

Memory usage for squid via mallinfo() - статистика использования памяти службой:

- Total space in arena - общий объем памяти,выделенный службе;
- Ordinary blocks,Small blocks и др. - прочая информация о памяти возвращаемая функцией mallinfo();

Memory accounted for - информация по учету выделенной памяти (размер указан в Кб):

- Total accounted - общий объем отслеживаемой памяти;
- memPool accounted - общий объем отслеживаемой памяти пулов (стуктур фиксированного размера);
- memPool unaccounted - общий объем неотслеживаемой памяти для пулов;
- memPoolAlloc calls - число вызовов функции memPoolAlloc, выделяющей пулы;

- memPoolFree calls - число вызовов функции memPoolAlloc, освобождающей пулы;

- File descriptor usage for squid - статистика использования файловых дескрипторов (ФД):
  - Maximum number of file descriptors - максимально возможное число ФД для службы;
  - Largest file desc currently in use - максимальное число ФД, задействованных на данный момент;
  - Number of file desc currently in use - число ФД, используемых в настоящее время;
  - Files queued for open - число файлов в очереди на открытие (ненулевое значение возможно только для типа кэша aufs);
  - Available number of file descriptors - доступное число ФД;
  - Reserved number of file descriptors - зарезервированное число ФД;
  - Store Disk files open - число открытых файлов в настоящий момент;

- Internal Data Structures - статистика мест хранения объектов:
  - StoreEntries - число закэшированных службой объектов; каждый объект потребляет около 100 байт памяти службы (пул StoreEntry; см. далее info mem);
  - StoreEntries with MemObjects - число объектов, закэшированных в памяти, и объектов, к которым обращаются в настоящий момент;
  - Hot Object Cache Items - число объектов, закэшированных в памяти;
  - on-disk objects - число объектов закэшированных на диске.

32.15.2 Активные соединения

Следующая команда показывает список активных соединений на данный момент времени:

show service proxy info active—requests

Формат вывода команды:

- Connection - адрес памяти структуры соединения;
- FD - дексриптор сокета для TCP соединения, после которого следует объем принятых и переданных через него данных;
- FD desc - краткое описание сокета, обычно это URI;
- in - адрес памяти буфера приема, смещение следующих принятых данных и размер буфера приема;
- peer - удаленный сокет для TCP-соединения (для режима перерывания - это сокет клиента);
- te - локальный сокет для TCP-соединения удаленного сервера (для режима перерывания - это сокет клиента);
- nrequests - число запросов, полученных по данному соединению;
- defer - осуществляется ли отложенное чтение на сокете соединения;
- uri - запрашиваемый URI из запроса клиента;
- log_type - статус кэша по данному запросу - то, что появится в журнале доступа после завершения обработки запроса;
- out.offset - смещение по которому запрашиваются данные (имеет смысл во время скачивания больших файлов);
- out.size - размер данных в ответе;

- req_sz - размер HTTP-запроса клиента;
- entry - адрес памяти структуры хранения (StoreEntry) и ее хэш;
- start - сколько секунд назад инициировано соединение.

32.15.3 AUFS

Следующая команда выводит данные по асинхронной обработке запросов при типе кэша aufs:

```
# show service proxy info aufs
```

Формат вывода команды:

- open, close, read, write и т.д. - счетчики асинхронно выполненных файлов операций (открытие, закрытие, чтение, запись и т.д.);
- Threads status: число асинхронных файловых операций;
- queue: размер очереди запросов; при превышении 80 (5*16, где 16 - число асинхронных потоков) - будет выдано предупреждение о загруженности системы.

32.15.4 Аутентификация

Следующая команда показывает статистику по аутентификации пользователей:

```
# show service proxy info auth
```

Имеет смысл только при включённой аутентификации пользователей.

32.15.5 Клиенты

Следующая команда показывает статистику по по клиентам службы:

```
# show service proxy info clients
```

Формат вывода команды:

- Address: IP-адрес клиента;
- Name: FQDN-имя клиента;
- Currently established connections: число открытых соединений клиента и службы;
- HTTP Requests: число HTTP запросов клиента;
- TCP_HIT,TCP_MISS,...: статистика результатов запросов.

32.15.6 Трафик и ресурсы

Следующая команда показывает статистику по трафику и ресурсам:

show service proxy info counters

Формат вывода команды:

- sample_time - время последнего расчета счетчиков; расчет производится не реже, чем один раз в минуту;
- client_http.requests - число HTTP-запросов, полученных от клиентов;
- client_http.hits - число кэш-попаданий в ответ на HTTP-запросы клиентов; соответствует числу строк с типом результата TCP_H1T в журнале доступа;
- client_http.errors - число клиентских транзакций, приведших к ошибке;
- client_http.kbytes_in - объем HTTP-трафика в Кб, полученного от клиентов (HTTP-запросы);
- client_http.kbytes_out - объем HTTP-трафика в Кб, переданного клиентам (HTTP-ответы);
- client_http.hit_kbytes_out - объем HTTP-трафика в Кб, переданного клиентам (HTTP-ответы) в результате кэш-попаданий, включая ответы с кодом 304 (Not Modified);
- server.all.requests - число запросов, переданных на УС;
- server.all.errors - число запросов на УС, приведших к ошибке;
- server.all.kbytesjn - объем трафика в Кб, полученого с УС;
- server.all.kbytes_out - объем трафика в Кб, переданного на УС;
- server.http... - аналогично server.all, но только для HTTP-запросов;
- server.ftp... - аналогично server.all, но только для FTP-запросов;
- server.other... - аналогично server.all, но только для прочих запросов (Gopher, WAIS, SSL);
- page_faults - число ошибок «Обращение к отсутствующей странице»;
- selectjoops - число раз вывзова seclect()/poll() в главном цикле ввода-вывода службы;
- cpu_time - накопленное время CPU в секундах;
- wall_time - время, прошедшее с последнего расчета счетчиков;
- swap.outs - количество объектов(файлов), записанных в дисковый кэш службы;
- swap.ins - количество объектов(файлов), считанных с дискового кэша;
- swap.files_cleaned - количество объектов(файлов), удаленных периодической процедурой очистки;
- aborted_requests - число отмененных запросов на УС, произошедших из-за отмены их клиентами.

32.15.7 Пулы задержки

Следующая команда показывает статистику по пулам задержки (в байтах):

show service proxy info delay

Формат вывода команды:

Aggregate/Network/Individual - тип пула (общий/сетевой/индивидуальный);

- Max - размер пула;
- Restore(Rate) - объем данных, добавляемых к пулу каждую секунду;
- Current - текущий объем пула; узлы опредлеляются последним значимым октетом.

Пример для agr: Current: 12345

Пример для agr-host24: Current: 1:1234 2:5678, где 1,2 - последний октет IP-адреса хоста.

Пример для agr-net24-hostl6: Current [Network 5]: 1:1234 2:5678, где 1,2 - последний октет IP-адреса хоста, 5 - предпоследний октет IP-адреса сети, например 10.0.5.0/24.

32.15.8 Открытые файлы

Следующая команда показывает статистику по файлам, открытым службой:

show service proxy info filedescriptors

Формат вывода команды:

- File - дескриптор файла;
- Type - тип файла: File - для дискового кэша, лог-файла; Pipe - канал для IPC, межпроцессного взаимодействия; Socket - соект для свзяи с клиентами, УС и IPC;
- Tout - таймаут для файлов типа Socket; такой файл за кроется, если по нему не будет активности по истечении указанного таймаута;
- Nread - количество байт, прочитанных из файла;
- Nwrite - количество байт, записанных в файл;
- Remote Address - для файлов типа Socket это удаленный TCP-адрес соединения (удаленный сокет);
- Description - описание файла для типов Socket/Pipe или путь к файлу для типа File.

32.15.9 X-Forwarded-For-заголовки

Следующая команда показывает полученные X-Forwarded-For заголовки:

show service proxy info forward —headers

X-Forwarded-For-заголовок содержит IP-адреса клиентов,для которых данный HTTP-запрос был ретранслирован прокси-сервером. Например,если служба получает HTTP-запрос от узла, в котором содержится X-Forwarded-For-заголовок с адресом 1.1.1.1, это значит, что служба получила данный запрос от другого прокси-сервера, который, в свою очередь, получил этот запрос от узла 1.1.1.1.

32.15.10 Via-заголовки

Следующая команда показывает полученные Via-заголовки:

show service proxy info via—headers

Via-заголовок содержит имена и, возможно, порты и другую идентифицирующую информацию прокси-сереверов, через которые прошел полученный службой HTTP запрос.

32.15.11 HTTP-заголовки

Следующая команда показывает статистику по HTTP-заголовкам:

show service proxy info http—headers

Формат вывода команды:

- Header Stats: request - статистика по HTTP-запросам;
  - Field type distribution - распределение HTTP-заголовков:
    * id - внутренний идентификатор;
    * name - имя заголовка;
    * count - количество заголовков;
    * #/header - частота появления заголовков; например, частота 1.0 означает, что заголовок присутствует в каждом запросе;
  - Cache-control directives distribution - распределение директив кэширования заголовка Cache-Control:
    * id - внутренний идентификатор;
    * name - имя директивы;
    * count - количество директив;
    * #/cc_field - частота появления директив кэширования;
  - Number of fields per header distribution - распределение количества HTTP-заголовков:
    * id - внутренний идентификатор;
    * #flds - количество заголовков;
    * count - число запросов с указанным в поле #flds количеством заголовков;
    * %total - доля запросов с указанным в поле #flds количеством заголовков в общем числе запросов;
- Header Stats: reply - статистика по HTTP-ответам. Поля аналогичны полям для Header Stats: request;
- Http Fields Stats (replies and requests) - статистика по HTTP-запросам и HTTP-ответам:
  - id - внутренний идентификатор;
  - name - имя заголовка;
  - #alive - количество заголовков типа name, хранящихся в данный момент в памяти (заголовки активных содеинений и для объектов,хранящихся в кэш-памяти службы;
  - %err - процент ошибочных заголовков типа name;
  - %repeat - доля запросов/ответов, с повторяющимися однотипными заголовками типа name;
- Headers Parsed - число обработанных HTTP-запросов/ответов;
- Hdr Fields Parsed - число обработанных HTTP-заголовков.

32.15.12 DNS-клиент

Следующая команда показывает статистику по внутреннему DNS-клиенту службы:

show service proxy info idns

Формат вывода команды:

- The Queue - очередь неразрешенных DNS-запросов:
  - ID - внутренний идентификатор;
  - SIZE - размер запроса;
  - SENDS - число попыток запроса;
  - FIRST SEND / LAST SEND - промежуток времени между последним и первым запросом;
- Nameservers - статистика запросов/ответов на сервера имен:
  - IP ADDRESS - адрес сервера имен;
  - # QUERIES - число посланных на сервер имен запросов;
  - # REPLIES - число полученных от сервера имен ответов;
- Rcode Matrix - статистика DNS-ответов:
  - RCODE - код ответа:
    * 0 - успешный ответ;
    * 1 - сервер имен не смог понять запрос (Format Error);
    * 2 - проблема с сервером имен (Server Failure);
    * 3 - доменное имя не существует (Name Error);
    * 4 - сервер имен не поддерживает указанный тип запроса (Not Implemented);
    * 5 - отказ в обработке запроса из-за политики безопасности сервера (Refused);
  - ATTEMPT1 - число одинарных повторных попыток запроса в результате получения RCODE=2;
  - ATTEMPT2 - число двойных повторных попыток запроса в результате получения RCODE=2;
  - ATTEMPT3 - число тройных повторных попыток запроса в результате получения RCODE=2;
- Search list - список доменов,используемых при разрешении имен.

32,15.13 DNS-кэш

Следующая команда показывает статистику по внутреннему DNS-кэшу службы:

show service proxy info ipcache

Формат вывода команды:

- IP Cache Statistics - статистика по DNS-кэшу:

- IPcache Entries In Use - количество записей кэша, используемых в настоящее время;
- IPcache Entries Cached - количество записей кэша;
- IPcache Requests - число DNS-запросов;
- IPcache Hits - число DNS-запросов, разрешенных из кэша DNS-службы;
- IPcache Negative Hits - число DNS-запросов, негативно разрешенных из кэша DNS-службы;
- IPcache Numeric Hits - число запросов разрешения адреса в имя, разрешенных из кэша DNS-службы;
- IPcache Misses - число DNS-запросов, разрешенных через DNS-сервер,а не из кэша DNS-службы;
- IPcache Retrieved A - число полученных A-записей;
- IPcache Retrieved AAAA - число полученных AAAA-записей;
- IPcache Retrieved CNAME - число полученных CNAME-записей;
- IPcache CNAME-Only Response - число полученных только CNAME-записей;
- IPcache Invalid Request - число неверных DNS-запросов;

• IP Cache Contents - кэш DNS-службы для наиболее популярных имен:

- Hostname - доменное имя;
- Fig - флаги: N - кэширование негативно разрешенного имени; H - разрешение пришло из статического назначения системы (см. ip resolver hosts);
- Istref - показывает, сколько секунд назад запись использовалась последний раз;
- TTL - время жизни записи в кэше (секунд);
- N(b) - N: число IP-адресов имени (адреса с суффиксом OK); Ь: число IP-адресов имени, которые недоступны в настоящее время (адреса с суффиксом BAD);
- последняя колонка - показывает IP-адреса с суффиксом OK или BAD (см. N(b) колонку).

32.15.14.1 Память

Следующая команда показывает статистику использования памяти службой:

```
# show service proxy info mem
```

Формат вывода команды:

• Largest pools stats - статистика по двум максимальным пулам фиксированных структур:

- Pool name - имя пула: StoreEntry - пул структур,создаваемая на каждый кэшированный объект; MD5-digest - пул хэшей ответа; All-pools - все пулы;
- Size - размер экзампляра стуктуры (байт);
- Number - число структур в пуле;
- TotSize - общий размер пула;
- HiSize - максимально наблюдаемый размер пула;

• Cumulative allocated volume - общий объем памяти, выделенной службе; часть ее может быть освобождена; учитывается только выделяемая память;
• Total Pools created - общее число созданных пулов;

- Pools ever used - число использованных по настоящий момент пулов;
- Currently in use - число используемых в настоящий момент пулов.

32.15.14.2 Свежесть

Следующая команда показывает статистику алгоритма проверки на свежесть:

show service proxy info refresh

Формат вывода команды:

• HTTP histogram - показывает распределение проверок на свежесть, приведших к решению о свежести объекта (Fresh/Stale - свежий/несвежий), при запросе его клиентом:

- Count - общее число проверок данного типа;
- %Total - доля проверок данного типа;
- Category - тип проверки на свежесть: —причины свежести—
  * Fresh: request max-stale wildcard - в запросе была директива max-stale, т.е. клиент хочет принять объект любой свежести;
  * Fresh: request max-stale value -в запросе была директива max-stale со значением больше времени,прошедшего с момента истечения времени жизни объекта (Expires);
  * Fresh: expires time not reached - момент истечения времени жизни объекта еще не наступил;
  * Fresh: refresh_pattern last-mod factor percentage - объект подпадает под refresh-правило; LM-фактор объекта меньше указанного в правиле LM-фактора;
  * Fresh: refresh_pattern min value - возраст объекта меньше указанного в значении min правила refresh, в которое попадает объект;
  * Fresh: refresh_pattern override expires - объект подпадает под refresh-правило с параметром override-expire;
  * Fresh: refresh_pattern override lastmod - объект подпадает под refresh-правило с параметром override-lastmod; —причины несвежести—
  * Stale: response has must-revalidate - запрос содержит директиву кэширования Cache-Control: must-revalidate;
  * Stale: changed reload into IMS - объект подпадает под refresh правило с параметром reload-into-ims;
  * Stale: request has no-cache directive - запрос содержит директиву кэширования Cache-Control: no-cache;
  * Stale: age exceeds request max-age value - в запросе была директива max-age со значением меньше возраста объекта;
  * Stale: expires time reached - момент истечения времени жизни объекта наступил;
  * Stale: refresh_pattern max age rule - возраст объекта больше указанного в значении max правила refresh, в которое попадает объект;
  * Stale: refresh_pattern last-mod factor percentage - объект подпадает под refresh-правило; LM-фактор объекта не меньше указанного в правиле LM-фактора;
  * Stale: by default - объект не подпадает ни под один критерий алгоритма проверки, в связи с чем признается несвежим по умолчанию;

* TOTAL - общее число проверок;

• On Store histogram - аналогично HTTP histogram, но для проверок на свежесть ответов УС для кэш-промахов.

32.15.16.1 Ретрансляция запросов

Следующая команда показывает статистику ретрансляции запросов УС:

show service proxy info requests

Формат вывода команды:

- Status - код HTTP-ответа:
  - lxx: Informational (информационные);
  - 2xx: Success (успешно);
  - 3xx: Redirection (перенаправление);
  - 4xx: Client Error (ошибка клиента);
  - 5xx: Server Error (ошибка сервера);
- try#l-10 - число попыток, предпринятых для получения ответа типа Status (try#l - одна попытка,...,try#10 - десять попыток).

32.15.16.2 Кэш

Следующая команда показывает статистику по кэшу:

show service proxy info storage

Формат вывода команды:

- Store Entries - число кэшированных объектов;
- Store Directory #0 - указывает тип кэша (aufs);
- FS Block Size - размер блока файловой системы (байт);
- First level subdirectories - число директорий первого уровня;
- Second level subdirectories - число директорий второго уровня;
- Maximum Size - максимальный размер кэша;
- Current Size - текущий размер кэша;
- Percent Used - текущая заполненность кэша;
- Filemap bits in use - сколько битов файловой карты использовано;
- Filesystem Space in use - объем места на диске, использованного службой;
- Filesystem Inodes in use - сколько использовано инодов;
- Flags: SELECTED - всегда значение SELECTED; значит что кэш в режиме чтения-запись; формат read-only для кэша не поддерживается;

- Removal policy - типа политики замены объектов в дисковом кэше: lru - для LRU, heap - для LFUDA, GDSF или HLRU;
- LRU reference age - показывает дату самого старого объекта кэша; только для политики замены объектов LRU.

32.15.18.1 Рекомендации по настройке

32.15.18.1 Размер дискового кэша

Размер дискового кэша определяется опытным путем в зависимости от нужд клиентов службы, их числа, а также размера доступной оперативной памяти.

Например при использовании 4Гб дискового кэша объем потребляемой памяти при полном заполнении кэша будет варьироваться в пределах 150-200М6.

33. Служба SNMP

Dionis-NX имеет службу SNMP.

Данная служба позволяет другим узлам получить SNMP-информацию о системе, а также отсылает другим узлам SNMP-нотификации о старте и остановке службы.

33.1 Общая настройка службы SNMP

Чтобы войти в режим настройки службы, следует выполнить команду:

(config)# service snmp

Настройка интерфейса и/или порта, который служба будет использовать для приема запросов и посылки ответов, нотификаций,zz например:

(config—service—snmp)# listen 192.168.0.1 udp

По умолчанию,используется UDP-порт 161 и любой локальный интерфейс, которому назначен IP-адрес.

Для работы с адресами IPv6 используется соответствующая команда **listen6.**

33.2 Настройка базовой SNMP информации

Настройка общей информации о системе. Например:

(config—service—snmp)# sysinfo location russia
(config—service—snmp)# sysinfo name routerl (config—
service—snmp)# sysinfo name admin@domain

Этими командами можно задать, соответственно, физическое местонахождение системы, имя системы и адрес электронной почты администратора системы.

33.3 Настройка правил доступа

Настройка правил, по которым другим узлам разрешено получать информацию о системе по SNMP-протоколу. Например:

(config—service—snmp)# acl pasl (config—
service—snmp)# acl pas2 1.2.3.4 (config—
service—snmp)# acl pas3 2.2.2.0/24

Рассмотрим приведенные в примере команды:

- первая команда: первый параметр команды (pasl) - это обязательный параметр, задающий пароль доступа, который должен использоваться узлом, желающим получить информацию о системе по протоколу SNMP. Адрес узла может быть любой, т.к. он не указан.
- вторая команда: только узел с адресом 1.2.3.4 и паролем доступа pas2 может получить информацию о системе по протоколу SNMP.
- третья команда: только узлы сети 2.2.2.0/24 и паролем доступа pas3 могут получить информацию о системе по протоколу SNMP.

Для работы с адресами IPv6 используется соответствующая команда **ac16.**

33.4 Настройка правил нотификаций

Настройка правил, по которым другим узлам разрешено получать нотификации от службы SNMP. Например:

(config—service—snmp)# notify pasl 1.2.3.4 (config—
service—snmp)# notify pas2 1.2.3.5:5555 v2 (config—
service—snmp)# notify pas3 1.2.3.6:5556 v2c tcp

Рассмотрим третью команду в примере.

Первый обязательный параметр команды (pas3) - это пароль доступа, который должен быть установлен в конфигурации SNMP-клиента, который хочет получать нотификации.

Второй обязательный параметр команды (1.2.3.6) - это IP-адрес клиента, которому разрешено посылать нотификации.

Остальные параметры - это порт, версия нотификаций и транспортный протокол. По умолчанию, используется UDP-порт 162 и версия нотификаций v2c.

Для работы с адресами IPv6 используется соответствующая команда **notify6.**

Нотификации, посылаемые по умолчанию (не требуется настройка): [12]

12NET-SNMP-AGENT-MIB:: nsNotifyShutdown - посылается при выключении службы;

- SNMPv2-MIB: :coldStart - посылается при включении службы;
- NET-SNMP-AGENT-MIB:: nsNotifyRestart - посылается при перезапуске службы (например, во время настройки уже запущенной службы).
- FACTOR-DIONISNX-NOTIFICATION-MIB: TdnxSomeAdminLoggedln - осуществлен вход первым администратором
- FACTOR-DIONISNX-NOTIFICATION-MIB: :fdnxAIIAdminLoggedOut - осуществлен выход последним администратором
- FACTOR-DIONISNX-NOTIFICATION-MIB: :fdnxLoginRetriesExceeded - превышено количество попыток входа (вместе с нотификацией отправляется имя пользователя, под которым была осуществлена попытка входа)
- FACTOR-DIONISNX-NOTIFICATION-MIB: :fdnxStartupConfigUpdated - стартовая конфигурация обновлена
- FACTOR-DIONISNX-CLUSTER-MIB: :fdnxClusterStateChanged - изменилось состояние кластера

Дополнительно можно включить следующие типы нотификаций (команда указана после описания нотификации):

- SNMPv2-MIB: :authenticationFailure - посылается при неуспешной аутентификации SNMP клиента (например, когда он послал серверу неверный пароль).
- IF-MIB::linkDown/IF-MIB::linkUp - посылается при включении/отключении интерфейса.

Включение дополнительных нотификаций осуществляется командой:

```
(config—service—snmp)# trap auth
(config—service—snmp)# trap iface
```

33.5 Работа со службой

Для запуска службы выполните команду:

```
(config—service—snmp)# enable
```

Для остановки службы выполните команду:

```
(config—service—snmp)# disable
```

Для просмотра журналов службы выполните команду:

```
(config—service—snmp)# do show service snmp log
```

34. SSH

В рамках системы Dionis-NX протокол SSH не считается защищенным и, соответственно, его использование недостаточно для установления доверенного канала передачи данных. Для создания доверенных каналов передачи данных должны использоваться крипто-туннели.

34.1 Сервер SSH

В Dionis-NX реализована возможность удалённого доступа к командному интерфейсу по протоколу SSH.

Чтобы разрешить удалённый доступ к данному узлу, необходимо активировать службу SSH следующими командами (из режима конфигурации):

(config)# service ssh (config—
service—ssh)# enable

По умолчанию служба будет принимать SSH-соединения на всех интерфейсах, на IPv4-адресах. По-умолчанию использование IPv6 отключено. Есть возможность настроить службу для приёма SSH-соединений на одном локальном IP-адресе и/или изменить TCP порт по умолчанию (22). Для этого необходимо указать опцию «listen». Например:

(config—service—ssh)# listen 192.168.1.1

или
(config—service—ssh)# listen 0.0.0.0 2222

Данная опция может быть очищена с помощью команды «no listen».

Для использования IPv6-адресов необходимо указать команду "listen6":

(config—service—ssh)# listen6 fd55:1: :ca60:ff:fe61:4177

Чтобы сервис ожидал соединений на любых IPv6-адресах, необходимо использовать следующую команду:

(config—service—ssh)# listen6 ::

Директива "listen6" может удалена из настроек с помощью команды "no listen6".

Также можно отдельно задать порт для приема соединений:

(config—service—ssh)# port 2222

В этом случае указанный порт будет использоваться для всех слушающих IP-адресов, если в соответствующей команде «listen» порт не задан явно. Допустимо указание нескольких слушающих портов. Очистить опцию можно с помощью команды «no port»:

(config—service—ssh)# no port 2222

По умолчанию возможен доступ извне только к учётной записи «cli», которая обеспечивает доступ к командам непривилегированного режима, и для входа в привилегированный режим будет необходимо ввести команду «enable». Чтобы ускорить доступ к привилегированному режиму через учётную запись «adm», необходимо включить опцию:

(config—service—ssh) # permit—adm—login

Данная опция может быть очищена командой «по permit-adm-login».

По-умолчанию сервис SSH не использует DNS для получения имени удаленного хоста. Если необходимо получать имя удаленного хоста и затем проверять, что обратное преобразование имени в IP-адрес совпадает с исходным IP-адресом, то следует использовать директиву "use-dns":

(config—service—ssh) # use—dns

Проверка имен может приводить к большим задержкам при подключении клиентов к SSH-серверу. Это происходит при некорректном или не полностью сконфигурированном DNS. На DNS может быть не настроена обратная зона для сопоставления имени IP-адресу. Для отключения проверки используется команда "no use-dns".

Следует помнить, что если настройки службы редактируются при работающей (активированной) службе, необходимо перезапустить службу, чтобы настройки вступили в силу:

(config—service—ssh)# disable
(config—service—ssh)# enable

либо

(config—service—ssh)# reload

Чтобы остановить службу и удалить все настройки, нужно выполнить команду режима конфигурации:

(config)# no service ssh

34.1.1 Контроль доступа

Для контроля доступа к сервису ssh предусмотрены команды «allow» и «deny». Контроль осуществляется на основании имени пользователя и удаленного адреса, с которого происходит попытка соединения (user[@host]). В имени пользователя или удаленном адресе могут быть использованы шаблоны («*» - ноль или более символов, «?» - ровно один символ).

Команды «allow» и «deny» соответственно разрешают или запрещают доступ для указанных пользователей. Команда «deny» имеет более высокий приоритет, чем команда «allow». Команд «allow» и «deny» может быть несколько. Если используется команда «allow», то доступ будет предоставлен только указанным пользователям (и адресам). Всем остальным пользователям в доступе будет отказано. И наоборот, если используется команда «deny», то доступ будет запрещен указанным пользователям (и адресам). Всем остальным пользователям доступ будет разрешен.

Если команды «allow» и «deny» не используются - доступ по-умолчанию разрешен. Обратите внимание на то, что администратор adm имеет полный контроль над системой. Возможность удаленного доступа администратора является важным параметром безопасности и контролируется

дополнительной командой режима конфигурирования сервиса ssh - «permit-adm-login» (команда описана выше). Для учетной записи администратора команды «allow» и «deny» лишь добавляют возможности по контролю доступа на основании удаленного адреса, с которого происходит попытка соединения.

Удаленный адрес может быть не указан. В этом случае будет учитываться только имя пользователя.

Следующая команда разрешит доступ оператору cli и запретит всем остальным (в данном случае adm):

(config—service—ssh) # allow cli

Следующая команда разрешит доступ оператору cli и администратору adm с удаленного адреса 192.168.2.3. Доступ с других адресов запрещен:

(config—service—ssh) # allow *@192.168.2.3

Следующая команда запретит доступ администратору adm из подсети 192.168.2.0/24. Доступ с других адресов разрешен. Доступ оператору cli разрешен с любых адресов:

(config—service—ssh) # deny adm@192.168.2.*

34.2 Клиент SSH

В Dionis-NX также реализован клиент SSH для удалённого доступа к другим узлам Dionis-NX. Команда доступна как из привилегированного, так и из непривилегированного режима. Формат команды:

> ssh <user> <host> [<port>]

Для узлов Dionis-NX в качестве <user> можно указывать учётные записи «cli» или «adm».

При обращении на удаленные узлы, информация об этих узлах заносится в список известных хостов (know-hosts). При повторном обращении на тот же удаленный узел, сравнивается сохраненный (при первом обращении) ключ удаленного хоста и текущий ключ удаленного хоста. Если ключи не совпадают, соединение считается небезопасным и связь не устанавливается. Это сделано для предотвращения подмены удаленного хоста. Для просмотра списка известных хостов используется следующая команда:

Router# show ssh known —hosts

Если администратор знает, что удаленный хост был легально заменен или изменился ключ удаленного хоста, он может удалить из списка известных хостов информацию о таком узле. Это позволит установить связь и сохранить новый ключ хоста в списке известных хостов:

Router# clear ssh known —hosts 192.168.1.33

также можно полностью очистить список известных хостов:

Router# clear ssh known —hosts all

34.3 Соединение без использования паролей

Если соединение с удаленным узлом по протоколу SSH является частой операцией, неудобно каждый раз вводить пароль. Для установления соединения без использования паролей могут использоваться открытые ключи.

Если администратор, находясь на хосте А, хочет устанавливать соединение с хостом Б, то на хосте А он должен создать закрытый и открытый ключи:

```
A# ssh key generate
```

Затем открытый ключ должен быть отправлен на хост Б. В случае, если хост Б работает под управление системы Dionis-NX, администратор может выполнить команду:

```
A# ssh key export host adm 192.168.1.2
```

Предполагается, что хост Б имеет IP-адрес 192.168.1.2 и администратор желает устанавливать соединение используя учетную запись adm на удаленном хосте. Последним аргументом может быть указан удаленный порт. В данном случае порт не указан и, соответственно будет использован стандартный номер для SSH протокола - 22. После получения ключа, хост Б добавит этот ключ в список авторизованных ключей (authorized-keys). Все последующие соединения с хоста А на хост Б будут происходить без использования пароля.

В случае, если хост Б работает под управлением операционной системы отличной от Dionis-NX, администратор может записать созданный открытый ключ в файл на хосте А:

```
A# ssh key export file open.key
```

Где open.key - произвольное имя файла, в который будет сохранен открытый ключ. После этого файл может быть скопирован на хост Б любым удобным способом и добавлен в список авторизованных ключей. Например в Linux-системах файл с авторизованными ключами находится в домашней директории пользователя и называется ~/.ssh/authorized_keys.

Если же на хосте А установлена система, отличная от Dionis-NX, администратор может скопировать файл с открытым ключом с хоста А на хост Б любым удобным способом, а затем добавить полученный ключ в список авторизованных ключей на хосте Б:

```
B# ssh key import file open.key
```

Где open.key - имя файла с открытым ключом.

Администратор может просмотреть список авторизованных ключей используя команду:

```
B# show ssh authorized —keys
```

Для более детального вывода, можно указать опцию «verbose».

```
B# show ssh authorized —keys verbose
```

Если какой-либо ключ больше не требуется, он может быть удален из списка авторизованных ключей:

```
B# clear ssh authorized —keys adm@A
```

Где «adm@A» идентифицирует ключ в списке авторизованных ключей и является последним полем при выводе на экран детального списка авторизованных ключей. Можно удалить сразу все ключи из списка авторизованных ключей:

```
B# clear ssh authorized—keys all
```

34.4 Передача файлов

Протокол SSH может использоваться для передачи файлов. Если администратор хочет получить файл с удаленного хоста, он может выполнить следующую команду:

```
Router# ssh get petrov 192.168.1.2 /tmp/test.txt
```

Где «petrov» - учетная запись на уделенном хосте, «192.168.1.2» - IP-адрес удаленного хоста, «/tmp/test.txt» - путь к файлу. Для отправки файла на удаленный хост, администратор может выполнить команду:

```
Router# ssh put test.txt petrov 192.168.1.2
```

Описанные команды соответствуют команде scp в Linux-системе. Файлы с Linux-системы (и других систем, поддерживающих протокол SSH) могут быть отправлены на хост с системой Dionis-NX.

35. Telnet

Система Dionis-NX имеет службу Telnet, реализующую сетевой протокол уровня приложений для создания текстового интерфейса по сети.

35.1 Настройка

Для входа в режим конфигурации службы следует выполнить команду:

(config)# service telnet

Для настройки сокета, на котором служба будет ожидать Telnet-соединение, выполните:

(config—service—telnet)# listen 192.168.0.1 1023 (config—service—telnet)# listen6 fe80: :be5f:f4ff:fec4:7edd

Если порт не указан, по умолчанию используется порт 23. В данном случае мы предписываем службе принимать запросы на адресе 192.168.0.1 и использовать порт 1023, и на адресе fe80::be5f:f4ff:fec4:7edd и использовать порт 23.

Возможно указание множества сокетов для принятия соединений.

По умолчанию,если не указано ни одной опции listen и listen6, служба ожидает соединения на всех доступных IPv4 адресах и на порту 23.

Чтобы включить службу выполните

(config—service—telnet)# enable

Чтобы выключить службу выполните

(config—service—telnet)# disable

Если служба включена, для изменения ее опций выполните:

- измените нужную опцию
- выполните команду disable
- выполните команду enable

36. Сервис DIWEB

В Dionis-NX реализована возможность удалённого доступа к визуальному Web-интерфейсу по протоколу HTTP. Сервис позволяет конфигурировать часть функций системы.

В текущей версии Dionis-NX работа через Web-интерфейс возможна только с помощью учетной записи "adm". При этом учетная запись "adm" должна иметь статус "supervisor", т.е. иметь полный доступ к возможностям системы.

Чтобы разрешить удалённый доступ к данному узлу, необходима активировать службу diweb командами (из режима конфигурации):

```
(config)# service diweb
(config—service—web)# enable
```

По умолчанию служба будет принимать соединения на всех интерфейсах. Есть возможность настроить службу для приёма HTTP-соединений на одном локальном IP-адресе и/или изменить порт по умолчанию (80). Для этого необходимо указать опцию «listen». Например:

```
(config—service—web)# listen 192.168.1.1
```

или

```
(config—service—web)# listen 0.0.0.0 8080
```

Данная опция может быть очищена с помощью команды «no listen».

Аналогично команде "listen" работает команда "listen6" для IPv6 адресов. Например:

```
(config—service—web)# listen6 ::
```

или

```
(config—service—web)# listen6 :: 8080
```

Опция очищается с помощью команды «no listen6».

Доступно ssl-шифрование, включается командой "ssl":

```
(config—service—web)# ssl
```

Опция очищается командой "no ssl".

Следует помнить, что если настройки службы редактируются при работающей (активированной) службе, необходимо перезапустить службу, чтобы настройки вступили в силу:

```
(config—service—web)# disable
(config—service—web)# enable
```

Чтобы остановить службу и удалить все настройки, нужно выполнить команду режима конфигурации:

```
(config)# no service diweb
```

37. Сервис XMPP

В Dionis-NX реализован сервис xmpp (Jabber)

Приведем пример минимально необходимых настроек для запуска сервиса xmpp:

```
(config)# service xmpp
(config—service—xmpp)# host 192.168.1.1
```

Таким образом мы настраиваем сервер принимать соединения на виртуальном хосте 192.168.1.1. При этом вместо IP-адреса может быть указано доменное имя.

Чтобы запустить сервер необходимо выполнить команду

```
(config—service—xmpp)# enable
```

Сервер будет ожидать соединения на порту 5222, «plain»-аутентификация и шифрование отключены.

Если потребуется поддержка «plain»-аутентификации, необходимо выполнить комманду в настройках сервиса:

```
(config—service—xmpp—192.168.1.1)# allow—plain—auth
```

Для включения ssl/tls шифрования потребуется выполнить команду:

```
(config—service—xmpp—192.168.1.1)# ssl
```

Отключение отдельного виртуального хоста производится командой «off»:

```
(config—service—xmpp—192.168.1.1)# off
```

Разрешение регистрации пользователей включается командой «allow-client-reg»:

```
(config—service—xmpp—192.168.1.1)# allow—client—reg
```

Данные опции могут быть очищены с помощью команд «по allow-plain-auth», «по ssl», «оп» и "no allow-client-reg" соответственно.

По умолчанию служба будет принимать соединения на всех интерфейсах. Есть возможность настроить службу для приёма соединений на нескольких локальных IP-адресах и/или изменить порт по умолчанию (5222). Для этого необходимо требуемое число раз указать опцию «listen» и/или «port». Например:

```
(config—service—xmpp)# listen 192.168.1.1
    и/или
(config—service—xmpp)# port 5223
```

Для протокола IPv6 используйте вместо «listen» «listen6». Например:

```
(config—service—xmpp)# listen6 2001:db8::2
```

Все эти опции могут быть очищены с помощью команд «no listen 192.168.1.1», «no port 5223» и «no listen 2001:db8::2» соответственно.

Следует помнить, что если настройки службы редактируются при работающей (активированной) службе, необходимо перезапустить службу, чтобы настройки вступили в силу:

(config—service—xmpp)# disable
(config—service—xmpp)# enable

Чтобы остановить службу и удалить все настройки, нужно выполнить команду режима конфигурации:

(config)# no service xmpp

Для создания пользователя "user" в enable режиме требуется ввести комманду:

service xmpp account create 192.168.1.1 user passwd

Для удаления пользователя "user" в enable режиме требуется ввести комманду:

service xmpp account remove 192.168.1.1 user

Для изменения пароля пользователя "user" в enable режиме требуется ввести комманду:

service xmpp account password 192.168.1.1 user newpasswd

38. Служба netperf

38.1 Настройка службы netperf режима configure

Позволяет запустить службу измерения пропускной способности сети.

Для измерения пропускной способности между узлами А и Б, на которых установлены изделия Dionis-NX:

- запустите данный сервис на узле А;
- используйте команду netperf режима enable на узле Б, в которой в качестве первого параметра задайте IP-адрес узла А.

Чтобы войти в режим конфигурации службы, следует использовать команду:

(config)# service netperf

Если необходимо, порт, на котором будет запущена служба, может быть настроен при помощи команды:

(config—service—netperf)# listen 12345
(config—service—netperf)# listen6

По умолчанию, если не указано ни одной опции listen и listen6, служба ожидает соединения на всех доступных IPv4 адресах и на порту 12865. В данном случае мы предписываем службе принимать запросы на адресах IPv4 и порту 12345 и на адресах IPv6 и порту 12865

Включить службу следует командой:

(config—service—netperf)# enable

Для выключения службы следует выполнить команду:

(config—service—netperf)# disable

38.2 Команда netperf режима enable

netperf <IP[:PORT] | IPv6[:PORT] | HOST[:PORT]> <TEST> [time TIME] [cycles NUM] [local-sock-size LSZ] [remote-sock-size RSZ] [message-size MSZ] [verbose <low|average|hi>] [nodelay <both|local|remote>]

Данная команда осуществляет тестирование пропускной способности сети. Имеет два обязательных параметра:

IP или IPv6 или HOST - IPv4 или IPv6 адрес или имя узла, на котором запущена служба netperf;
TEST - тип теста: tcp или udp.

Остальные параметры являются необязательными:

- PORT - задает порт, на котором запущена служба netperf (по умолчанию 12865);
- TIME - время одного цикла теста в секундах (по умолчанию: 10);
- NUM - число циклов теста (по умолчанию: 1);
- LSZ - размер локального буфера сокета (по умолчанию: 64000 байт);
- RSZ - размер локального буфера сокета (по умолчанию: 64000 байт);
- MSZ - размер сообщения;
- verbose - задает уровень подробности выдачи команды (низкий,средний или высокий);
- nodelay - включает на локальном, удаленном или обоих сокетах опцию TCP_NODELAY, которая может ускорить передачу большого количества данных маленького размера (частые посылки).

39. Служба IPERF

39.1 Настройки службы iperf режима configure

Позволяет запустить службу iperf для измерения пропускной способности сети.

Для измерения пропускной способности между узлами А и Б, на которых установлены изделия Dionis-NX:

- следует запустить данный сервис на узле А;
- затем необходимо выполнить команду iperf режима enable на узле Б; в этой команде в качестве первого параметра следует задать IP-адрес узла А.

Чтобы войти в режим конфигурации службы, следует выполнить команду:

(config)# service iperf

Следует задать тип теста и, если необходимо, пару IP-адрес и порт, на котором будет запущена служба:

(config—service—netperf)# listen udp 10.0.0.1 12345

По умолчанию: listen tcp 0.0.0.0 5001

Для задания IPv6 адреса используется команда **listen6:**

(config—service—netperf)# listen6 tcp fe80::be5f:f4ff:fec4:7edd 12346

Другие варианты типов теста: udp (UDP-тест), udp-single(oflHonoT04Hbm UDP-тест).

Включить службу следует командой:

(config—service—netperf)# enable

Для выключения службы следует выполнить команду:

(config—service—netperf)# disable

39.2 Команда iperf режима enable

iperf <IP[:PORT] | IPv6[:PORT]> <TEST> [SPEED] [time TIME] [cycles NUM] [threads NT] [size SOCKSZ] [tos Crelaiablity|throughput|delay|TOS>] [window-size WIN] [mss MSS] [bidirectional BPORT simultaneous|individual]

Данная команда осуществляет тестирование пропускной способности сети. Имеет два обязательных параметра:

IP - IPv4 или IPv6 адрес узла, на котором запущена служба iperf;

TEST - тип теста: tcp или udp.

Остальные параметры являются необязательными:

- PORT - задает порт, на котором запущена служба iperf;
- SPEED - задает скорость передачи данных в службу iperf (по умочанию: 1 Mbit/s, только для udp теста);
- TIME - время одного цикла теста в секундах (по умолчанию: 10);
- NUM - число циклов теста (по умолчанию: 1);
- NT - число потоков (по умолчанию: 1);
- SOCKSZ - размер буфера сокета (по умолчанию: 8Кб);
- tos - тип обслуживания: задается либо байтом TOS, либо reliability - высокая надежность, throughput - высокая пропускная способность, delay - низкая задержка передачи 1P-сегмента;
- WIN - размер TCP-окна (по умолчанию: 16Кб, только для tcp-теста);
- MSS - максимальный размер сегмента TCP (только для tcp-теста);
- bidirectional - задает двунаправленный тест - передача данных в обе стороны последовательно, BPORT - локальный порт для двунаправленного теста.

40. Служба SLAGENT

Система Dionis-NX имеет службу slagent, реализующую возможность генерации эхо-пакетов полученных по протоколу UDP.

А также команды **udp-ping** и **udp6-ping** для генерации запросов, приема эхо-пакетов и проверки целостности и качества соединений по протоколу UDP.

40.1 Настройка службы slagent из режима configure

Чтобы войти в режим конфигурации службы, следует выполнить команду:

```
(config)# service slagent
```

Задайте пары адрес-порт, на которых система будет слушать UDP-трафик:

```
(config—service—slagent)#   listen   10.0.0.1  1000
(config—service—slagent)#   listen   10.0.0.1   1001
(config—service—slagent)# listen 10.0.0.2 1000
```

Для прослушивания UDP-трафика на IPv6 адресах используется команда **listen6:**

```
(config—service—slagent)# listen6 fe80: :be5f:f4ff:fec4:7edd 1005
```

Можно задать максимум 1000 пар адрес-порт.

Для включения журнала службы следует выполнить команду:

```
(config—service—slagent)# log
```

Для включения службы следует выполнить команду:

```
(config—service—slagent)# enable
```

Для выключения службы следует выполнить команду:

```
(config—service—slagent)# disable
```

40.2 Герерация UDP-запросов

Для генерации UDP-запросов на IPv4 адреса используется команда udp-ping из режима enable,

udp-ping <IP> <PORT> [size <SZ>] [indefinite | repeat <R>]

Для генерации UDP-запросов на IPv6 адреса используется команда udp-ping6 из режима enable.

udp-ping6 <IPv6> <PORT> [size <SZ>] [indefinite | repeat <R>]

Данные команды имеют по два обязательных параметра:

- IP или IPv6 - IPv4 адрес (для udp-ping) или IPv6 адрес (для udp-ping6) узла на котором запущена служба slagent;

- PORT - задает порт, на котором запущена служба slagent;

Остальные параметры являются необязательными:

- size <SZ> - размер пакета;
- repeat <R> - количество запросов (по умолчанию 4 запроса)
- indefinite - бесконечное число запросов (Для прерывания генерации запросов необходимо нажать на клавиатуре Ctrl-C.)

41. Служба LLDP

Dionis-NX имеет службу LLDP, реализующую протокол канального уровня, который позволяет сетевым устройствам анонсировать в сеть информацию о себе и о своих возможностях, а также собирать эту информацию о соседних устройствах.

LLDP-информация может получаться и отправляться (анонсироваться) только с/на непосредственно подключенные к данной системе устройства (в дальнейшем - соседи), либо подсоединенные через концентратор или повторитель. Анонсируемая данной системой информация, будучи полученной соседями, не пересылается далее по сети. Переданные LLDP анонсы не требуют своего подтверждения или какой-либо другой реакции от получателей данных анонсов.

Таким образом, LLDP-служба позволяет данной системе:

- передавать LLDP-информацию о себе соседям;
- получать LLDP-информацию от соседей;
- сохранять и управлять полученной LLDP-информацией в LLDP MIB, которую можно получить удаленно, если включена служба snmp на системе.

LLDP-служба передает анонсы в виде пакетов, называемых LLDP PDU, состоящих из набора TLV элементов, каждый из которых содержит информацию определенного типа об устройстве, либо о сетевом порте, который передает данный анонс.

Служба LLDP позволяет настраивать различные параметры для отдельных физических интерфейсов (в дальнейшем - портов), либо для всех вместе.

41.1 Базовые настройки службы и настройки обязательных TLV

Описываемые в данном разделе обязательные TLV соответсвуют стандарту Mandatory Base TLVs—IEEE 802.1AB-2005.

Команды данного раздела позволяют задать такой обязательный TLV как TTL. Другие обязательные TLV формируются и передаются службой автоматически (Port ID TLV и Chasis ID TLV).

Чтобы войти в режим настройки службы, следует выполнить команду:

(config)# service lldp

Чтобы указать порт, который служба будет использовать для приема и посылки LLDP-фрэймов, следует выполнить команду:

(config—service—lldp)# listen <IFACE>

По умолчанию: использовать для LLDP все порты.

Следующая команда задает длительность задержки между посылками LLDP-фрэймов:

(config—service—lldp)# tx—interval <NSEC>

По умолчанию: 30 сек.

Следующая команда определяет параметр для расчета TTL отправленного LLDP-пакета по следующей формуле:

TTL = <значение tx-multiplier> *<значение tx-interval>

(config—service—lldp)# tx—multiplier <NSEC>

По умолчанию: 4.

Значение TTL в LLDP-пакете определяет промежуток времени, в течение которого информация, полученная в данном LLDP-пакете, остается актуальной.

Указать поддерживаемые, помимо LLDP, протоколы:

(config—service—lldp)# proto < a 111 cd p | fd p | sd p | ed p >

Параметры команды:

- cdp - Cisco Discovery Protocol;
- fdp - Foundry Discovery Protocol;
- edp - Extreme Discovery Protocol;
- sdp - SynOptics Network Management Protocol;
- all - все перечисленные выше протоколы.

По умолчанию: не включать поддержку дополнительных протоколов.

Включить режим службы, при которой она только принимает LLDP-пакеты, однако не передает их, т.е. не анонсирует свою информацию:

(config—service—lldp)# read-only

По умолчанию: режим отключен.

Команда задает тип MAC-адреса отправителя в LLDP-фреймах, посылаемых на подчиненные bond-интерфейсы:

(config—service—lldp)# bond—slave—mac <real|zero|fixed|local>

Параметры:

- real - настоящий мак-адрес подчиненного интерфейса;
- zero - нулевой мак-адрес;
- fixed - фиксированный мак 00:60:08:69:97:ef;
- local - настоящий мак-адрес подчиненного интерфейса, с установленным специальным битом в мак-адресе; если данный бит уже взведен в настоящем мак-адресе – используется фиксированный мак-адрес 00:60:08:69:97:ef.

По умолчанию: bond-slave-mac local.

41.2 Настройка опциональных TLV (DOT1)

Описываемые в данном разделе опциональные TLV соответсвуют стандарту Optional Base TLVs—IEEE 802.1AB-2005.

Команды данного раздела позволяют задать следующие опциональные TLV:

- описание системы;
- описание порта;
- имя системы;
- IP-адрес управления.

Пример настройки базовой анонсируемой информации о системе:

```
(config—service—lldp)# system hostname hostl
(config—service—lldp)# system description "main core"
(config—service—lldp)# system iface—descr
```

Первая команда задает имя текущего хоста для аносирования вместо реального системного имени хоста. Вторая команда задает описание системы вместо стандартного описания, в которое входит: версия и название ядра системы, имя хоста, дата сборки и тип процессорной архитектуры системы. Последняя команда указывает, что вместо заданного в системе описания интерфейса нужно анонсировать имя соседнего узла или их число.

Указать управляющий IP-адрес системы следует при помощи команды:

```
(config—service—lldp)# management—address <IP>
```

Это адрес, который используется для получения LLDP-информации с текущей системы.

По умолчанию: использовать первый найденный IP-адрес текущей системы.

41.3 Настройка опциональных TLV (DOT3)

Описываемые в данном разделе опциональные TLV соответствуют стандарту IEEE 802.3 Organizationally Specific TLVs (802.3 TLVs)-IEEE 802.1AB-2005 Annex G.

Следующая команда задает TLV, описывающий POE-MDI-опции для порта - параметры передачи энергии через MDI-портов.

Анонсирование PoE(Power over Ethernet) - информации для MDI-портов:

```
(config—service—lldp)# dot3 power < *|IFACE > <TYPE> powerpairs <PWP> [supported]
      [enabled] [paircontrol] [CLASS] [802.3at type <ATYPE> source <SRC> prio <PRIO>
      <PWREQ> <PWALC>]
```

- *|IFACE> - задает все порты, либо конкретный порт системы;

Параметры:

- TYPE - задает энергетический тип порта: источник энергии (pse), либо получатель энергии (Pd);
- PWP - способ передаче энергии по витой паре:
  - spare - использовать 2 свободные витые пары кабеля Ethernet (те,что не используются для передачи данных);
  - signal - использовать витые пары кабеля, занятые в передаче данных;
- suppored - данный порт поддерживает передачу энергии через MDI;
- enabled - на данном порту включена передача энергии через MDI;
- paircontrol - контроль выбора витых пар для передачи энергии;
- CLASS - класс энергии (от 0 до 4);
- ATYPE - определяет типа стандарта 803.3at: type 1, либо type 2;
- source - определяет источник энергии для данного порта:
  - дляTYPE=pd:
    * unknown - неизвестный источник энергии;
    * pse - источник энергии;
    * local - источник энергии - локальный;
    * both - источник энергии - локальный pse;
  - для TYPE=pse:
    * unknown - неизвестный источник энергии;
    * primary - первичный источник энергии;
    * backup - резеврный источник энергии (например,UPS);
- PRIO - приоритет источника энергии (неизвестный (unknown), критичный(сгШса1), высо-кий(ЫдЬ) или ннЗкн^Iow));
- PWREQ - требуемая мощность в миливаттах;
- PWALC - выделяемая мощность в миливаттах.

Справочная таблица для CLASS (мощность указана в Ваттах):

Класс мощности	Мощность для PD	Мощность от PSE
0	0.44-12.95	15.4
1	0.44-3.84	4.0
2	3.84-6.49	7.0
3	6.49-12.95	15.4
4	12.95-25.5	30

Приоритет PRIO определяет насколько данный порт важен для обеспечения его энергией: в случае приоритета critical у порта - даже при недостатке мощности у PSE будет сделано все (отключены от питания другие, менее приоритетные порты) для обеспечения мощностью данного порта.

41.4 Настройка расширения LLDP-MED

LLDP-MED - это расширение LLDP-протокола для оконечных медиа-устройства (Media Endpoint Devices, MED), используемое для LLDP-взаимодействия между сетевыми устройствами LAN (маршрутизаторы, концентраторы и др.) и оконечными медиа-устройствами, подсоединенными к ним, например IP-телефонами.

Настройка класса устройств, для которых будет анонсироваться LLDP-MED информация, осуществляется с помощью команды:

(config—service—lldp)# med transmit <classl|class2|class3|class4>

Классы устройств означают следующее:

- classl - эти устройства поддерживают базовые возможности обнаружения по LLDP, анонсирование сетевых политик (VLAN ID, приоритеты 802.1p и DSCP), управление PoE; этот класс включает такие устройства как IP контроллеры вызовов и communication-related сервера;
- class2 - включает в себя classl плюс возможности передачи медиаинформации; это такие устройства, как голосовые/медиа-шлюзы, устройства для конференц-связи,медиа-сервера;
- class3 - включает в себя class2 плюс идентификацию местоположения, номер экстренной связи(ЕЕШ), поддержку коммутатора 2го уровня, управление информацией об устройстве; как правило, это IP-телефоны или софт-телефоны;
- class4 - прочие стевые устройства: хабы, мосты,сетевые карты,маршрутизаторы и другие устройства сетевого взаимодействия.

Анонсирование физического адреса местонахождения портов следует задавать при помощи команды:

(config—service—lldp)# med address < *|IFACE > <CC> city <CITY> street <STR> bid <BLD> app <APP>

Параметры:

- <*|IFACE> - задает все порты, либо конкретный порт системы;
- CC - код страны;
- CITY - город;
- STR-улица;
- BLD - номер строения;
- APP - номер квартиры/комнаты в указанном строении BLD.

Анонсирование географических координат местонахождения портов следует задавать при помощи команды:

(config—service—lldp)# med coordinate < *|IFACE > latitude <LAT N|S> longtitude <LNG W|E> altitude <ALT> <DATUM>

Параметры:

- LAT - широта (число с плавающей точкой), N - северная широта, S - южная широта;
- LNG - долгота (число с плавающей точкой), W - западная долгота, E - восточная долгота;
- ALT - высота над уровнем моря (метры, число с плавающей точкой);
- DATUM - датум (nad83, wgs84 или nad83-mllw).

Анонсирование номера экстренной связи следует задавать при помощи команды:

(config—service—lldp)# med elin < *|IFACE > <ELIN>

Параметры:

- <*|IFACE> - задает все порты, либо конкретный порт системы;
- ELIN - строка, задающая номер экстренной связи, например 112 (для Европы), 911(для США).

Большинство IP-телефонов (или других IP-устройств для передачи голоса) имеют два интерфейса: один для соединения с сетью и другой для соединения с другим устройством, например компьютером, в результате чего компьютер может подсоединиться к сети через IP-телефон. Желательно различать информационный и голосовой трафик так, чтобы разные параметры QoS применялись для каждого типа трафика.

Следующая команда позволяет передать MED-устройству значения параметров VLAN ID и DSCP для выбранного типа трафика. В результате MED-устройство (например, IP-телефон) будет передавать голосовой трафик исходя из сконфигурированных значений, а обычный информационный трафик, идущий с данной системы через MED-устройство, будет передаваться, исходя из значений по умолчанию.

Анонсирование сетевой политики для портов следует задавать при помощи команды:

(config—service—lldp)# med policy < *|IFACE > <ATYPE> [unknown] [vlan <VID>] [dscp <DSCP>] prio <PRIO>

Параметры:

- <*|IFACE> - задает все порты, либо конкретный порт системы;
- ATYPE - задает тип приложения:
  - voice - передача голоса/телефония;
  - voice-signaling - голосовая сигнализация;
  - guest-voice;
  - guest-voice-signaling;
  - softphone-voice - программные телефоны;
  - video-conferencing - видео-конференция;
  - streaming-video - потоковое видео;
  - video-signaling - видео-сигнализация;
- VID - анонсируемое VLAN ID, используемое для сетевой политики указанного типа приложения;
- DSCP - анонсируемое значение DSCP (от 0 до 63), используемое для сетевой политики указанного типа приложения;

- PRIO - значение CoS, используемое для указанного типа приложения;
- unknown - сетевая политика неизвестна для данного типа приложения.

С помощью следующей команды можно настроить параметры PoE. PoE - это механизм для передачи мощности сетевому устройству по тому же кабелю, который используется устройством для передачи сетевых данных. Понятия используемые в данной команде:

- Powered Device (PD) - устройства потребляющие энергию;
- Power Sourcing Equipment (PSE) - устройства, являющиеся источниками энергии.

Анонсирование PoE(Power over Ethernet)-параметров следует задавать при помощи команды: (config—service—lldp)# med power < *|IFACE > <TYPE> source <SRC> prio <PRIO> <PWREQ>

Параметры:

- <*|IFACE> - задает все порты, либо конкретный порт системы;
- TYPE - задает энергетический тип порта: источник энергии (pse), либо получатель энергии (Pd);
- source - определяет источник энергии для данного порта:
  - дляTYPE=pd:
    - unknown - неизвестный источник энергии;
    - pse - источник энергии - pse;
    - local - источник энергии - локальный;
    - both - источник энергии - локальный pse;
  - для TYPE=pse:
    - unknown - неизвестный источник энергии;
    - primary - первичный источник энергии;
    - backup - резеврный источник энергии (например,UPS);
- PRIO - приоритет источника энергии (неизвестный (unknown), критичный(critical), высокий(high) или низкий(low));
- PWREQ - значение мощности в миливаттах, требуемое (для pd), либо доступное (для pse).

Приоритет PRIO определяет, насколько данный порт важен для обеспечения его энергией: в случае приоритета critical у порта - даже при недостатке энергии у PSE будет сделано все (отключены от питания другие, менее приоритетные порты) для обеспечения данного порта энергией.

Пример PD-устройств: WAP, VoIP-устройства, IP-камеры.

Пример PSE-устройств: коммутатор.

41.5 Работа со службой

Следующие команды задают передачу различной информации о службе. Введем ряд параметров, которые могут быть показаны администратору в результатах выдачи команд данного раздела:

- Chassis ID - идентификатор платформы, например, это может быть MAC-адрес;
- Port ID - порт, который послал LLDP PDU.

Для запуска службы следует выполнить команду:

(config—service—lldp)# enable

Для остановки службы следует выполнить команду:

(config—service—lldp)# disable

Для просмотра журналов службы следует выполнить команду:

(config—service—lldp)# do show service lldp log

Для просмотра статуса работы службы следует выполнить команду:

(config—service—lldp)# do show service lldp status

Для просмотра статистики работы службы следует выполнить команду:

(config—service—lldp)# do show service lldp statistics [summary] [IFACE]

Если указан порт - будет показана информацию по нему. Если указан параметр summary - будет показана краткая статистическая сводка по работе службы.

Для просмотра информации по физическим сетевым портам следует выполнить команду:

(config—service—lldp)# do show service lldp neighbors [verbose|summary] [hidden] [IFACE]

Если указан порт - будет показана информацию по нему. Если указан параметр verbose - будет показана более подробная информация. Если указан параметр hidden - будет показана информация об удаленных интерфейсах. Если указан параметр summary - будет показана краткая статистическая сводка.

42. MAILER - служба пересылки почтовых сообщений

42.1 Введение

Dionis-NX имеет возможность отправки сообщений на указанный e-mail с помощью службы пересылки почтовых сообщений - MAILER

42.2 Настройка службы пересылки почтовых сообщений

Для настройки службы пересылки почтовых сообщений используется команда: service mailer из режима configure

adm@DionisNX(config)# service mailer adm@DionisNX(config—service—mailer)#

Настройка службы происходит в два этапа:

1. Настройка учетных записей службы;
2. Прочая настройка.

Настройка учетных записей службы

Учетная запись представляет собой настройки профиля для подключения к smtp-серверу

Для создания новой учетной записи или для редактирования существующей учетной записи необходимо выполнить команду account <Имя>

adm@DionisNX(config—service—mailer)# account gmail
adm@DionisNX(config—service—mailer—gmail)#

Команды доступные для настройки учетной записи.

команда	параметр
auth <on/off>	Включить или отключить аутентификацию
user <username>	Имя пользователя для аутентификации
password <password >	Пароль для аутентификации
tls <on/off>	Включить или отключить TLS/SSL
tls_certcheck <on/off>	Включить или отключить проверку TLS/SSL сертификатов
from <envelope-from>	Адрес отправителя
host <hostname>	SMTP сервер
port <port>	Порт SMTP сервера
timeout <Num>	Тайм-аут подключения в секундах (необязательный параметр)

Прочая настройка

Для завершения настройки службы необходимо выбрать учетную запись по умолчанию из списка уже настроенных учетных записей. Для этого необходимо выполнить команду default-acc <account-name>

adm@DionisNX(config—service—mailer)# default—acc gmail

42.2.1 Примеры настроек учетных записей для различных smtp-серверов

gmail

adm@DionisNX(config—service—mailer)# account gmail
adm@DionisNX(config—service—mailer—gmail)# host smtp.gmail.com
adm@DionisNX(config—service—mailer—gmail)# user username@gmail.com
adm@DionisNX(config—service—mailer—gmail)# password Secret
adm@DionisNX(config—service—mailer—gmail)# port 587
adm@DionisNX(config—service—mailer—gmail)# auth on
adm@DionisNX(config—service—mailer—gmail)# tls on
adm@DionisNX(config—service—mailer—gmail)# tls_certcheck off
adm@DionisNX(config—service—mailer—gmail)# from username@gmail.com

mail

adm@DionisNX(config—service—mailer)# account mail
adm@DionisNX(config—service—mailer— mail)# host smtp.mail.ru
adm@DionisNX(config—service—mailer— mail)# user username@mail.ru
adm@DionisNX(config—service—mailer— mail)# password Secret
adm@DionisNX(config—service—mailer— mail)# port 25
adm@DionisNX(config—service—mailer— mail)# auth on
adm@DionisNX(config—service—mailer— mail)# tls on
adm@DionisNX(config—service—mailer— mail)# tls_certcheck off
adm@DionisNX(config—service—mailer— mail)# from username@mail.ru

42.3 Отправка сообщения или файла с помощью службы MAILER

Для отправки сообщения на e-mail-адрес в enable-режиме необходимо выполнить следующую команду:

adm@DionisNX# mailertest@mail.com [message <message>]

Все параметры команды mailer:

команда	параметр
via <account>	Имя настроенной учетной записи. (Необязательный параметр. Если не указан, произойдет отправка с учетной записи, установленной по умолчанию)
<addr1, addr2...>	Список E-mail адресов для рассылки (Обязательный параметр)
message <Text>	Текст сообщения (Необязательный параметр)
subject <Text>	Тема сообщения (Необязательный параметр)
attach <file|running-config|startup-config|default-config>	Вложение (Необязательный параметр)

42.4 Настройка service-watcher для отправки сообщений с помощью службы MAILER

Dionis-NX имеет возможность отслеживания появления заданных сообщений в журнальных файлах системы и дальнейшую отправку сообщений на указанный e-mail.

Для этого в настройке службы service-watcher необходимо указать параметры для службы пересылки почтовых сообщений.

Ниже приведен пример для поиска и отправки новых сообщений, содержащих строку usb в kernel.log на адрес test@mail.com adm@DionisNX(config—service—

watcher— kernel)# mailer usb test@mail.com

Все параметры команды mailer для настройки service-watcher:

команда	параметр
<rx>	Регулярное выражения для поиска. (Обязательный параметр)
via <account>	Имя настроенной учетной записи службы mailer. (Необязательный параметр. Если не указан, произойдет отправка с учетной записи, установленной по умолчанию)
<addrl,addr2...>	Список e-mail-адресов для рассылки (Обязательный параметр)
subject <Text>	Тема сообщения (Необязательный параметр)

43. Служба автоконфигурирования IPv6

Возможности по автоматическому конфигурированию IPv6 достаточно широки. Для реализации этих возможностей в локальной сети должен присутствовать маршрутизатор, который отвечает на запросы автоконфигурации от хостов локальной сети. Такой маршрутизатор реализует протокол NDP (Neighbor Discovery Protocol). Метод автоконфигурирования называется SLAAC (StateLess Address AutoConfiguration).

Специальный RA-сервис (Router Advertisement service), работающий на маршрутизаторе, ожидает сетевых запросов RS (Router Solicitation) от хостов и отвечает на них сообщениями RA (Router Advertisement). Также в сеть периодически рассылаются сообщения RA без получения запроса. Сообщения RA содержат информацию, которую хосты используют для конфигурирования своих интерфейсов. В частности, передается префикс сетевого адреса, MTU, информация о маршрутизаторе по-умолчанию.

43.1 Секция настройки сервиса RA

Для настройки сервиса RA в ОС Dionis-NX предусмотрена отдельная секция в режиме конфигурирования.

```
adm@DionisNX(config)# service ra
```

Для включения сервиса RA используется команда:

```
adm@DionisNX(cfg—srv— ra)# enable
```

Для отключения сервиса RA используется команда:

```
adm@DionisNX(cfg—srv— ra)# disable
```

Для применения измененных настроек предусмотрена команда:

```
adm@DionisNX(cfg—srv— ra)# reload
```

Любые настройки сервиса RA связаны с конкретным сетевым интерфейсом. На маршрутизаторе может присутствовать сразу несколько интерфейсов и к каждому из них может быть подключена отдельная локальная IPv6 сеть. Из каждой локальной IPv6 сети могут приходить запросы RS на свой интерфейс. Т.е. настраивая параметры конкретного интерфейса на маршрутизаторе, изменяются настройки для подключенной к этому интерфейсу локальной сети.

Чтобы зайти в секцию настроек для конкретного интерфейса используется команда:

```
adm@DionisNX(cfg—srv— ra)# iface ethernet 0
```

Все настройки о которых пойдет речь в дальнейшем вводятся внутри секции, относящейся к конкретному интерфейсу.

43.2 Общие настройки интерфейса

Чтобы сервис периодически рассылал сообщения RA, а также отвечал на запросы RS на указанном интерфейсе, необходимо включить его в активное состояние.

adm@DionisNX(cfg—srv—ra—ethernet0)# adv send on

Обратная команда: adm@DionisNX(cfg—srv—

ra—ethernet0)# adv send off

Без флага "adv send" секция настройки интерфейса может хранить установленные значения, но функционирование сервиса RA на данном интерфейсе будет остановлено.

Чтобы сервис самостоятельно не рассылал групповые (multicast) сообщения RA, а только отвечал на запросы RS от конкретных хостов, используется специальная опция.

adm@DionisNX(cfg—srv—ra—ethernet0)# unicast—only

Сообщения RA будет отправлено адресно, тому хосту, который прислал запрос RS.

Максимальный интервал времени между отправкой RA сообщения по инициативе маршрутизатора задается опцией "max-adv-interval" и измеряется в секундах. Может изменяться в пределах от 4 сек до 1800 сек. Значение по-умолчанию 600 сек.

adm@DionisNX(cfg—srv—ra—ethernet0)# max—adv—interval 600

Минимальный интервал времени между отправкой RA сообщения по инициативе маршрутизатора задается опцией "min-adv-interval" и измеряется в секундах. Может изменяться в пределах от 3 сек до 75% от значения "max-adv-interval". Значение по-умолчанию 33% от "max-adv-interval".

adm@DionisNX(cfg—srv—ra—ethernet0)# min—adv—interval 500

Минимальный интервал времени между отправкой группового RA сообщения в ответ на запрос задается опцией "min-sol-delay" и измеряется в секундах. Значение по-умолчанию 3 сек.

adm@DionisNX(cfg—srv—ra—ethernet0)# min—sol—delay 3

Флаг "adv managed-flag" указывает хостам использовать ли управляемый (stateful) протокол, например DHCPv6, для автоконфигурирования адреса в дополнение к автоконфигурированию адресов с помощью сервиса RA (stateless).

adm@DionisNX(cfg—srv—ra—ethernet0)# adv managed—flag on

Флаг "adv other-flag" указывает хостам использовать ли управляемый (stateful) протокол, например DHCPv6, для автоконфигурирования не-адресной информации.

adm@DionisNX(cfg—srv—ra—ethernet0)# adv other—flag on

Когда MTU (Maximum Transfer Unit) в сети не зафиксирован жестко, используется опция "adv mtu", чтобы гарантировать единое значение MTU для всех хостов в сети. Может изменяться в пределах от 1280 до максимально допустимого значения MTU для данного типа соединения.

adm@DionisNX(cfg—srv—ra—ethernet0)# adv mtu 1500

Опция "adv reachable-time" определяет время, в миллисекундах, в течении которого хост считает соседний хост доступным после получения подтверждения доступности. Используется алгоритмом выявления недоступности соседнего хоста (Neighbor Unreachability Detection). Значение не должно превышать 3600000 миллисекунд (1 час).

adm@DionisNX(cfg—srv— ra—ethernet0)# adv reachable—time 2000000

Опция "adv retrans-timer" задает время, в миллисекундах, между посылкой запросов соседних хостов (Neighbor Solicitation). Используется для разрешения адресов и для алгоритма выявления недоступности соседнего хоста (Neighbor Unreachability Detection).

adm@DionisNX(cfg—srv—ra—ethernet0)# adv retrans-timer 20000

Опция "adv hop-limit" позволяет задать начальное значения поля "Hop Limit" (TTL в IPv4) в заголовке исходящих IPv6 пакетов. Значение по-умолчанию 64.

adm@DionisNX(cfg—srv— ra—ethernet0)# adv hop—limit 64

Опция "adv default-lifetime" определяет время в течении которого маршрутизатор считается маршрутизатором по-умолчанию. Определяется в секундах. Значение "0" означает, что маршрутизатор не является маршрутизатором по-умолчанию и не должен появляться в списке маршрутизаторов по-умолчанию. Значение поля может изменяться от значения "max-adv-interval" до 9000 секунд. Значение по-умолчанию - утроенное значение "max-adv-interval".

adm@DionisNX(cfg—srv— ra—ethernet0)# adv default—lifetime 8000

Опция "adv default-pref" определяет насколько маршрутизатор по-умолчанию предпочтителен. Может иметь значения "low", "medium", "high". По-умолчанию используется "medium", т.е. среднее значение.

adm@DionisNX(cfg—srv— ra—ethernet0)# adv default-pref low

Опция "adv ll-addr" определяет будет ли адрес канального уровня (link-layer) исходящего интерфейса включен в сообщение RA.

adm@DionisNX(cfg—srv— ra—ethernet0)# adv ll—addron

43.3 Префикс

Префикс, полученный от маршрутизатора в сообщении RA, хост использует для формирования глобального адреса своего интерфейса. Стандарт описывает, что заданный на маршрутизаторе префикс должен иметь маску "/64", т.е. 64 значимых бит (из 128 в полном IPv6 адресе). Хост формирует IPv6 адрес, используя префикс в качестве старшей части и MAC-адрес интерфейса (в специальной форме EUI-64) в качестве младшей части.

Префикс задается следующей командой:

adm@DionisNX(cfg—srv— ra—ethernet0)# prefix 2001:db8:0:1: :/64

При вводе такой команды администратор входит в секцию, описывающую настройки для данного префикса. Все далее описанные в разделе команды относятся к этой секции.

Флаг "adv autonomous" определяет может ли данный префикс использоваться для автономного конфигурирования адреса согласно RFC 4862. Значение по-умолчанию - "on".

adm@DionisNX(cfg—srv—ra—ethernetO—2001:db8:0:1: :/64)# adv autonomous on

Флаг "adv on-link" определяет может ли префикс быть признаком принадлежности хостов с таким префиксом к одной локальной сети. В общем случае в сетях IPv6 одинаковый префикс в IP-адресах хостов не означает, что хост будет доступен напрямую. Иногда хосты с одинаковым префиксом вынуждены использовать маршрутизатор для обмена трафиком. Значение по-умолчанию
- "on".

adm@DionisNX(cfg—srv—ra—ethernet0—2001:db8:0:1: :/64)# adv on —link on

Опция "adv valid-lifetime" определяет время (в секундах), в течении которого префикс считается актуальным для определения локальности сети (on-link). Т.е. в течении этого времени единый префикс может считаться признаком того, что хост с таким же префиксом будет доступен по прямому подключению. Может изменяться в пределах от 2 часов до бесконечности (специальное значение "infinity"). Значение по-умолчанию - 86400 сек (1 день).

adm@DionisNX(cfg—srv—ra—ethernet0—2001:db8:0:1: :/64)# adv valid —lifetime infinity

Опция "adv pref-lifetime" определяет время (в секундах) в течении которого адреса, сформированные на основания префикса, считаются предпочтительными. Специальное значение "infinity"
- означает бесконечное время. Значение по-умолчанию 14400 секунд (4 часа).

adm@DionisNX(cfg—srv—ra—ethernet0—2001:db8:0:1: :/64)# adv pref— lifetime 14400

Флаг"deprecate-prefix" указывает, что, при выключении маршрутизатора, сервис объявит префикс недействительным. Эта опция используется только в тех случаях, когда только один маршрутизатор экспортирует такой префикс. Иначе хосты будут считать адреса, построенные на основании этого префикса недействительными, хотя существуют другие маршрутизаторы экспортирующий данный префикс.

adm@DionisNX(cfg—srv—ra—ethernet0—2001:db8:0:1: :/64)# deprecate-prefix on

43.4 Маршруты

Сервис RA на конкретном интерфейсе может экспортировать маршруты для передачи хостам. Маршрут задается следующей командой.

adm@DionisNX(cfg—srv—ra—ethernet0)# route 2002:db8:0:2: :/64

После подобной команды администратор попадает в секцию конфигурирования маршрута.

Опция "adv route-lifetime" определяет время (в секундах), в течении которого маршрут считается актуальным. Если маршрут должен быть актуальным всегда, используется специальное значение "infinity". Значение по-умолчанию - утроенное значение "max-adv-interval".

adm@DionisNX(cfg—srv—ra—ethernet0—2002:db8:0:2: :/64)# adv route—lifetime infinity

Опция "adv route-pref" определяет степень предпочтения. Может иметь значения "low", "medium", "high". По-умолчанию используется "medium", т.е. среднее значение.

adm@DionisNX(cfg—srv—ra—ethernet0—2002:db8:0:2: :/64)# adv route-pref low

Флаг "remove-route" указывает, что, при выключении маршрутизатора, сервис объявит маршрут недействительным. Значение по умолчанию - "on" (включено).

```
adm@DionisNX(cfg—srv— ra—ethernet0—2002:db8:0:2: :/64)# remove—route on
```

43.5 Клиентские хосты

По-умолчанию сервис RA рассылает сообщения RA на групповой (multicast) адрес, поэтому любой клиентский хост в сети получит это сообщение. Однако список клиентских хостов, которым будет рассылаться сообщения RA и на RS-запросы которых сервис будет отвечать, можно задать явно. В этом случае сообщения RA будут отправляться на индивидуальные (unicast) адреса хостов, а RS-запросы от других хостов будут игнорироваться.

```
adm@DionisNX(cfg—srv— ra—ethernet0)# client 2346: :1
adm@DionisNX(cfg—srv—ra—ethernet0)# client 2346: :2
```

В данном случае разрешена работа с двумя клиентскими хостами.

43.6 Рекурсивный DNS-сервер

Для автоконфигурирования списка DNS-серверов на клиентских хостах существует специальная опция сообщения RA.

```
adm@DionisNX(cfg—srv— ra—ethernet0)# rdnss
```

Внутри секции "rdnss" можно указать список DNS-серверов.

```
adm@DionisNX(cfg—srv— ra—ethernet0—rdnss)# server 2346: :1
adm@DionisNX(cfg—srv— ra—ethernet0—rdnss)# server 2347::2
```

Опция "adv rdnss-lifetime" определяет время (в секундах), в течении которого список серверов считается актуальным. Если список должен быть актуальным всегда, используется специальное значение "infinity". Специальное значение "0" означает, что список неактуален. Ненулевое значение опции не может быть меньше чем "max-adv-interval". Значение по-умолчанию - удвоенное значение "max-adv-interval".

```
adm@DionisNX(cfg—srv— ra—ethernet0—rdnss)# adv rdnss-lifetime infinity
```

Флаг "flush-rdnss" указывает, что, при выключении маршрутизатора, сервис объявит список DNS-серверов недействительным. Значение по умолчанию - "on" (включено).

```
adm@DionisNX(cfg—srv— ra—ethernet0—rdnss)# flush-rdnss
```

43.7 Список поиска DNS

Для автоконфигурирования списка поиска DNS (DNS search list) на клиентских хостах существует специальная опция сообщения RA.

adm@DionisNX(cfg—srv—ra—ethernet0)# dnssl

Внутри секции "dnssl" можно указать список суффиксов. adm@DionisNX(cfg—srv—ra—ethernet0—dnssl)# suffix my—company.ru adm@DionisNX(cfg—srv—ra—ethernet0—dnssl)# suffix other.com

Опция "adv dnssl-lifetime" определяет время (в секундах), в течении которого список суффиксов считается актуальным. Если список должен быть актуальным всегда, используется специальное значение "infinity". Специальное значение "0" означает, что список неактуален. Ненулевое значение опции не может быть меньше чем "max-adv-interval". Значение по-умолчанию - удвоенное значение "max-adv-interval".

adm@DionisNX(cfg—srv—ra—ethernet0—dnssl)# adv dnssl —lifetime infinity

Флаг "flush-dnssl" указывает, что, при выключении маршрутизатора, сервис объявит список DNS суффиксов недействительным. Значение по умолчанию - "on" (включено).

adm@DionisNX(cfg—srv—ra—ethernet0—dnssl)# flush—dnssl

44. L2TP-туннели

44.1 Введение

Dionis-NX имеет поддержку протокола L2TPv3. L2TP (англ. Layer 2 Tunnelling Protocol) — сетевой протокол туннелирования канального уровня, сочетающий в себе протокол L2F (Layer 2 Forwarding), разработанный компанией Cisco, и протокол PPTP корпорации Microsoft.

Протокол L2TP позволяет передавать пакеты PPP через TCP/IP-сеть посредством инкапсуляции PPP в L2TP в UDP.

Основные сущности протокола:

- LAC (концентратор доступа L2TP) - настраивается командой lac в интерфейсе l2tp;
- LNS (сетевой сервер L2TP) - настраивается командой lns в службе l2tp
- Виртуальный IP-адрес - это IP-адрес конца туннеля, назначаемый при его создании.

Допустим, удаленная система хочет соединиться с удаленной LAN по L2TP туннелю. Существует две схемы взаимодействия удаленной системы, LAC и LNS:

1. Удаленная система инициирует PPP-соединение с LAC через коммутируемую телефонную сеть PSTN. LAC затем прокладывает туннель для PPP-соединения через Интернет или другие сети к LNS, и таким образом осуществляется доступ к удаленной LAN.
2. Удаленная система может сама являться LAC-клиентом и участвовать в туннелировании до исходной LAN без использования отдельного LAC, если ЭВМ, содержащая программу LAC-клиента, уже имеет соединение с Интернет. Создается "виртуальное" PPP-соединение и LAC-клиент формирует туннель до LNS.

В Dionis-NX используется 2-я схема: lac-клиент в виде interface l2tp и lns в виде службы l2tp.

Сообщения протокола:

- управляющие: создание, поддержка и удаление туннеля; состоят из множества AVP(пара «атрибут-значение»)-заголовков; в Dionis-NX обрабатываются в режиме пользователя;
- информационные сообщения: передача данных по туннелю; в Dionis-NX обрабатываются в режиме пользователя или в режиме ядра (по умолчанию), зависит от настройки.

Настройка L2TP в системе Dionis-NX сводится к настройке серверной части (служба l2tp) и клиентской части (интерфейс l2tp или LAC).

Для клиентских и серверных интерфейсов l2tp возможно применение правил NAT, ACL и прочих настроек интерфейсов.

44.2 Настройка службы l2tp

Данная служба представляет собой пул динамических интерфейсов, создаваемых в системе по мере подключения клиентов L2TP.

Формат реально создаваемого динамического интерфейса следующий: l2tp@<N>

Рассмотрим данный формат:

- l2tp@ - это тип интерфейса, знак @ говорит о том, что это серверный интерфейс.
- N - это номер серверного интерфейса

Для настройки службы l2tp выполните:

```
(config)# service l2tp
```

В службе далее необходимо создать 1 или больше Ins, которая описывает серверный пул динамических интерфейсов, создаваемых в системе по мере подключения клиентов (LAC):

```
(config—service—l2tp)# Ins srv
```

Клиент lac попадет в ту Ins, в которой разрешенный интервал IP-адресов клиентов будет содержать IP-адрес клиента. Настройка интервала разрешенных IP-адресов LAC будет рассмотрена далее разделе "Настройка LNS".

В службе может быть несколько LNS, каждая со своими L2TP- и PPP-параметрами.

44.2.1 Глобальные настройки службы l2tp

44.2.1.1 debug

Эта команда включает вывод в лог службы более подробную информации о ее функционировании по части L2TP протокола.

44.2.1.2 listen <IP>

Эта команда задает IP-адрес для принятия L2TP-запросов.

По умолчанию: 0.0.0.0

44.2.1.3 port <PORT>

Эта команда задает порт для принятия L2TP-запросов.

По умолчанию: 1701

44.2.1.4 userspace

Эта команда определяет, что обработка информационных сообщений L2TP будет производится в режиме пользователя.

По умолчанию: в режиме ядра.

44.2.1.5 lns <1ЧАМЕ>

Эта команда создает LNS с именем NAME.

44.2.2 Общие настройки для LNS и LAC (interface l2tp)

44.2.2.1 challenge-auth

Эта команда включает использование Challenge AVP протокола L2TP для взаимной аутентификации концов туннеля посредством общего секрета (см. далее «L2TP аутентификация»).

По умолчанию: выключено.

44.2.2.2 hidden-bit

Эта команда включает скрытие поля данных в AVP, содержащих важную информацию управляющих сообщений, такую как пароль пользователя или его ID.

По умолчанию: выключено.

44.2.2.3 length-bit

Эта команда включает использование поля длины сообщения в заголовке пакета L2TP.

По умолчанию: выключено.

44.2.2.4 txspeed <VAL>

Эта команда устанавливает скорость отправления данных через туннель в значение VAL бит/сек.

По умолчанию: ЮМбит/сек.

44.2.2.5 rxspeed <VAL>

Эта команда устанавливает скорость приема данных через туннель в значение VAL бит/сек.

По умолчанию: 10Мбит/сек.

44.2.2.6 rws <VAL>

Эта команда устанавливает максимальное число входящих неподтвержденных пакетов контрольного канала.

По умолчанию: 4.

44.2.3 Настройка LNS

Далее перечислены команды, специфичные для LNS.

44.2.3.1 localip <IP>

Команда устанавливает виртуальный IP-адрес концу туннеля на стороне LNS.

44.2.3.2 permit lac-range <IP_START> [IP_END]

Эта команда задает IP-адрес или интервал IP-адресов, от которых данному LNS разрешено принимать запросы на создание туннеля.

Если не указано ни одного интервала и нет ни одной LNS с интервалами lac-range, то попытка соединения разрешена всем LAC.

44.2.3.3 deny lac-range <IP_START> [IP_END]

Эта команда задает IP-адрес или интервал IP-адресов, от которых данному LNS не разрешено принимать запросы на создание туннеля.

44.2.3.4 permit ip-range <IP_START> [IP_END]

Эта команда задает интервал виртуальных IP-адресов, которые разрешено назначать удаленному концу туннеля (L2TP клиенту).

44.2.3.5 deny ip-range <IP_START> [IP_END]

Эта команда задает интервал виртуальных IP-адресов, которые не разрешено назначать удаленному концу туннеля (L2TP клиенту).

44.2.3.6 template <*|USER> {#l2tp-templ>

Команда задает привязку шаблона интерфейса с номером NUM к имени аутнтифицированного пользователя, для которого создается туннель, либо к любому пользователю, если имя пользователя задано символом в том числе к анонимному.

Команда позволяет применить различные настройки к вновь создаваемому интерфейсу, например привязать NAT или ACL, отключить multicast и другие настройки, доступные по команде ip.

Для работы данной команды необходимо создать шаблон интерфейса:

44.2.3.7 (config)# interface template-l2tp O

В данном шаблоне можно описать различные настройки, которые будут применены к реально создаваемому серверному интерфейсу типа l2tp@.

44.2.4 Настройка PPP

Настройки опций протокола PPP задаются для interface l2tp и LNS службы l2tp, а также для других протоколов и служб динамических туннелей, например, PPTP (см. «PPTP-туннели»).

Все команды для задания PPP-опций, кроме команд для задания PPP-аутентификации, начинаются со слова ppp.

Некоторые ppp-команды возможно задать только для служб протоколов, а другие имеют смысл только для клиентских интерфейсов.

44.2.4.1 ppp debug

Эта команда включает вывод в лог службы более подробной информации о ее функционировании по части PPP протокола.

44.2.4.2 ppp compression bsd [LEV]

Эта команда включает степень сжатия данных по BSD-compresion алгоритму.

Если значение LEV установлено в 0, то сжатие не будет применяться.

По умолчанию: 0.

44.2.4.3 ppp compression deflate [LEV]

Эта команда включает степень сжатия данных по Deflate-алгоритму.

Если значение LEV установлено в 0, то сжатие не будет применяться.

По умолчанию: 0.

44.2.4.4 ppp idle <N>

Эта команда задает максимальный период (в сек.) бездействия туннеля. По истечении данного периода времени туннель будет закрыт. Команда ppp redial-interval не относится к случаю закрытия по бездействию.

Доступно только для интерфейсов l2tps, pptps, pppoes.

По умолчанию: не проверять период бездействия туннеля.

44.2.4.5 ppp redial-interval <N>

Эта команда задает время ожидания после прерывания связи (отключения туннеля) перед повторной попыткой соединения с сервером. Только для клиентских интерфейсов, за исключением l2tp интерфейса, для него существует аналог этой команды - redial-interval.

По умолчанию: 30

44.2.4.6 ppp maxconnect <N>

Эта команда задает максимальный период (в сек.) активности туннеля (передачи по нему пакетов данных). По истечении данного периода времени туннель будет закрыт.

Доступно только для интерфейсов l2tps, pptps, pppoes.

По умолчанию: не проверять период работы туннеля.

44.2.4.7 ppp lcp-echo-failure <N>

Эта команда задает максимальное число LCP Echo-запросов без ответа. При превышении этого предела туннель будет закрыт.

По умолчанию: 5.

44.2.4.8 ppp lcp-echo-interval <N>

Эта команда задает интервал времени в секундах между LCP Echo-запросами.

По умолчанию: 3.

44.2.4.9 ppp mppe [stateless]

Эта команда включает использование MPPE-протокола - протокол шифрования данных, используемый поверх PPP.

Можно дополнительно включить режим stateless - без состояния.

Для работы команды ppp mppe stateless необходимо задать данную опцию как на серверном, так и на клиентском интерфейсе.

Для работы команды ppp mppe достаточно задать данную опцию только на серверном интерфейсе.

Данная опция отключает использование любых протоколов компрессии, заданных командой ppp compression.

По умолчанию: отключено.

44.2.4.10 ppp mru <VAL>

Эта команда устанавливает размер MRU (Maximum receive unit) в байтах.

По умолчанию: 1500.

44.2.4.11 ppp mtu <VAL>

Эта команда устанавливает размер MTU (Maximum transmit unit) в байтах.

По умолчанию: 1500.

44.2.4.12 ppp dns <IP> [IP2]

Эта команда предписывает передавать указанные адреса DNS-серверов.

Доступно только для интерфейсов l2tps, pptps, pppoes.

44.2.4.13 ppp getdns

Эта команда предписывает запрашивать адреса DNS-серверов у провайдера (серверного интерфейса).

Особенности взаимодействия с DHCP и ip resolver nameserver см. в п. 5.6.1 .

44.2.4.14 ppp proxyarp

Эта команда предписывает добавлять запись об IP- и MAC-адресах удаленного узла в ARP-таблицу. При этом будет полезным, если VPN-сеть разделяет адресное пространство с реальной сетью LAN.

По умолчанию: не добавлять записи.

44.2.4.15 ppp pap require

Эта команда устанавливает обязательную аутентификацию удаленного узла по PAP-протоколу.

При использовании данной команды необходимо задать секреты PAP командой pap.

По умолчанию: не требуется аутентификация узла по PAP-протоколу.

44.2.4.16 ppp chap require

Эта команда устанавливает обязательную аутентификацию удаленного узла по CHAP-протоколу.

При использовании данной команды необходимо задать секреты CHAP командой chap.

По умолчанию: не требуется аутентификация узла по CHAP-протоколу.

44.2.4.17 ppp ms-chap-v2 require

Эта команда устанавливает обязательную аутентификацию удаленного узла по MS-CHAPv2-протоколу.

при использовании данной команды необходимо задать секреты MS-CHAPv2 командой chap.

По умолчанию: не требуется аутентификация узла по M5-CHAPy2-протоколу.

44.2.4.18 ppp pap refuse

Эта команда предписывает отказать удаленному узлу в аутентификации себя по PAP-протоколу.

По умолчанию: не отказывать удаленному узлу в аутентификации себя по PAP-протоколу.

44.2.4.19 ppp chap refuse

Эта команда предписывает отказать удаленному узлу в аутентификации себя по CHAP-протоколу.

По умолчанию: не отказывать удаленному узлу в аутентификации себя по CHAP-протоколу.

44.2.4.20 ppp ms-chap-v2 refuse

Эта команда предписывает отказать удаленному узлу в аутентификации себя по MS-CHAPv2-протоколу.

По умолчанию: не отказывать удаленному узлу в аутентификации себя по MS-CHAPv2-протоколу.

44.2.5 L2TP аутентификация

Для настройки взаимной аутентификации LNS и LAC необходимо задать в их настройках опцию challenge-auth (см. «Общие настройки для LNS и 1_AC» выше в данной главе), которая предписывает узлу добавлять в управляющее сообщение L2TP заголовок Challange AVP.

Далее в глобальных настройках интерфейса l2tp и в службе l2tp необходимо задать общий секрет следующей командой:

44.2.5.1 auth <LOCAL|*> <REMOTE|*> <SECRET>

Эта команда задает общий секрет SECRET для локального узла с именем LOCAL и удаленного узла с именем REMOTE.

Знак «*» - означает любой узел.

Если LAC и LNS создаются в системе Dionis-NX, то:

- для службы l2tp: LOCAL - это имя LNS; REMOTE - это имя LAC
- для интерфейса l2tp: LOCAL - это имя LAC; REMOTE - это имя LNS

Если LAC и/или LNS создаются не в системе Dionis-NX, то передаваемые имена узлов LOCAL и REMOTE могут быть, например, доменными именами соответствующих узлов.

44.2.6 PPP аутентификация

Аутентификация уровня PPP имеет своими целями:

- задать пароли доступа клиентов к серверу L2TP, т.е. доступ LAC к LNS;
- для клиентов, прошедших аутентификацию, возможно задать IP-адрес, который будет присвоен данному клиенту при создании туннеля с ним.

Для задания паролей для алгоритмов аутентификации CHAP/MS-CHAP-V2 следует использовать команду chap, а для задания паролей для PAP следует использовать команду pap:

44.2.6.1 <chap|pap> <HOST_PASSIVE|*> <HOST_ACTIVE|*> <SECRET> [IP]

Параметры и поля:

- HOST_PASSIVE: имя узла, который должен быть аутентифицирован;
- HOST_ACTIVE: имя узла, на котором должен быть аутентифицирован HOST_PASSIVE;
- SECRET: пароль узла HOST_PASSIVE на узле HOST_ACTIVE;
- IP: адрес или сеть, который может быть назначен после успешной аутентификации; особенности:
  - имеет приоритет над адресами из permit ip-range интервала;
  - при задании активен, только если HOST_PASSIVE не равен «*».

Также необходимо сообщить службе и клиентскому интерфейсу имена узлов. Особенности задания HOST_PASSIVE и HOST_ACTIVE:

- для интерфейсов l2tp и l2pt@: имена узлов соответсвуют именам LAC (задано командой lac) и LNS (задано командой lns) соответственно;
- для всех других интерфейсов: имена узлов задаются командой ppp localname.

Знак «*» - означает любой узел.

44.2.7 Radius аутентификация

Данную настройку необходимо произвести только в службе динамического интерфейса (например, service pptp). На клиентском интерфейсе необходимо только задать логин и пароль командами pap или chap.

Логин и пароль от клиента, как и в PPP-аутентификации, передается (не напрямую) по протоколу PAP, CHAP или MSCHAPv2, однако дальнейшая аутентификация и выделение IP-адреса осуществляется Radius-сервером.

Какой именно протокол передачи пароля использовать необходимо сообщить в настройке службе командой ppp chap/pap/mschap-v2 require.

Задать Radius-сервер аутентификации:

44.2.7.1 radius authserver <IP>[:PORT]

Задать адрес Radius-сервер аккаутинга:

44.2.7.2 radius acctserver <IP>[:PORT]

Обычно authserver и acctserver - один и тот же сервер.

Задать секрет (SECRET) для доступа к указанному Radius-серверу (IP):

44.2.7.3 radius secret <IP> <SECRET>

Можно задать таймаут соединения с Radius-сервером:

44.2.7.4 radius timeout <VAL>

Для использования Radius-аутентификации необходимо указать это командой ppp active-auth:

44.2.7.5 ppp active-auth <radius | pap-chap >

По умолчанию используется active-auth pap-chap.

44.3 Настройка клиентского интерфейса l2tp

Для настройки LAC необходимо войти в режим настройки L2TP-интерфейса:

(config)# interface l2tp 0

Данный интерфейс представляет собой клиентский динамический интерфейс, появляющийся в системе при создании туннеля с LNS.

44.3.1 Команды настройки интерфейса

После входа в режим настройки L2TP-интерфейса строка приглашения будет иметь следующий вид:

(config—if— l2tp0)#

Команды настройки интерфейса l2tp аналогичны п.«Глобальные настройки службы l2tp» кроме команды Ins и п."Общие настройки для LNS и LAC.", с учетом того, что вместо службы l2tp или LNS в описании команд следует понимать интерфейс l2tp или LAC соответственно.

Также есть специфичные для данного интерфейса команды:

44.3.1.1 lac <NAME>

Эта команда задает имя LAC, используемое для подключения к LNS.

44.3.1.2 srv <IP[:PORT]>

Эта команда указывает IP-адрес и, возможно, порт LNS, с которому нужно создать туннель.

44.3.1.3 redial-interval <TO>

Эта команда задает интервал времени в секундах между попытками повторного соединения.

По умолчанию: 30 сек.

44.3.2 Настройка PPP

См. п. 44.2.4

44.3.3 L2TP аутентификация

См. п. 44.2.5

44.3.4 PPP аутентификация

См. п. 44.2.6

44.4 Прочая работа

44.4.1 Просмотр журналов

Для просмотра журналов службы l2tp следует ввести команду:

44.4.1.1 show service l2tp log

Для просмотра журналов интерфейса l2tp следует ввести команду:

44.4.1.2 show interface log l2tp

IFNUM - номер интерфейса.

44.4.2 Информация о туннелях службы l2tp

Для просмотра информации о подключенных пользователях к службе l2tp можно по команде

44.4.2.1 show service l2tp users

Число анонимных пользователей показано в строке "ANONYMOUS:".

44.4.2.2 service l2tp kill <USER>

Разорвать связь с пользователем USER.

44.5 Пример настройки

44.5.1 L2TP-туннель с локальной аутентификацией и привязкой NAT-группы

Рассмотрим пример настройки серверного и клиентского L2TP-интерфейсов.

Имеется два изделия Dionis-NX:

- L2TP-сервер сети (LNS) с адресом 10.0.0.1/24 из сети 10.0.0.0/24; также имеются nat-list n0 и n1 с некоторыми настройками.
- L2TP-концентратор доступа (LAC) с адресом 10.0.0.2/24 из сети 10.0.0.0/24.

Цель: создать VPN 192.168.1.0/24 между сервером и клиентом в виде L2TP-туннеля.

Настройка службы L2TP:

```
(config)# service l2tp (config—
service—l2tp)# Ins nx1
(config—service—l2tp—nx1)# permit ip—range 192.168.1.2 192.168.1.100
(config—service—l2tp—nx1)# localip 192.168.1.1
(config—service—l2tp—nx1)# permit lac—range 10.0.0.2 10.0.0.100
(config—service—l2tp—nx1)# ppp chap requrie (config—service—
l2tp—nx1)# chap nx2 nxl 123 (config—service—l2tp—nx1)#
template nx2 0 (config—service—l2tp—nx1)# template * 1 (config—
service—l2tp—nx1)# enable
(config)# interface template—l2tp 0 (config—
if—template—l2tp0)# ip nat—group n0
(config)# interface template—l2tp 1 (config—
if—template—l2tp0)# ip nat—group n1
```

Реальные l2tp@* интерфейсы будут создаваться по мере успешного подключения L2TP-клиентов. Для клиента с именем пх2 будет создан интерфейс с привязкой NAT-списка п0. Для остальных клиентов будет привязан NAT-список nl.

Клиентский интерфейс L2TP:

```
(config)# interface l2tp 0 (config—
if—l2tp0)# lac nx2 (config—if—
l2tp0)# srv 10.0.0.1 (config—if—
l2tp0)# chap nx2 nxl 123 (config—
if—l2tp0)# enable
```

После успешного подключения реальный l2tp0 интерфейс будет создан на пх2. Кроме этого на nx1 появится интерфейс l2tp@0 (в том случае, если это 1й l2tp@ интерфейс, иначе вместо "0" может быть другой номер интерфейса).

44.5.2 L2TP-туннель с удаленной Radius-аутентификацией

Как пример, рассмотрим PAP-аутентификацию LAC(10.0.0.2) сервером LNS(10.0.0.1) на Radius-сервере (192.168.0.1).

Настройка LAC (10.0.0.2):

```
(config)# interface l2tp 0 (config—
if—l2tp0)# lac nx2 (config—if—
l2tp0)# srv 10.0.0.1 (config—if—
l2tp0)# chap lacl Insl 123 (config—
if—l2tp0)# enable
```

При соединении с LNS, LAC передает ему PAP логин и пароль.

Настройка LNS (10.0.0.1):

```
(config)# service l2tp (config—service—
l2tp—nx1)# Ins nx1
(config—service—l2tp—nx1)# permit ip—range 192.168.1.2 192.168.1.100
(config—service—l2tp—nx1)# permit lac—range 10.0.0.2 10.0.0.2
(config—service—l2tp—nx1)# localip 192.168.1.1 (config—
service—l2tp—nx1)# ppp chap requrie (config—service—
l2tp—nx1)# pap lacl Insl 123 (config—service—l2tp—nx1)#
radius authserver 192.168.0.1 (config—service—l2tp—nx1)#
radius acctserver 192.168.0.1 (config—service—l2tp—nx1)#
radius secret 192.168.0.1 pas (config—service—l2tp—nx1)#
enable
```

Рассмотрим, что происходит при формировании туннеля:

- При соединении с LNS, LAC передает ему CHAP логин и пароль.
- LNS осуществляет соединение с Radius-сервером, аутентифицируя себя на нем по секрету "pas", заданному командой radius secret 192.168.0.1 pas
- после успешной аутентификации LNS по секрету "pas", LNS передает CHAP логин и пароль на Radius-сервер для аутентификации клиента
- после успешной аутентификации LAC по секрету PAP-логину и паролю, Radius-сервер передает для LNS выделенные для туннеля параметрыдакие как удаленный IP-адрес туннеля и другие параметры.
- в свою очередь LNS передает LAC окончательные параметры туннеля
- на LAC и LNS успешно создан L2TP-туннель.

45. PPTP-туннели

45.1 Введение

Dionis-NX поддерживает протокол PPTP. PPTP - это туннельный протокол типа точка-точка, позволяющий компьютеру устанавливать защищённое соединение с сервером за счёт создания специального туннеля в незащищённой сети. PPTP инкапсулирует кадры PPP в IP-пакеты для передачи по IP-сети.

Настройка PPTP в системе Dionis-NX сводится к настройке серверной части (служба pptp) и клиентской части (интерфейс pptp).

Для клиентских и серверных интерфейсов pptp возможно применение правил NAT, ACL и прочих настроек интерфейсов.

45.2 Настройка службы pptp

Данная служба представляет собой пул динамических интерфейсов, создаваемых в системе по мере подключения клиентов PPTP.

Формат реально создаваемого динамического интерфейса следующий: pptp@<N>

Рассмотрим данный формат:

- pptp@ - это тип интерфейса, знак @ говорит о том, что это серверный интерфейс.
- N - это номер серверного интерфейса

Для настройки службы pptp выполните:

(config)# service pptp

45.2.0.1 log

Включить ведение журналов службы pptp.

По умолчанию: отключены

45.2.0.2 listen <IP>

Эта команда задает сокет IP: 1723 для принятия PPTP запросов.

По умолчанию: 0.0.0.0:1723.

45.2.0.3 localip <IP_START> <N>

Эта команда задает один или несколько IP-адресов, которые будут использоваться в качестве локальных адресов туннеля.

45.2.0.4 remoteip <IP_START> <N>

Эта команда задает один или несколько IP-адресов, которые будут использоваться в качестве удаленных адресов туннеля.

45.2.0.5 template <*|USER>

См. п. ??

45.2.1 Настройка PPP

См. п. 44.2.4

45.2.2 PPP аутентификация

См. п. 44.2.6

45.2.3 RADIUS аутентификация

См. п. 44.2.7

45.3 Настройка интерфейса pptp

Для настройки клиентского интерфейса необходимо войти в режим настройки pptp-интерфейса при помощи команды:

(config)# interface pptp O

Данный интерфейс представляет собой динамический интерфейс, создаваемый в системе после успешного соединения с сервером PPTP.

45.3.1 Настройка интерфейса

45.3.1.1 srv <DOMAIN | IP>

Эта команда задает имя или IP-адрес PPTP-сервера, с которым нужно создать туннель.

45.3.2 Настройка PPP

См. п. 44.2.4

45.3.3 PPP аутентификация

См. п. 44.2.6

45.4 Пример настройки

Рассмотрим пример настройки службы pptp и интерфейса pptp для соединения со службой и создания туннеля pptp.

Имеются два изделия Dionis-NX:

- одно из них - сервер pptp слушающий на интерфейсе ethernetO
- другое - клиент pppoe
- Radius-сервер аутентификации с адресом 192.168.32.201

Цель: создать pptp VPN 192.168.1.0/24 между сервером и клиентом в виде pptp-туннеля.

Настройка службы pptp:

```
(config)# service—pptp (config—service—
pptp)# ac acl (config—service—pptp)# srv inet
(config—service—pptp)# localip 192.168.1.1 1
(config—service—pptp)# remoteip 192.168.1.2 100
(config—service—pptp)# ppp mschap—v2 require
(config—service—pptp)# ppp localname srv (config—
service—pptp)# radius acctserver 192.168.32.201
(config—service—pptp)# radius authserver 192.168.32.201
(config—service—pptp)# radius secret 192.168.32.201 123
(config—service—pptp)# ppp active—auth radius (config—
service—pptp)# enable
```

Настройка клиентского интерфейса pppoe:

```
(config)# interface pppoe 0 (config—if—
pppoeO)# ac acl
(config—if—pppoeO)# srv inet
(config—if—pppoeO)# chap cli srv pas
(config—if—pppoeO)# ppp localname cli
(config—if—pppoeO)# enable
```

Настройка шаблона интерфейса template-pptp командой template аналогична примеру для службы l2tp (см. п. 44.5.1).

46 PPPOE-туннели

46.1 Введение

Dionis-NX поддерживает протокол PPPOE. PPPOE (англ. Point-to-point protocol over Ethernet) — сетевой протокол канального уровня передачи кадров PPP через Ethernet. Предоставляет такие дополнительные возможности, как аутентификация, сжатие данных, шифрование.

Настройка PPPOE в системе Dionis-NX сводится к настройке серверной части (служба pppoe) и клиентской части (интерфейс pppoe).

Для клиентских и серверных интерфейсов pptp возможно применение правил NAT, ACL и прочих настроек интерфейсов.

46.2 Настройка службы PPPOE

Данная служба представляет собой пул динамических интерфейсов, создаваемых в системе по мере подключения клиентов PPPOE.

Формат реально создаваемого динамического интерфейса следующий: pppoe@

Рассмотрим данный формат:

- pppoe@ - это тип интерфейса, знак @ говорит о том, что это серверный интерфейс.
- N - это номер серверного интерфейса

Для настройки службы pppoe выполните: ***(ssp – не понятно***)

(config)# service pppoe

46.2.0.1 ac <NAME>

Команда задает имя концентратора доступа. Клиенты могут указать в своих PPPOE-запросах, к какому именно концентратору доступа они хотят подсоединиться.

46.2.0. 2 srv <NAME>

Команда задает имя одного или нескольких сервисов, предоставляемых данным концентратором доступа. Клиенты могут указать в своих PPPOE-запросах, какие именно сервисы они хотят использовать на данном концентраторе доступа.

46.2.0. 3 iface <IFACE>

Эта команда задает интерфейс, на котором принимать запросы PPPOE-клиентов.

46.2.0.4 localip <IP_START>

Эта команда задает начальный IP-адрес серверного конца туннелей.

46.2.0.5 remoteip <IP_START>

Эта команда задает начальный IP-адрес клиентского конца туннелей (IP_START)

46.2.0.6 frames <DISCOVERY-TYPE> <SESSION_TYPE>

Этак команда переопределяет стандартные значения типов PPPOE-фрэймов. Можно задать новое значение типа для Discovery PPPOE фрэйма (вместо стандартного 0x8863) и новое значение типа для Session PPPOE фрэйма (вместо стандартного 0x8864).

46.2.0.7 max-client-sessions <N>

Установить максимальное число сессий для одного клиента (определяемого по MAC-адресу).

По умолчанию: максимальное число сессий для всех клиентов - 65534.

46.2.0.8 radnom-sid

Назначать ID сессии клиентам из разрешенного интервала ID сессий не последовательно^ случайным образом.

46.2.1 Настройка PPP

См. п. 44.2.4

46.2.2 PPP аутентификация

См. п. 44.2.6

46.2.3 RADIUS аутентификация

См. п. 44.2.7

46.3 Настройка интерфейса pppoe

Для настройки клиентского интерфейса необходимо войти в режим настройки pppoe-интерфейса при помощи команды:

(config)# interface pppoe 0

46.3.1 Настройка интерфейса

46.3.1.1 log

Эта команда включает ведение журналов.

По умолчанию: отключены.

46.3.1.2 ac <NAME>

Команда задает имя концентратора доступа, к которому подключаться.

46.3.1.3 srv <NAME>

Подключаться только к тем концентраторам доступа,заданным командой ac, которые предоставляют сервис с указанным именем NAME.

46.3.1.4 mac <MAC>

Подключаться только к тем концентраторам доступа, которые имеют указанный MAC-адрес.

46.3.2 Настройка PPP

См. п. 44.2.4

46.3.3 PPP аутентификация

См. п. 44.2.6

46.4 Пример настройки

Рассмотрим пример настройки службы pppoe и интерфейса pppoe для соединения со службой и создания туннеля PPPOE.

Имеются два маршрутизатора Dionis-NX:

- один из них - сервер pppoe слушающий на интерфейсе ethernet0
- другой - клиент pppoe
- Radius-сервер аутентификации с адресом 192.168.32.201

Цель: создать pppoe VPN 192.168.1.0/24 между сервером и клиентом в виде pppoe-туннеля.

Настройка службы pppoe:

```
(config)# service—pppoe (config—service—
pppoe)# ac acl (config—service—pppoe)# srv
inet (config—service—pppoe)# localip
192.168.1.1
(config—service—pppoe)# remoteip 192.168.1.2 100 (config—
service—pppoe)# ppp mschap—v2 require (config—service—
pppoe)# ppp localname srv (config—service—pppoe)# radius
acctserver 192.168.32.201 (config—service—pppoe)# radius
authserver 192.168.32.201 (config—service—pppoe)# radius
secret                                    192.168.32.201 123
(config—service—pppoe)# ppp active—auth radius (config—
service—pppoe)# enable
```

Настройка клиентского интерфейса pppoe:

```
(config)# interface pppoe 0 (config—
if—pppoe0)# ac acl (config—if—
pppoe0)# srv inet (config—if—pppoe0)#
chap cli srv 123 (config—if—pppoe0)#
ppp localname cli (config—if—pppoe0)#
enable
```

Настройка шаблона интерфейса template-pppoe командой template аналогична примеру для службы l2tp (см. п. 44.5.1).

47 Механизмы качества обслуживания (QoS)

Механизмы QoS Dionis-NX позволяют классифицировать проходящий через маршрутизатор трафик и применять к различным классам разную политику обслуживания. Обработке подвергается исходящий трафик интерфейса, если к нему применена политика обслуживания, которая позволяет задать:

- гарантированную полосу пропускания;
- максимальную полосу пропускания;
- приоритет;

47.1 Классификация

Для классификации трафика используются списки отображения классов (ip class-map). Каждый список представляет собой правила классификации с набором критериев отбора. При выполнении всех правил списка принимается решение о принадлежности трафика к описываемому классу.

Для создания списка отображения класса, используется команда: ip class-map <имя класса> в режиме configure.

Например:

```
DionisNX(config)# ip class—map web
DionisNX(config—стар—web)# match tcp dport 80
DionisNX(config—стар—web)# match tcp sport 80
```

После выполнения этих команд создается класс web, к которому будет отнесен tcp-трафик с 80 и на 80 TCP-порт.

Для работы с элементами данного списка применяется тот же подход, что и при работе со списками контроля доступа. Команда no <номер правила>|all - удаляет соответствующие элементы. Правила отбора, начинающиеся с числового префикса, добавляют правило в заданную позицию. Удаление списка осуществляется командой: no ip class-map <имя>.

Для классификации трафика используются два правила: match и exclude, после которых следуют критерии отбора (подмножество критериев списков контроля доступа). Список просматривается сверху вниз, при этом последовательно анализируются критерии отбора каждого правила. При выполнении критериев match трафик начинает относиться к заданному классу. При выполнении критерия exclude снимается принадлежность трафика к заданному классу. В обоих случаях, анализ дальнейших правил продолжается. В качестве критериев отбора могут быть применены значения TOS и DSCP.

Для просмотра информации о списках отображения классов используется команда: show ip class-map [имя]. Команда определена для enable-режима. Если не указано имя списка, будет показана информация о всех списках.

Например (из режима configure):

DionisNX(config)# do show ip class—map web

Несмотря на возможность создания неограниченного количества классов, количество *активных* (т.е. используемых в загруженных политиках или других подсистемах) ограничено и не может превышать 64. При превышении этого лимита на консоль и в лог выводится предупреждение, и команда, требующая активации дополнительных классов может не сработать.

47.2 Политика обслуживания policy-map

Политика обслуживания описывает то, каким образом обслуживаются различные классы трафика. Для создания/редактирования политики обслуживания необходимо выполнить команду: ip policy-map <имя политики> из режима configure.

Например:

DionisNX (config)# ip policy—map outworld
DionisNX (config —pmap—outworld)#

При этом, произойдет вход в режим редактирования политики. Каждая политика состоит из списка правил. Правило определяет качество обслуживания конкретного класса и задается в форме: class <имя класса> rate сгарантированная скорость> [другие необязательные параметры]. Для пакета, попадающего в несколько классов политики, выполнится первое правило, которому он соответствует. Для пакетов, не попадающих ни в один из классов политики, необходимо создать правило при помощи команды default rate сгарантированная скорость> [другие необязательные параметры].

В случае наличия класса A команда class A [...] изменит существующее правило вместо добавления нового. Для удаления правила используется команда no class <имя класса>

Для просмотра редактируемого списка удобно воспользоваться командой: do show.

Пропускная способность описывается в правилах в виде числа с необязательным постфиксом.

Постфикс	Смысл
нет	бит в секунду
kbps	килобайт в секунду
mbps	мегабайт в секунду
kbit	килобит в секунду
mbit	мегабит в секунду
bps	байт в секунду

Основные параметры правил политики:

Название параметра	Смысл
rate спропускная способность>	Гарантированная пропускная способность
ceil спропускная способность>	Пиковая пропускная способность
priority <число>	Приоритет обработки (чем меньше значение - тем выше приоритет)

Если в правиле не указан параметр ceil, то пропускная способность заданного класса не будет превышать rate, то есть отсутствие параметра ceil равносильно наличию параметра ceil, равному rate.

Для корректной работы параметра ceil на интерфейсе должна быть задана пропускная способность (команда bandwidth).

Для удаления списка политики необходимо использовать команду: no policy-map <имя> в режиме configure.

В качестве примера приведем реализацию простой приоритезации на основе tos-значений:

```
  ip class—map prt0
    match tos 0/0xe0

 ip class—map prtl
match tos 0x20/0xe0
 !
 ip class—map prt2
match tos 0x40/0xe0
 !
 ip class—map prt3
match tos 0x60/0xe0
 !
 ip class—map prt4
match tos 0x80/0xe0
 !
 ip class—map prt5
match tos 0xa0/0xe0
 !
 ip class—map prt6
match tos 0xc0/0xe0
 !
   ip class—map prt7
match tos0xe0/0xe0

  ip policy—map prio

 class  prt0  rate  1kbit  ceil   100000mbit     priority     7
class  prt1  rate  1kbit  ceil  100000mbit     priority     6
class  prt2  rate  1kbit  ceil  100000mbit     priority     5
class  prt3  rate  1kbit  ceil  100000mbit     priority     4
class  prt4  rate  1kbit  ceil  100000mbit     priority     3
class  prt5  rate  1kbit  ceil  100000mbit     priority     2
class  prt6  rate  1kbit  ceil  100000mbit     priority     1
class prt7 rate 1kbit ceil       100000mbit     priority     0
```

Для просмотра списка/списков политик обслуживания следует использовать команду show ip policy-map [имя], доступную из enable-режима. Например:

```
DionisNX(config)# do show ip policy—map prio
```

Если имя не задано, будут показаны все политики policy-map.

Для просмотра загруженной в ядро действующей кофигурации используется команда show ip policy [интерфейс].

Также на классы policy-group можно повесить другие политики, например, class cls1 rate 1mbit red-map a, или class cls1 rate 1mbit policy-map poll, что позволяет создавать иерархичность политик. Подробнее про поведение политик обслуживания в этом случае будет рассказано далее.

47.3 Пропускная способность интерфейса

Пропускная способность интерфейса задаётся командой bandwidth:

```
DionisNX(config)# interface ethernet 0
 DionisNX(config — if—ethernet0)# bandwidth rate 10mbit burst 150000 latency 5msec
```

47.4 Политики обслуживания prio-map и prio-tos-map

Политики обслуживания prio-map и prio-tos-map позволяют задать приоритет пакетов без явного выделения полосы пропускания для каждой из очередей. Так, если заданы три очереди, то сначала будут отправляться пакеты из первой очереди, потом если первая очередь пуста, то из второй, и только если и первая и вторая очередь пусты, то трафик будет отправляться из третьей очереди. В политике обслуживания prio-map пакет попадает в очередь исходя из class-map'ов, в которые он входит, а в политике prio-tos-map задаётся соответствие между полем TOS и очередью, что позволяет создать простую схему приоретизации трафика без лишних настроек. На очереди prio-map и prio-tos-map также можно повесить другие политики обслуживания. Кроме того, можно ограничить скорость очереди при помощи параметра rate. Параметры burst, limit и peakrate соответствуют таким же параметрам в команде bandwidth.

Примеры настройки:

```
 ip class—map ssh
match tcp sport 22
match tcp dport 22
 !
 ip class—map http
match tcp sport 80
match tcp dport 80
match tcp sport 443
match tcp dport 443
 !
 ip prio-map prio
   1  band class ssh rate lmbit burst 100000 limit 1000000
   2  band default fq
   3  band class http
!
    interface ethernet 0
     ip prio—group prio
```

i

В данном случае наивысший приоритет у класса ssh, а наименьший — у класса http. Прочий трафик попадает в очередь default, а из неё - в политику fq.

Также рассмотрим, как с помощью prio-tos-map можно задать поведение, соответствующее поведению по умолчанию на многих Linux-системах:

```
ip prio—tos—map prio
  1   band
  2   band
  3   band
  tos-map 2 3 3 3 2 3 1 1 2 2 2 2 2 2 2 2
```

Таким образом трафик будет попадать в различные очереди в зависимости от битов 3-6 поля TOS.

47.5 Политика обслуживания red-map

Политика обслуживания red-map является политикой борьбы с перегрузками, которая позволяет достичь более равномерного ограничения трафика за счёт использования вероятностного метода отбрасывания пакетов при перегрузке.

Для создания red-map используется команда ip red-map. Параметры:

Параметр	Значение
limit	Ограничение на размер очереди
avpkt	Средний размер пакета
bandwidth	Пропускная способность RED (используется для рассчёта вероятности отбрасывания, RED не ограничивает пропускную способность сам по себе)
ecn	Использовать Explicit Congestion Notification вместо отбрасывания пакетов при перегрузке

Пример:

```
ip red —map red limit 500000 avpkt 1000 bandwidth 10mbit ecn
```

47.6 Политика обслуживания codel-map

Политика обслуживания codel-map является политикой борьбы с перегрузками, в которой вместо ограничения размера очереди ограничивается максимальное время нахождения пакета в очереди.

Параметр target используется для задания целевого времени нахождения пакета в очереди, а параметр ecn позволяет использовать механизм Explicit Congestion Notification

47.7 Политика обслуживания fq_codel-map

Политика, которая отличается от codel-map тем, что пакеты разделяются на некоторое количество (определяется параметром flow, по умолчанию 1024) виртуальных очередей на основе соединения, которому они принадлежат. В первую очередь отбрасываются (или помечаются при использовании ecn) пакеты, которые принадлежат наиболее загруженному соединению. Позволяет обеспечить одновременно быстрое прохождение пакетов и отсутствие потерь на низкозагруженных соединениях без сложной настройки.

47.8 Политика обслуживания sfq-map

Политика sfq-map также содержит некоторое количество виртуальных очередей, и отбрасывает в первую очередь пакеты из более нагруженных соединений. В отличие от fq_codel-map, sfq-map отбрасывает пакеты при заполнении очереди, а не при превышении времени нахождения пакета в очереди. Также возможно использовать sfq-map в комбинации с RED. Параметр limit определяет максимальное количество пакетов в виртуальной очереди, параметр perturb — время, через которое меняется хэш-функция, относящая пакет к какой-либо очереди.

Для определения очереди используется хэш от IP-адреса источника, IP-адреса назначения и портов TCP/UDP.

47.9 Политика обслуживания gred-map

Политика обслуживания gred-map похожа на политику red-map, но позволяет задать несколько red-очередей с разным приоритетом. Пакет попадает в одну из 16 очередей на основе значения младших четырёх байтов поля DSCP. В случае, если необходимая очередь не задана, пакет попадает в очередь default.

Пример:

```
ip gred—map gred
default 0
  queue 0 limit 20000 avpkt 1000 priority 0
queue 1 limit 20000 avpkt 1000 priority  1
queue 2 limit 20000 avpkt 1000 priority 2
bandwidth 10000mbit
```

47.10 Ingress

Также, существует возможность ограничения входящего трафика. Для этого используется команда ingress: ingress rate <скорость>.

```
DionisNX(config)# interface ethernet 0
DionisNX(config—if—ethernet0)# ingress rate 10mbit
```

47.11 Привязка политики к интерфейсу

Политика обслуживания начинает действовать на исходящий трафик только после привязки политики к интерфейсу. Чтобы осуществить такую привязку, необходимо перейти в режим настройки интерфейса и выполнить команду: ip policy-group <имя политики>, например:

```
DionisNX(config)# interface ethernet 0
DionisNX(config —if—ethernet0)# ip policy—group prio
```

Для удаления связи с интерфейсом, необходимо выполнить команду: no ip policy-group <имя>, например:

```
DionisNX(config — if—ethernet0)# no ip policy—group prio
DionisNX(config —if—ethernet0)# do show
```

Аналогично привязываются другие политики, например, ip red-group red

Для просмотра информации о политиках на выбранном интерфейсе (или на всех интерфейсах) следует использовать команду: show ip policy [интерфейс] в enable-режиме, например:

```
DionisNX# show ip policy ethernet 0
```

47.12 Туннельный трафик

Классификация трафика с помощью ip class-map осуществляется до первого удачного сопоставления с классом. Обычно этот нюанс не имеет значения при администрировании, однако в случае туннельного трафика это поведение иногда необходимо учитывать.

Например, рассмотрим трафик GRE туннеля. Пусть имеется класс:

```
DionisNX(config—cmap—gre)# match gre
```

Тогда при приходе трафика протокола GRE он будет классифицирован. Затем, из туннеля будет извлечен внутренний пакет (предположим, UDP) и этот пакет уже не будет классифицироваться - с ним будет по прежнему связан уже ранее сопоставленный класс gre.

На самом деле, обычно нет никакого смысла классифицировать трафик до того, как он будет извлечен из туннеля, но иногда это все-таки необходимо.

Кроме того, можно привести более наглядный пример:

```
DionisNX(config—cmap—udp)# match udp
```

Предположим, что UDP-пакеты заворачиваются в gre-туннель на выходе из маршрутизатора. Тогда класс GRE-пакетов останется тем же, что при классификации его ранее, как трафик UDP. Это может быть нежелательным в некоторых случаях.

В рассматриваемых ситуациях могут оказаться полезными классы типа tunnel. Для того, чтобы отметить class-map как туннельный, в режиме редактирования класса нужно выполнить команду:

DionisNX(config—cmap—gre)# tunnel

Для отмены режима туннеля:

DionisNX(config—cmap—gre)# no tunnel

Туннельный класс работает следующим образом:

1. Для обычного трафика, не пришедшего из туннеля, сопоставление с классом работает обычным образом.
2. Для туннельного трафика, он сопоставляется с классом до извлечения из туннеля и (в случае успешного первого сопоставления) после извлечения из туннеля.
3. Для туннелируемого трафика (который будет завернут в туннель) выполняется сопоставление до заворачивания и (в случае успешного 1-го сопоставления) после упаковки в туннель.
4. В обоих случаях (2 и 3) при втором сопоставлении с классом трафик уже промаркирован этим классом, но он может быть исключен из класса с помощью exclude.
5. При втором сопоставлении с классом, если трафик оказывается исключенным из класса (на шаге 4), происходит классификация обычными классами (без режима tunnel).

Например, рассмотрим несколько вариантов применения классов:

DionisNX(config—cmap—gre)# tunnel
DionisNX(config—cmap—gre)# match gre
DionisNX(config—стар—gre)# exclude icmp

Здесь в класс gre попадут:

- gre-дата граммы;
- весь трафик внутри gre-датаграмм, кроме icmp-датаграмм.

DionisNX(config—cmap—esp)# tunnel
DionisNX(config—cmap—esp)# exclude
DionisNX(config—cmap—esp)# match esp

Здесь в класс esp попадут:

- весь esp-трафик;
- все содержимое из esp-туннелей будет исключено из класса esp.

DionisNX(config—cmap—gre)# tunnel
DionisNX(config—cmap—gre)# exclude
DionisNX(config—cmap—gre)# match gre
DionisNX(config—cmap—gre)# match udp

Здесь в класс gre попадут:

- весь gre-трафик;
- udp-трафик из дге-туннелей;
- другой трафик, извлеченный из туннелей gre, будет исключен из класса gre;

48 Расширенная статическая маршрутизация

В некоторых случаях функций статической маршрутизации может быть недостаточно. Например, когда необходимо маршрутизировать разные подсети через разные узлы. Для этого можно воспользоваться расширенным механизмом статической маршрутизации.

Внимание!!! Приоритет расширенных правил статической маршрутизации ниже, чем у правил маршрутизации, задаваемых командой ip route, поэтому, если используются расширенные правила статической маршрутизации, предварительно необходимо удалить правила ip route default. Ели этого не сделать, весь трафик будет маршрутизироваться правилами, задаваемыми ip route, так как он попадет под правило маршрута по умолчанию.

Задание маршрута IPv4 с использованием расширенных правил выполняется с помощью команды ip policy-route, например:

```
DionisNX(config)# ip policy—route src 192.168.32.0/24 gateway 192.168.33.254
```

Команда имеет следующий синтаксис:

```
[префикс] ip policy—route справила отбора>
[class <ip class—list>] cgateway шлюз|interface сетевой интерфейс!blackhole>
```

Правила ip policy-route упорядочены, поиск маршрута осуществляется последовательно, правило за правилом. При этом, администратор может вставлять правила в произвольную позицию и удалять правила с заданной позицией, явно используя порядковый номер правила, например:

```
DionisNX(config)# ip policy—route src 192.168.32.0/24 gateway 192.168.33.254
DionisNX(config)# 1 ip policy—route icmp src 192.168.32.0/24 gateway 192.168.33.1
DionisNX(config)# do show ip policy—route config
!
1 ip policy—route icmp src 192.168.32.0/24 gateway 192.168.33.1
2 ip policy—route src 192.168.32.0/24 gateway 192.168.33.254
DionisNX(config)# no ip policy—routing 1
DionisNX(config)# do show ip policy—route config
!
1 ip policy—route src 192.168.32.0/24 gateway 192.168.33.254
```

В качестве правил отбора используются такие-же правила, как и в списках доступа (ip access-list). Кроме того, можно задать класс трафика (см. главу "Механизмы качества облуживания (QoS)") с помощью параметра class.

Назначение трафика задается в виде IP-адреса, сетевого интерфейса или blackhole - для уничтожения трафика.

Для просмотра информации о правилах следует использовать команду show ip policy-route [config]. Для быстрого удаления всех правил следует использовать команду no ip policy-route all.

Как уже было сказано выше, при использовании ip policy-route совместно с правилами ip route, необходимо, чтобы маршрут по умолчанию также был задан с помощью ip policy-route. Для этого существует специальная команда: ip policy-route default

DionisNX(config)# no ip route default

DionisNX(config)# ip policy—route default gateway 192.168.33.1

Синтаксис команды:

[префикс] ip policy—route default [src <адрес/сеть>] [dst <адрес/сеть>] <gateway шлюз|interface сетевой интерфейс!blackhole> [priority <приоритет>]

Может быть задано несколько маршрутов "по умолчанию". Работа с маршрутами по умолчанию, в отличие от правил ip policy-route, ведется не по номерам, а по их описанию. Приоритет правил определяется параметром priority. Для просмотра правил "по умолчанию" используйте show ip policy-route default.

DionisNX(config)# ip policy—route default gateway 192.168.33.1 priority 1
DionisNX(config)# ip policy—route default gateway 192.168.22.1 priority 2
DionisNX(config)# no ip policy—route default gateway 192.168.22.1 priority 2
DionisNX(config)# do show ip policy—route default

Расширенные правила маршрутизации поддерживают механизм icmp-проб. При этом, соответствующий маршрут будет активным только при успешном прохождении icmp-пробы. Для задания icmp-пробы, необходимо дописать к правилу следующие аргументы:

keepalive <1P адрес> [via интерфейс] <время ожидания> [число попыток]

Например:

DionisNX(config)# ip policy—route gateway 192.168.33.1 keepalive 192.168.33.1 5

Правила ip policy-route и ip policy-route default выполняются после правил статической маршрутизации. Однако, существует возможность задать правила, которые действуют до всех остальных правил маршрутизации. Для создания такого правила, необходимо добавить параметр preroute в команде ip policy-route.

DionisNX(config)# ip policy—route preroute in ethernet 0 interface ethernet 1.1

Маршрутизация IPv6

Для задания маршрутов IPv6 (при условии, что этот протокол был ранее активирован) используйте команды с префиксом ip6. Ниже приводится таблица соответствий команд IPv4 и IPv6.

IPv6 команда	IPv4 команда
[no] ip6 policy-route	[no] ip policy-route
[no] ip6 policy-route default	[no] ip policy-route default
show ip6 policy-route [default]	show ip policy-route [default]

49. Динамическая маршрутизация

49.1 Списки

Этот раздел в настоящее время отсутствует.

См. документацию Cisco и FRRouting.

49.2 RIP

49.2.1 Описание протокола RIP

RIP обеспечивает маршрутизацию внутри автономной системы (AC), RFC 2453 «RIP Version 2».

RIP использует UDP (порт 520) в качестве транспорта для анонсируемых маршрутов. Пакеты UDP инкапсулируются в мультикаст-датаграммы (1P-адрес: 224.0.0.9).

RIP-маршрутизатор отправляет и принимает мультикаст-датаграммы в широковещательных сетях. При широковещании, фактически, не происходит установления соседства, так как RIP-маршрутизаторы анонсируют маршруты не для конкретного соседа, а для всех.

Если сеть не поддерживает широковещание (например, NBMA), то возможно использование уникаст-датаграмм. В этом случае между RIP-маршрутизаторами необходимо установить сосед-ские отношения.

RIP - дистанционно-векторный протокол. RIP-маршрутизаторы присваивают маршрутам мет-рики (дистанции), по сумме которых выбирается наилучший путь. Маршрут и метрика составляют вектор, который передается соседнему RIP-маршрутизатору в анонсе. Сосед увеличивает метрики полученного вектора на величину метрики маршрута к источнику анонса и добавляет к вектору свои маршруты с метрикой. Полученный вектор анонсируется другим соседям.

Фактически, метрика, это счетчик хопов, число маршрутизаторов на пути к цели. Напрямую подключенная сеть имеет метрику 0, недостижимая сеть - метрику 16. Такой малый диапазон метрик делает RIP не пригодным для сетей большой вложенности.

49.2.2 RIP-маршрутизатор

49.2.2.1 Включение RIP-маршрутизатора (router rip)

Первое, что требуется для начала настройки RIP, это включить RIP-маршрутизатор. Сделать это можно командой:

```
(config)# router rip
```

49.2.2.2 Интерфейсы RIP-маршрутизатора (network)

Для работы RIP-маршрутизатора требуется указать, какие интерфейсы 1P-маршрутизатора следует использовать. Это можно сделать либо, явно указав имя и номер интерфейса, либо, объявив сеть, настроенную на интерфейсе. Сделать так можно командой:

(config—rip)# network <ip/m> | <iface>

<ip/m> - IP-адрес и маска сети.

<iface> - имя и номер интерфейса.

Пример использования RIP-маршрутизатором интерфейса Ethernet 1 (IP-адрес 192.168.1.1).

```
!
interface ethernet 0
  ip address 192.168.0.1/24
  enable
!
interface ethernet 1
  ip address 192.168.1.1/24
  enable
!
interface ethernet 2
  ip address 192.168.2.1/24
  enable
!

(config—rip)# network 192.168.1.1/24
```

49.2.2.3 Объявление соседа (neighbor)

Для нешироковещательных сетей следует явно указать соседа. Следующая команда позволяет организовать с соседом соединение точка-точка:

(config—rip)# neighbor <ip>

<ip> - IP-адрес соседнего RIP-маршрутизатора.

49.2.2.4 Пассивный интерфейс (passive-interface)

Иногда требуется запретить рассылку анонсов с определенных интерфейсов. Сделать это можно, обозначив интерфейс пассивным при помощи команды:

(config—rip)# passive—interface default| <iface>

passive-interface default - делает все интерфейсы пассивными, в этом случае RIP-маршрутизатор только принимает анонсы.

На пассивном интерфейсе анонсы принимаются и сеть этого интерфейса анонсируется соседям. Также можно статически указать соседа командой neighbor и тогда для такого соседа анонсы будут и приниматься, и отправляться.

49.2.2.5 Версия (version)

По умолчанию RIP-маршрутизатор принимает анонсы 1-ой и 2-ой версии протокола, а отправляет только 2-ой версии. Если требуется использовать определенную версию протокола, сделать это можно командой:

(config — rip)# version 112

Восстановить значения по умолчанию можно командой:

(config—rip)# no version

Можно установить использование определенной версии только на выбранных интерфейсах про помощи команды:

(config—if—ethernet)# ip rip receive|send version 112

ip rip receive - версия для приема,

ip rip send - версия для передачи.

49.2.2.6 Избежание петель (split-horizon)

RIP избегает петель при помощи механизма «Split horizon». Суть работы split horizon в том, чтобы не отправлять анонсы о сетях в интерфейс, через который они были получены.

По умолчанию split horizon включен. Если требуется его выключить, сделать это можно командой на определенном интерфейсе:

(config—if—ethernet)# no ip rip split—horizon

Вернуть работу по умолчанию:

(config—if—ethernet)# ip rip split—horizon

Механизм split horizon можно также настроить таким образом, что RIP-маршрутизатор будет отправлять анонс о сети в тот же интерфейс, через который его получил, но при этом маршрут в эту сеть будет иметь метрику 16 «недостижим». Такое поведение можно настроить командой:

(config—if—ethernet)# ip rip split—horizon poisoned—reverse

49.2.2.7 Таймеры (timers)

RIP-маршрутизатор отправляет анонсы каждые 30 секунд, если анонс с определенным маршрутом не приходит в течение 180 секунд, маршрут помечается, как неиспользуемый, но пока остается в таблице маршрутизации, если еще через 120 секунд анонс не приходит, то маршрут удаляется.

Если требуется изменить интервалы отправки и ожидания, сделать это можно командой:

(config—rip)# tinners basic <update> <timeout> <garb_collect>

<update> - интервал отправки анонсов, по умолчанию 30 секунд.

<timeout> - интервал, после которого маршрут помечается как неиспользуемый, по умолчанию 180 секунд.

<garb_collect> - интервал после которого неиспользованные маршруты удаляются, по умолчанию 120 секунд.

Вернуть значения по умолчанию можно командой:

(config—rip)# no tinners basic

49.2.3 Метрики

49.2.3.1 Метрика по умолчанию (default-metric)

По умолчанию RIP-маршрутизатор присваивает перераспределенным маршрутам метрику 1. Изменить это значение можно командой:

(config—rip)# default—metric <n>

Это работает для всех маршрутов, кроме непосредственно подключенных (connected). Изменить метрику для них можно либо командой «redistribute connected metric», либо командой «offset-list».

49.2.3.2 Изменение метрики по списку (offset-list)

По умолчанию RIP-маршрутизатор увеличивает метрику маршрутов на 1. Существует механизм, позволяющий увеличивать метрику на определенное значения для маршрутов, выбранных по списку доступа. Сделать это можно при помощи команды:

(config—rip)# offset—list <racl_name> in|out cmetrio [<iface>]

<racl_name> - имя списка доступа, по которому отбирать маршруты,

in|out - применять метрику к принимаемым либо отправляемым маршрутам,

cmetrio - значение, на которое увеличивается метрика; по умолчанию 1.

<iface> - определенный интерфейс.

Пример увеличения метрики на 10 для входящих анонсов, содержащих маршруты в сеть **10.0.0.0/8**

```
!
ip access—list myacl
   1 permit dst 10.0.0.0/8


(config—rip)# offset—list myacl in 10
```

49.2.3.3 Административная дистанция (distance)

AD используется для изменения приоритета путей, полученных от разных протоколов. Работает после выбора лучшего пути до помещения пути в таблицу маршрутизации. Чем меньше AD, тем приоритетнее путь. Значения AD: подключенный интерфейс 0, статический маршрут 1, EBGP 20, OSPF 110, RIP 120, IBGP 200.

Изменить административную дистанцию можно командой:

(config—rip)# distance <n> [<ip/m> [<racl_name>]]

<n> - новое значение дистанции, по умолчанию 120.

<ip/m> - префикс источника маршрута, дистанцию будет изменяться только для маршрутов, полученных от этих источников.

<racl_name> - список доступа с параметрами источников маршрута.

49.2.4 Аутентификация

Аутентификация возможна только для RIP версии 2. При использовании аутентификации следует принудительно установить версию протокола 2 при помощи команды «version» для обеспечения защиты таблицы маршрутизации. Если этого не сделать, то RIP-маршрутизатор по умолчанию будет принимать анонсы, как аутентифицированные (версии 2) так и не аутентифицированные (версии 1).

Настройка аутентификации может быть выполнена с использованием простого текстового пароля или с использованием хэшей MD5.

49.2.4.1 Простой текстовый пароль (authentication mode text)

Выбор типа аутентификации по паролю выполняется командой:

(config—if—ethernet)# ip rip authentication mode text

Текстовый пароль задается при помощи команды:

(config—if—ethernet)# ip rip authentication string <passw>

<passw> - пароль не более 16-ти смволов.

Пример настройки аутентификации с текстовым паролем:

```
!
interface ethernet 0
  ip rip authentication string secertpasswd
  ip rip authentication mode text
!
router rip
  version 2
!
```

49.2.4.2 Хэш MD5 (authentication mode md5)

Выбор типа аутентификации по хэшу выполняется командой:

(config — if— ethernet)# ip rip authentication mode md5

Для типа MD5 можно дополнительно настроить режим совместимости

(config —if—ethernet)# ip rip authentication mode md5 [auth—length old — ripd|rfc]

auth-length old-ripd - совместимость со старыми реализациями ripd.

auth-length rfc - совместимость с реализациями по RFC.
Ключи, для которых высчитывается хэш MD5, задаются командой:

(config—if—ethernet)# ip rip authentication key—chain <name>

<name> - имя связки ключей

Сами ключи создаются в режиме конфигурирования связки ключей, попасть в который можно при помощи команды:

(config)# router key chain <name>

<name> - имя связки ключей, которое будет использовано при аутентификации.

В этом режиме можно создать несколько ключей по команде:

(config—router—keychain—name)# key <n>

<n> - номер ключа.

Создав ключ, следует создать его содержимое командой:

(config—router— keychain—name—n)# key—string <str>

<str> - строка с ключом.

Дополнительно для каждого ключа можно задать свои сроки действия.

Сроки действия ключа на приём задаются при помощи команды:
(config — router— keychain—?—?)# accept—lifetime <HH>:<MM>:<SS> <month> <day> <year>
infinite|(duration <secs>)|(<HH>:<MM>:<SS> <month> <day> <year>)

Срок действия ключа на отдачу задаются при помощи команды:

(config — router— keychain—?—?)\# send —lifetime <HH>:<MM>:<SS>

infinite|(duration)|(::)

Первый параметр <HH>:<MM>:<SS> <month> <day> <year> - соответственно час, минута, секунда, месяц, день и год начала действия срока. Конец срока можно задать в таком же виде. Либо в виде продолжительности в секундах «duration <secs>», либо бесконечным «infinite».

Пример настройки аутентификации в режиме MD5 с использованием цепочки ключей:

```
!
interface ethernet 0
  ip rip authentication key—chain mykey
  ip rip authentication mode md5
!
router rip
  version 2
!
key chain mykey
  key 1
  key—string secretpasswd
i
```

49.2.5 Анонсирование

49.2.5.1 Перераспределение (redistribute)

RIP-маршрутизатор распространяет маршруты, полученные по протоколу RIP. Помимо этих маршрутов, возможно анонсировать сети путем перераспределения маршрутов из таблицы IP-маршрутизатора в BGP-маршрутизатор. Сделать это можно командой:

```
(config—rip)# redistribute kernel|connected|static|ospf|bgp [metric <n>] [route—map
        <rmap_name>]
```

redistribute kernel - анонсирует маршруты, используемые ядром linux.

redistribute connected - анонсирует маршруты интерфейсов, подключенных к коммутатору.

redistribute static - анонсирует статические маршруты, т.е. прописанные вручную администратором.

redistribute ospf - анонсирует маршруты, полученные по OSPF.

redistribute bgp - анонсирует маршруты, полученные по bgp.

metric <n> - метрика, с которой будут анонсированы эти маршруты,

route-map <rmap> - анонсирует сеть с параметрами карты маршрута.

49.2.5.2 Маршрут по умолчанию (default-information originate)

Маршрут по умолчанию можно сообщить соседу и при этом его можно не создавать в таблице IP-маршрутизации. Сделать так можно командой:

```
(config—rip)# default—information originate
```

49.2.5.3 Статический маршрут (route)

RIP-маршрутизатор позволяет создавать в таблице RIP-маршрутизации статические маршруты, которые анонсируются, но не попадают в таблицу IP-маршрутизатора. Создать такой маршрут можно командой:

(config — rip)# route <ip/m>

49.2.6 Фильтрация анонсов

49.2.6.1 Списки доступа (distribute-list)

Фильтрация при помощи distribute-list использует списки доступа маршрутизатора и префиксные списки. Включить такую фильтрацию можно командой:

(config—rip)# distribute-list <racl_name>|(prefix <prlist_name>) in|out [\<iface\>]

<racl_name> - имя списка доступа маршрутизатора,

prefix <prlist_name> - имя префиксного списка.

distribute-list in - фильтрует входящие анонсы, distribute-list out - исходящие.

<iface> - название интерфейса к которому применяется список.

49.2.6.2 Карты маршрутов (route-map)

Фильтрацию при помощи карты маршрутов можно включить командой:

(config—rip)# route—map <rmap_name> in|out <iface>

<rmap_name> - имя карты маршрутов.

in|out - применяет карту к входящим либо исходящим путям.

<iface> - название интерфейса к которому применяется карта маршрутов.

49.3 RIPNG (RIP для IPv6 сети)

49.3.1 Описание протокола

Протокол обеспечивает поддержку IPv6 адресов.

Чтобы активировать службу и войти в режим настроек RIPNG, нужно ввести команду конфигурации:

(config)# router ripng
(config—ripng)#

Также существуют настройки RIPNG, относящиеся к сетевым интерфейсам. Для редактирования таких настроек необходимо войти в режим конфигурации конкретного интерфейса и ввести необходимые опции с префиксом «ip6 ripng». Например:

```
(config)# interface ethernet 0
(config — if—ethernet0)# ip6 ripng опция параметры ...
```

Процесс настройки сильно похож на RIP, только ip адрес меняется на адрес ipv6, и используется соответствующий формат адресов.

49.3.2 Доступные команды

Для более детального описания команд необходимо обратиться к разделу RIP данного руководства, а также к документации Cisco и FRRouting.

default-information originate	Сообщить соседу маршрут по умолчанию
default-metric <n>	Назначить метрику импортированным маршрутам из других протоколов маршрутизации
network <ip/m>|<iface>	Включение сети или интерфейса в работу RIPNG маршрутизатора
offset-list <racl_name> in|out <metric> [<iface>]	Настройка увеличения метрики
passive-interface	Запрет рассылки LSA с определённых интерфейсах
redistribute [metric <n>] [route-map <rmap_name>]	Импортировать маршруты из других протоколов маршрутизации
route <ip/m>	Создать статический маршрут для анонса
timers basic <update> <timeout> <garb_collect>	Настройка таймеров протокола

Конадны для диагностики маршрутизатора имеют вид:

```
# show ripng [params]
```

(Команды «show ripng ...» и «show router ripng ...» являются синонимами).

49.4 OSPF

В данном разделе приводится краткая информация о настройке протокола динамической маршрутизации OSPF на Dionis-NX. Перед прочтением раздела администратору настоятельно рекомендуется изучить соответствующую подробную литературу о протоколе OSPF (в частности, документацию от компании Cisco).

49.4.1 Основные понятия

OSPF (Open Shortest Path First) - протокол динамической маршрутизации внутри автономной системы.

Автономная система - группа сетей и маршрутизаторов, управляющаяся одним администратором (или группой администраторов, способных договориться между собой).

Протокол OSPF основан на алгоритме Дейкстры - алгоритме нахождения кратчайшего пути. Маршрутизаторы обмениваются информацией о *состоянии каналов (link-state advertisements - LSA).*

В OSPF вводится понятие *области (area). Область* - это набор маршрутизаторов, имеющих одинаковый идентификатор области (число).

Все маршрутизаторы OSPF должны принадлежать хотя бы одной области. Автономная система должна состоять хотя бы из одной области - области 0 (ноль).

Метрика (metric) - численный показатель «стоимости» пересылки данных по каналу. Чем больше - тем хуже канал.

Стоимость маршрута (cost) - сумма метрик каналов, через которые проходит маршрут.

Административное расстояние - численный показатель, определяющий «достоверность» информации о маршруте. Чем меньше - тем достоверней. Выбирается наиболее «достоверный» маршрут. Административное расстояние имеет приоритет над стоимостью маршрута.

Идентификатор маршрутизатора (routerID - RID) - 32-битовое число, которое уникально идентифицирует маршрутизатор в пределах одной автономной системы.

Суммирование (обобщение) маршрутов - объединение адресов нескольких подсетей в один с целью уменьшения количества анонсируемых маршрутов. Например, внутриобластные сети 192.168.32.0/24 и 192.168.33.0/24 можно объединить в 192.168.32.0/23 для анонсирования в другие области.

Импорт (redistribution) маршрутов - анонсирование в среде OSPF маршрутов, полученных из других протоколов маршрутизации.

Виртуальный канал (virtual link) - механизм OSPF, позволяющий связать удалённую область с опорной через другую область. Виртуальный канал не может пролегать через тупиковые области.

Типы областей:

Область 0 (backbone area - опорная область) - область, с которой должны быть соединены все остальные области автономной системы - либо через общий маршрутизатор (ABR), либо через виртуальный канал.

Стандартная область - область, которая может граничить как с другими областями, так и с другими автономными системами.

Для уменьшения объёма таблиц маршрутизации рекомендуется использовать тупиковые области:

Стандартная тупиковая область (stub area) - область, граничащая только с другими областями (желательно с одной). Стандартная тупиковая область не может граничить с другой автономной системой. Если необходимо передать пакет в другую автономную систему - используется маршрут по умолчанию (лежащий через граничную область). Стандартная тупиковая область может принимать маршруты от других областей.

Полностью тупиковая область (totally stubby area) - область, граничащая только с одной областью. При необходимости передачи пакета в другую область или автономную систему используется маршрут по умолчанию. Маршрутизаторы полностью тупиковой области содержат только внутриобластные маршруты.

Не полностью тупиковая область (not-so-stubby area - NSSA) - стандартная тупиковая область с возможностью введения граничного маршрутизатора другой системы динамической маршрутизации (например, RIP). В данной области разрешаются анонсы 1_БАтипа 7.

Полностью тупиковая область NSSA (NSSA no-summary) - аналогична полностью тупиковой области, но разрешены LSA типа 7.

Типы LSA, использующиеся в областях разных типов:

Тип LSA	Описание анонса	Стд. обл.	Стд. тупик.	Поли. тупик.	NSSA	NSSA no-summary
1	Внутриобластные маршруты через данный маршрутизатор	Да	Да	Да	Да	Да
2	Внутриобластные маршруты через DR	Да	Да	Да	Да	Да
3	Суммарные межобластные маршруты через ABR	Да	Да	Нет	Да	Нет
4	Суммарные маршруты через ASBR	Да	Да	Нет	Да	Нет
5	Маршруты через ASBR	Да	Нет	Нет	Нет	Нет
7	Маршруты NSSA через ABR	Нет	Нет	Нет	Да	Да

Типы маршрутизаторов:

Соседние маршрутизаторы - маршрутизаторы OSPF, находящиеся в одной сети.

Назначенный маршрутизатор (designated router - DR) - маршрутизатор, выбирающийся главным относительно остальных соседних маршрутизаторов. Все остальные соседние маршрутизаторы устанавливают с ним отношение *смежности (adjacency).* Маршрутизатор DR принимает анонсы маршрутов от соседей и осуществляет рассылку другим соседям. DR вводится для уменьшения трафика лавинной рассылки анонсов OSPF.

Резервный назначенный маршрутизатор (backup designated router - BDR) **- маршрутизатор.**

берущий на себя функции DR в случае отказа основного DR.

Межобластной граничный маршрутизатор (area border router - ABR) - маршрутизатор, соединяющий две (или более) областей OSPF одной автономной системы.

Гоаничный маршрутизатор автономной системы (autonomous system boundary router - ASBR) **- маршрутизатор, граничащий с другой автономной системы.**

Рекомендуемое количество устройств/маршрутизаторов/областей в автономной системе:

(На основе RFC 2329)

	Рекоменд.	Макс.
Количество соседних устройств на 1 маршрутизатор	50	100
Количество маршрутизаторов в области	< 150	350
Количество областей в автономной системе	< 25	60

Типы маршрутов:

Внутриобластные маршруты - маршруты к сетям, находящимся в пределах области. Стоимость маршрута = сумма метрик каналов.

Межобластные маршруты - маршруты к сетям, находящимся за пределами области. Стоимость маршрута = сумма метрик каналов.

Внешние маршруты E1 - маршруты к сетям, находящимся за пределами автономной системы OSPF. Стоимость маршрута = сумма метрик внутренних каналов + метрика внешнего маршрута.

Внешние маршруты E2 - маршруты к сетям, находящимся за пределами автономной системы OSPF. Стоимость маршрута = метрика внешнего маршрута. Маршрут по умолчанию в тупиковой области имеет класс E2.

49.4.2 Базовая настройка

Активация и настройка службы OSPF на узле Dionis-NX

Чтобы активировать службу и войти в режим настроек OSPF, нужно ввести команду конфигурации:

```
(config)# router ospf
(config—ospf)#
```

В режиме «config-ospf» вводятся общие настройки OSPF для данного узла Dionis-NX. Также существуют настройки OSPF, относящиеся к сетевым интерфейсам. Для редактирования таких настроек необходимо войти в режим конфигурации конкретного интерфейса и ввести необходимые опции с префиксом «ip ospf». Например:

```
(config)# interface ethernet 0
(config —if—ethernet0)# ip ospf опция параметры ...
```

Для отмены опций необходимо ввести соответствующую команду с префиксом «по».

Для останова службы OSPF и удаления всех настроек можно использовать команду:

```
(config)# no router ospf
```

Активация OSPF на интерфейсах, объявление сетей и областей

Чтобы узел начал выполнять функции маршрутизатора OSPF, необходимо объявить:

- Интерфейсы, участвующие в OSPF-маршрутизации;
- IP-адреса интерфейсов;
- Области OSPF, к которым подключены интерфейсы;
- Принадлежность интерфейсов к областям.

Все эти функции выполняет команда «network» в режиме «config-ospf»:

```
(config—ospf)# network <iface_ip>/<mask> area <area_id>
```

Команда «network» осуществляет «привязку» сетевого(ых) интерфейса(ов) данного узла к области с номером <area_id>. Все интерфейсы, IP-адреса которых попадают в диапазон <iface_ip>/<mask>, «привязываются» к данной области. На данных интерфейсах начинается OSPF-маршрутизация, и их IP-сети анонсируются соседним маршрутизаторам.

Если не указывать специальных команд «area», то области, объявленные командой «network», считаются стандартными (не тупиковыми).

Следует помнить, что для корректной работы протокола OSPF в конкретной сети, необходима поддержка многоадресных (multicast) рассылок в данной сети, чтобы маршрутизатор мог обнаруживать соседние маршрутизаторы. Если данная сеть не поддерживает multicast, то необходимо явно указать соседние маршрутизаторы с помощью команды «neighbor» (см. ниже).

Для корректной работы OSPF необходимы согласованные настройки на соседних маршрутизаторах (совпадение идентификаторов и типов областей для соответствующих интерфейсов).

Пример минимальной настройки маршрутизатора OSPF

Допустим, необходимо настроить межобластной (ABR) маршрутизатор с 3-мя сетевыми интерфейсами:

- Интерфейс 0. Подключён к опорной области. Сеть 192.168.1.0/24;
- Интерфейс 1. Подключён к области 1. Сеть 192.168.32.0/24;
- Интерфейс 2. Подключён к области 1. Сеть 192.168.33.0/24.

Минимально необходимая настройка:

```
interface ethernet 0
ip address 192.168.1.1/24
interface ethernet 1
ip address 192.168.32.1/24
```

```
 interface ethernet 2
ip address 192.168.33.1/24
router ospf
  network 192.168.1.0/24 area 0
network 192.168.32.0/23 area 0.0.0.1
```

Вторая команда «network» подключает к области 1 сразу 2 интерфейса - 1 и 2. Идентификатор области может задаваться как в виде числа, так и в четырехбайтном десятичном представлении A.B.C.D.

Явное указание соседей

Если локальная сеть не поддерживает multicast, то необходимо явно указать IP-адреса соседних маршрутизаторов с помощью команды «neighbor»:

```
(config—ospf)# neighbor <ip> [poll—interval <secs>] [priority <n>]
```

Необязательные параметры:

- poll-interval - определяет интервал, с которым будут посылаться hello-пакеты соседу, даже когда он признан «умершим». В соответствии с RFC1247 рекомендуется устанавливать это время гораздо большим, чем hello-интервал. (См. «Таймеры» ниже);
- priority - принудительная установка приоритета соседнего маршрутизатора. Приоритет влияет на выбор DR. (см. «Приоритет маршрутизатора» ниже). По умолчанию - 0 (сосед не участвует в выборах на DR).

Явное указание типа сети

В OSPF различаются следующие типы локальных сетей:

- broadcast - соединение «все-со-всеми» с возможностью multicast рассылок. Выбираются DR и BDR;
- non-broadcast - соединение «все-со-всеми» без возможности multicast рассылок (Non-Broadcast Multi-Access - NBMA). DR и BDR выбираются только на основе приоритетов;
- point-to-multipoint - соединение «один-с-остальными». Multicast не возможен. DR и BDR не выбираются;
- point-to-point - соединение «точка-точка». DR и BDR не выбираются.

В зависимости от типа физического интерфейса OSPF задаёт для него соответствующий тип сети по умолчанию. Например, для интерфейсов Ethernet устанавливается тип broadcast. Если существует необходимость изменить тип сети по умолчанию (например, если Ethernet-интерфейс не поддерживает multicast), то это можно сделать в режиме конфигурации интерфейса командой:

```
(config — if— ethernetO)# ip ospf network <тип>
```

49.4.3 ID и приоритет маршрутизатора

В автономной системе OSPF каждый маршрутизатор должен иметь свой уникальный 32-битный номер. (В частности ID маршрутизатора используется при создании виртуальных каналов). По

умолчанию маршрутизатору присваивается ID, численно равный наибольшему IP-адресу сетевых интерфейсов. Если необходимо вручную установить ID, то это можно сделать с помощью команды режима конфигурации OSPF:

(config—ospf)# router—id <A.B.C.D>

Для каждого интерфейса маршрутизатора OSPF определено понятие приоритета. Приоритет маршрутизатора - это численное значение от 0 до 255. Приоритет играет роль при выборе маршрутизаторов на роль DR и BDR (в рамках одной локальной сети). Чем выше приоритет, тем выше вероятность назначения данного маршрутизатора в качестве DR/BDR. По умолчанию, каждый интерфейс маршрутизатора имеет приоритет 1. Если требуется исключить маршрутизатор из выборов DR/BDR, то необходимо назначить ему приоритет 0.

Приоритет назначается с помощью команды конфигурации интерфейса:

(config—if—ethernetO)# ip ospf priority <n>

49.4.4 Настройка тупиковых областей

По умолчанию, объявленная область считается стандартной.

Чтобы объявить область, как стандартную тупиковую, нужно ввести опцию:

(config—ospf)# area <area_id> stub

Чтобы объявить область, как полностью тупиковую, нужно ввести опцию:

(config—ospf)# area <area_id> stub no—summary

Чтобы объявить область, как стандартную NSSA, нужно ввести опцию:

(config—ospf)# area <area_id> nssa [<translate_mode>]

Чтобы объявить область, как полностью тупиковую NSSA, нужно ввести опцию:

(config—ospf)# area <area_id> nssa [<translate_mode>] no—summary

Для NSSA-областей существует необязательная настройка «translate», которая играет роль только для межобластных (ABR) маршрутизаторов тупиковых областей, и только тогда, когда их несколько. Опция влияет на то, какой именно ABR будет транслировать LSA типа 7 в LSA типа 5 при выходе из NSSA-области. Опция может принимать значения: [13]

13 translate-candidate - значение по умолчанию. Транслирующий ABR выбирается автоматически;
- translate-always - Данный ABR всегда будет являться транслятором;
- translate-never - Данный ABR никогда не будет являться транслятором.

49.4.5 Маршрут по умолчанию

В тупиковых (stub) и полностью тупиковых (stub no-summary) областях граничным маршрутизатором ABR автоматически распространяется маршрут по умолчанию (0.0.0.0) внутрь тупиковой области, указывающий на ABR. В областях NSSA и стандартных областях иногда требуется принудительно распространить маршрут по умолчанию (например, ведущий в другую автономную систему). Это делается на маршрутизаторе ASBR командой режима конфигурации OSPF:

(config—ospf)# default—information originate [always] [metric <n>] [metric—type 1|2]
[route—map <rmap_name>]

Необязательные параметры:

- always - всегда распространять маршрут 0.0.0.0, даже если он не определён на самом маршрутизаторе ASBR;
- metric <n> - установить значение метрики для маршрута по умолчанию;
- metric-type 1|2 - тип маршрута - E1 или E2. (По умолчанию - E2);
- route-map <name> - распространять маршрут 0.0.0.0 только в том случае, если он удовлетворяет указанной схеме (см. «Схемы маршрутов»).

49.4.6 Фильтрация маршрутов

В OSPF существует возможность фильтровать межобластные маршруты (LSA типа 3), если они по каким-то причинам не требуются в данной зоне. Фильтрация осуществляется на межобластных граничных маршрутизаторах (ABR) с помощью следующих команд (в режиме конфигурации OSPF).

Фильтрация межобластных маршрутов, анонсируемых **в** данную область, с помощью списка router ACL:

(config—ospf)# area <id_o6nac™> import—list <имя_или_номер_го^ег_AC1>

Фильтрация межобластных маршрутов, анонсируемых **в** данную область, с помощью префиксного списка:

(config—ospf)# area <id_o6nac™> filter—list prefix <имя_префиксного_списка> in

Фильтрация межобластных маршрутов, анонсируемых **из** данной области, с помощью списка router ACL:

(config—ospf)# area <id_o6nac™> export—list <имя_или_номер_го^ег_A^>

Фильтрация межобластных маршрутов, анонсируемых **из** данной области, с помощью префиксного списка:

(config—ospf)# area <id_o6nac™> filter—list prefix <имя_префиксного_списка> out

О префиксных списках и списках router ACL см. раздел «Списки».

Примеры:

В следующем примере из области 10 в опорную область будут анонсированы маршруты, попадающие в диапазон от 10.10.0.0 до 10.10.255.255. Другие маршруты (например, 10.11.0.0) анонсированы не будут.

```
router access—list foo permit 10.10.0.0/16
router access—list foo deny any
router ospf
  network 192.168.1.0/24 area 0.0.0.0
network 10.0.0.0/8 area 0.0.0.10
area 0.0.0.10 export—list foo
```

Следующий пример аналогичен предыдущему, но реализован с помощью префиксного списка.

```
router prefix—list foo2 permit 10.10.0.0/16
router prefix—list foo2 deny any
router ospf
  network 192.168.1.0/24 area 0.0.0.0
network 10.0.0.0/8 area 0.0.0.10
area 0.0.0.10 filter—list prefix foo2 out
```

49.4.7 Обобщение маршрутов

Для уменьшения таблиц маршрутизации необходимо по возможности «обобщать» маршруты, анонсируемые в другие области. Например, если область содержит подсети 10.1.0.0/24, 10.1.1.0/24, 10.1.2.0/24, 10.1.3.0/24, то на межобластном маршрутизаторе ABR можно обобщить маршруты к данным сетям в один маршрут 10.1.0.0/22.

Обобщение выполняется на ABR, и применяется к LSA типа 1 и 2 (транслируются в LSA типа 3). Обобщение для типов LSA 5 и 7 не поддерживается.

Для обобщения маршрутов необходимо задать явную команду (в режиме конфигурации OSPF):

(config—ospf)# area <иб_области> range <ip/m> [<параметры:»]

где <ip/m> - обобщённая подсеть (из примера выше - 10.1.0.0/22).

Необязательные параметры:

- not-advertise - вместо анонсирования обобщённого маршрута в LSA типа 3 (поведение по умолчанию), маршруты, попадающие в указанную подсеть, анонсироваться во внешнюю область не будут;
- cost <п> - назначить стоимость обобщённого маршрута;
- substitute <ip2/m> - анонсировать префикс <ip2/m> вместо <ip/m>.

49.4.8 Импорт маршрутов

Чтобы импортировать маршруты из других протоколов маршрутизации в OSPF, необходимо указать опцию(и) «redistribute» (в режиме конфигурации OSPF). Формат опции:

(config—ospf)# redistribute <тип_маршрута> [metric <n>] [metric—type 1|2] [route—map <rmap_name>]

Для каждого типа маршрута можно указать свою опцию «redistribute».

Типы маршрутов:

- connected - маршруты, появляющиеся автоматически при назначении IP-адресов сетевым интерфейсам;
- static - принудительно назначенные статические маршруты;
- kernel - маршруты, загруженные в ядро Linux, минуя систему конфигурации Dionis-NX (на данный момент таких нет);
- bgp, rip - маршруты, создаваемые соответствующими службами динамической маршрутизации.

Параметры:

- metric <n> - назначить данным импортируемым маршрутам метрику;
- metric-type 1|2 - тип импортируемых маршрутов (E1 или E2);
- route-map <name> - применить схему маршрута к импортируемым маршрутам (для фильтрации и установки параметров). См. «Схемы маршрутов».

Также импортируемые маршруты можно отфильтровать на основе списка router ACL (см. «Списки») с помощью команды:

(config—ospf)# distribute—list <имя_или_номер_списка_го^ег_ас1> out <тип_маршрута>

49.4.9 Пассивный интерфейс

Иногда возникает необходимость запретить рассылку LSA с определённых интерфейсов. Для этого надо объявить сетевой интерфейс пассивным с помощью команды режима конфигурации OSPF:

(config—ospf)# passive—interface default| <интерфейс>

Хотя пассивный интерфейс не рассылает анонсы LSA, он всё равно может принимать анонсы от других маршрутизаторов.

Если указать параметр «default», то все интерфейсы становятся пассивными. Также можно выборочно «активизировать» несколько интерфейсов, оставив остальные пассивными. Например, допустим есть интерфейсы ethernet 0, 1, 2, 3.

```
router ospf passive—
  interface default
no passive—interface ethernet 2
```

В данной конфигурации интерфейс 2 будет активным, а 0, 1, 3 - пассивными.

49.4.10 Метрики, стоимость, административное расстояние

Метрики интерфейсов

По умолчанию, всем интерфейсам присваивается метрика, соответствующая пропускной способности интерфейса. Чем выше пропускная способность, тем меньше метрика. Метрика 1 присваивается всем интерфейсам, чья пропускная способность >= 100 Мбит/с. Если в системе имеются более быстродействующие интерфейсы, то можно изменить формулу вычисления метрик от пропускной способности с помощью команды «auto-cost reference-bandwidth». Например:

(config—ospf)# auto—cost reference—bandwidth 1000

Данная опция указывает, что метрика 1 будет присваиваться интерфейсам с пропускной способностью >= 1000 Мбит/с. Соответственно интерфейсам с пропускной способностью 100 Мбит/с будет присвоена метрика 10.

Также можно указать явную метрику для интерфейса (в режиме конфигурации интерфейса):

(config—if—ethernetO)# ip ospf cost <метрика> [<ip>]

Параметр <ip> имеет значение, если интерфейсу назначено несколько IP-адресов.

Метрики импортированных маршрутов

Чтобы назначить метрику импортированным (redistributed) маршрутам из других протоколов маршрутизации, нужно указать опцию (в режиме конфигурации OSPF):

(config—ospf)# default—metric <n>

Стоимость маршрутов в тупиковой области

Чтобы задать стоимость суммарных маршрутов, импортируемых в тупиковую или NSSA-область, нужно указать опцию (в режиме конфигурации OSPF):

(config—ospf)# area <id_o6nac™> default—cost <n>

Административное расстояние

По умолчанию, административное расстояние для всех маршрутов OSPF равно 110. Если необходимо изменить это значение, то это можно сделать с помощью команды режима конфигурации OSPF:

(config—ospf)# distance <n>

Если требуются разные значения административного расстояния для маршрутов разных типов, то следует использовать команду:

(config—ospf)# distance ospf <[external <n>] [inter—area <n>] [intra—area <n>]>

1 external <n> - AP для маршрутов за пределы автономной системы OSPF;

- inter-area <n> - AP для межобластных маршрутов;
- intra-area <n> - AP для внутреобластных маршрутов.

49.4.11 Виртуальные каналы

В автономной системе OSPF требуется, чтобы каждая область была подключена к опорной области 0. Если нет возможности подключить область к области 0 непосредственно через ABR, но область (1) граничит с другой областью (2), подключённой к опорной (0), то можно создать между областью (0) и областью (1) *виртуальный канал* через область (2).

Чтобы настроить виртуальный канал между двумя ABR, надо на обоих маршрутизаторах прописать опцию (в режиме конфигурации OSPF):

(config—ospf)# area <id1_области> virtual—link <id1_маршрутизатора>
[<таймеры_ospf_для_канала>]

- id_области - номер области, через которую будет пролегать виртуальный канал;
- id_маршрутизатора - идентификатор противоположного маршрутизатора;
- таймеры_ospf - интервалы hello, dead, transmit, retransmit (см. «Таймеры»).

Также поддерживается режим shortcut для ABR-маршрутизаторов. См. draft-ietf-shortcut-abr-2. Следующая команда управляет режимом shortcut. (config—

ospf)# area <id_области> shortcut default|disable|enable

Для режима shortcut также необходимо указать опцию типа маршрутизатора:

(config—ospf)# ospf abr—type shortcut

49.4.12 Защита

В OSPF реализована возможность аутентификации маршрутизаторов между собой с целью исключения возможности подмены маршрутизаторов и навязывания ложных маршрутов.

Чтобы задать режим аутентификации для интерфейсов, подключённых к области, нужно указать опцию (в режиме конфигурации OSPF):

(config—ospf)# area <id_область> authentication [message—digest]

Опция «message-digest» предписывает использование алгоритма MD5. Если не указать «message-digest», то пароли будут передаваться в открытом виде.

Чтобы задать режим аутентификации для конкретного интерфейса, нужно указать опцию (в режиме конфигурации интерфейса):

(config—if— ethernet0)# ip ospf authentication [message—digest|null] [<ip>]

Опция <ip> имеет смысл, если интерфейсу назначено несколько IP-адресов.

Чтобы задать пароль для аутентификации (если не используется алгоритм MD5), следует указать опцию для интерфейса:

(config—if—ethernet0)# ip ospf authentication—key <пароль> [<ip>]

Пароли должны совпадать на всех соседних маршрутизаторах.

Чтобы задать пароль при использовании алгоритма MD5, нужно указать опцию интерфейса:

(config—if—ethernet0)# ip ospf message—digest—key <номер_пароля> md5 <пароль> [<ip>]

На всех соседних маршрутизаторах пары (номер, пароль) должны совпадать. Можно ввести несколько паролей. Это обычно делается при плановой замене ключей, чтобы каналы OSPF не прерывались.

Пример смены ключей. Допустим на узлах установлены пароли с номером 1:

Hostl (config)# interface ethernet 0

Hostl (config —if—ethernet0)# ip ospf message—digest—key 2 md5 NOVYPAROL

Host2 (config)# interface ethernet 0

Host2 (config —if—ethernet0)# ip ospf message—digest—key 2 md5 NOVYPAROL
Host2 (config—if—ethernetO)# no ip ospf message—digest—key 1

Hostl (config—if—ethernetO)# no ip ospf message—digest—key 1

Защита виртуального канала

На концах виртуального канала также можно установить взаимную аутентификацию с помощью опций (в режиме конфигурации OSPF):

area <id_область> virtual—link <id_маршрутизатора> authentication message—digest|null
area <id_область> virtual—link <id_маршрутизатора> authentication—key <пароль>
area <id_область> virtual—link <id_маршрутизатора> message—digest—key <номер_пароля>
md5 <пароль>

49.4.13 Таймеры

Для каждого сетевого интерфейса можно настроить следующие временные параметры протокола OSPF:

- hello-interval - интервал посылки hello-пакета соседним маршрутизатором. (По умолчанию - Ю с);
- retransmit-interval - время, по истечению которого маршрутизатор повторно отправит запрос соседу, если он не получил подтверждение. (По умолчанию - 5 с);
- dead-interval - время, по истечению которого сосед считается «умершим», если от него не было hello-пакетов в течение этого времени. (По умолчанию - 40 с);
- transmit-delay - время, добавляемое на пересылку анонса соседу (для медленных сетей). (По умолчанию - 1 с).

Если требуется изменить значения по умолчанию для таймеров, то это можно сделать следующей командой в режиме конфигурации соответствующего интерфейса:

(config —if—ethernet0)# ip ospf <тип_таймера> <секунды> [<ip>]

Опция <ip> имеет значение, когда интерфейсу назначено несколько IP-адресов.

Чтобы вернуть значение по умолчанию, нужно выполнить команду:

(config—if— ethernetO)# no ip ospf <тип_таймера> [<ip>]

Интервал рассылки LSA

(config—ospf)# refresh tinner <секунды>

Значение по умолчанию - 10 с.

SPF throttling

В системах с часто меняющейся топологией иногда необходимо регламентировать частоту вычислений маршрутов по алгоритму SPF. Это можно настроить с помощью следующей команды режима конфигурации OSPF:

(config—ospf)# timers throttle spf <init_delay> <init_hold> <max_hold>

- init_delay - начальное время задержки вычисления SPF (в мс) после получения LSA;
- init_hold - последующая задержка вычисления SPF после получения LSA (в мс). Удваивается каждый раз (до max_hold);
- max_hold - максимальное значение последующей задержки (в мс).

Если обновления топологии приходят часто, то вычисление SPF будет регламентировано следующим образом:

init_delay, init_hold, 2*init_hold, 4*init_hold, ..., max_hold, max_hold, ...

49.4.14 Тонкие настройки OSPF

Включение/выключение поддержки Opaque LSA (RFC 2370)

(config—ospf)# [no] capability opaque

Синоним:

(config—ospf)# [no] ospf opaque—lsa

Включение/выключение совместимости с RFC 1583

(config—ospf)# [no] compatible rfcl583

Синоним:

(config—ospf)# [no] ospf rfcl583compatibility

Тупиковый маршрутизатор (RFC 3137)

Если объявить маршрутизатор, как тупиковый, то он будет анонсировать маршруты к себе с бесконечной метрикой, и другие маршрутизаторы будут стараться не прокладывать маршруты через него.

Объявление тупикового маршрутизатора:

(config—ospf)# max—metric router—lsa <режим>

Возможные режимы:

- administrative - объявить маршрутизатор тупиковым немедленно и на неопределённое время (до отмены);
- on-startup <секунды> - объявлять тупиковым после старта системы на заданное время;
- on-shutdown <секунды> - после выполнения команды «по router ospf» не сразу останавливать службу OSPF, но объявлять маршрутизатор тупиковым на заданное время, а потом останавливать.

Команда «по» отменяет режим тупикового маршрутизатора:

(config—ospf)# no max—metric router—lsa <режим>

Типы маршрутизаторов ABR (RFC 3509)

Опция «ospf abr-type» настраивает поведение ABR маршрутизатора согласно RFC 3509 и draft - ietf-ospf-shortcut-abr-02.

(config—ospf)# ospf abr-type cisco|ibm|shortcut|standard

Игнорирование несовпадения MTU

По умолчанию в OSPF включена проверка совпадения MTU между соседними маршрутизаторами. В случае несовпадения не будет установлено отношение смежности. Если требуется отключить данную проверку, то это можно сделать опцией в режиме конфигурации интерфейса:

(config — if—ethernetO)# ip ospf mtu — ignore [<ip>]

Параметр <ip> имеет смысл, если интерфейсу назначено несколько IP-адресов.

49.4.15 Диагностика

Следующие команды привилегированного режима выводят различную диагностическую информацию о OSPF.

(Команды «show ospf ...» и «show router ospf ...» являются синонимами).

show log router [all|follow|number <n>| search <REGEX>]	Вывод журнала служб динамической маршрутизации
show ip route	Вывод всей таблицы маршрутизации узла (всех протоколов)
show ospf	Вывод краткой информации о состоянии службы OSPF
show ospf route	Вывод таблицы маршрутизации для OSPF
show ospf border-routers	Вывод только внешних и межобластных маршрутов
show ospf interface <iface>	Информация об интерфейсе (с точки зрения OSPF)
show ospf neighbor [all | <ip> | (<iface> [detail]) | detail]	Информация о соседнем(их) маршрутизаторе(ах)

Команды вывода базы данных OSPF:

show ospf database	Краткая информация о базе данных OSPF
show ospf database router	Информация о LSA type 1
show ospf database network	Информация о LSA type 2
show ospf database summary	Информация о LSA type 3
show ospf database asbr-summary	Информация о LSA type 4
show ospf database external	Информация о LSA type 5
show ospf database nssa-external	Информация о LSA type 7
show ospf database opaque-link	Информация о LSA type 9
show ospf database opaque-area	Информация о LSA type 10
show ospf database opaque-as	Информация о LSA type 11
show ospf database max-age	Информация о LSA, находящихся в списке MaxAge

Почти ко всем командам «show ospf database» применимы дополнительные параметры:

- self-originate - показать только те данные, источником которых является данный маршрутизатор;
- adv-router <ip> - показать только те данные, источником которых является указанный маршрутизатор.

Если необходима более подробная информация об изменении отношений смежности с соседними маршрутизаторами, то нужно указать опцию в режиме конфигурации OSPF:

```
(config—ospf)# log—adjacency—changes [detail]
```

Информация об изменениях отношений смежности будет протоколироваться в журнале служб динамической маршрутизации.

49.5 OSPF6 (OSPFv3 для IPv6 сети)

49.5.1 Описание протокола

Протокол обеспечивает поддержку IPv6 адресов.

Чтобы активировать службу и войти в режим настроек OSPFv3, нужно ввести команду конфигурации:

```
(config)# router ospf6
(config—ospf6)#
```

В режиме «config-ospf6» вводятся общие настройки OSPFv3 для данного узла Dionis-NX. Также существуют настройки OSPFv3, относящиеся к сетевым интерфейсам. Для редактирования таких настроек необходимо войти в режим конфигурации конкретного интерфейса и ввести необходимые опции с префиксом «ip6 ospf6». Например:

(config)# interface ethernet 0
(config —if—ethernet0)# ip6 ospf6 опция параметры ...

Процесс настройки сильно похож на OSPFv2, только ip адрес меняется на адрес ipv6, и используется соответствующий формат адресов. Кроме того необходимо включить OSPFv3 на интерфейсе с помощью команды **iface:**

(config—ospf6)# iface cinterface—type> cinterface—num> area <id_области>

Все IPv6-адреса, назначенные интерфейсу, участвуют в OSPFv3. Поскольку в hello-пакетах в качестве отправителя указывается локальный адрес интерфейса (link-local address), соседская связь будет сформирована, даже если сосед сконфигурирован с другим префиксом.

49.5.2 Доступные команды

Для более детального описания команд необходимо обратиться к разделу OSPF данного руководства, а также к документации FRRouting.

area <1c1_области> [params]	Объявление зоны
auto-cost reference-bandwidth	Настройка вычисления метрик в зависимости от пропускной способности канала
log-adjacency-changes [detail]	Более подробное протоколирование информации об изменениях отношений смежности в журнале служб динамической маршрутизации.
router-id <A.B.C.D>	ID маршрутизатора
redistribute <тип_маршрута> [route-map]	Импорт маршруов из других протоколов маршрутизации в OSPFv3
timers throttle spf <init___delay> <init_hold> <max_hold>	Частота вычислений маршрутов по алгоритму SPF
stub-router administrarive	Отключение режима транзитного маршрутизатора

Конадны для диагностики маршрутизатора имеют вид:

show ospf6 [params]

(Команды «show ospf6 ...» и «show router ospf6 ...» являются синонимами).

49.6 BGP

49.6.1 Описание протокола BGP

BGP обеспечивает маршрутизацию без петель между автономными системами (AS), RFC 4271 «A Border Gateway Protocol 4 (BGP-4)». Маршрутизаторы используют протоколы внутренней маршрутизации (IGP) внутри AS, а вне AS - протокол BGP.

Когда BGP работает между маршрутизаторами в одной AS, это называется внутренний BGP (iBGP). Когда BGP работает между маршрутизаторами, которые принадлежат к разным AS, это называется внешний BGP (eBGP). BGP использует TCP в качестве транспорта (порт 179). Два BGP-маршрутизатора устанавливают TCP-соединения между собой. Такие маршрутизаторы назы- ваются соседними маршрутизаторами (соседями).

Соседи обмениваются информацией о путях (BGP-анонсами). BGP-путь, это набор номеров AS, которые следует пройти к сети назначения.

BGP-пути хранятся в трех BGP-таблицах: Adj-RIB-In (Adjacent Routing Information Base, Incoming), Loc-RIB (Local Routing Information Base) и Adj-RIB-Out (Adjacent Routing Information Base, Outgoing). Получив анонс от соседа, BGP помещает пути из него в таблицу Adj-RIB-In, затем обрабатывает их в соответствии с политиками и перемещает в таблицу Loc-RIB. Также в Loc-RIB хранятся локальные пути, которые настроены администратором, и пути, которые перераспределены из маршрутов таблицы IP-маршрутизации. Пути, предназначенные для анонса соседям, BGP помещает из таблицы Loc-RIB в таблицу Adj-RIB-Out.

BGP выбирает лучший путь из Loc-RIB, сравнивает административную дистанцию (AD), между остальными путями в ту же сеть, но полученными из других протоколов, например OSPF, RIP. Путь с наименьшей AD помещаются в таблицу 1P-маршрутизатора.

49.6.2 BGP-маршрутизатор

49.6.2.1 Включение BGP-маршрутизатора (router bgp)

Первое, что требуется для начала настройки BGP, это включить BGP-маршрутизатор. Сделать это можно командой:

```
(config)# router bgp <AS>
```

<AS> - номер AS, в которую входит настраиваемый BGP-маршрутизатор.

После ввода команды система переходит в режим конфигурирования BGP, в котором вводятся остальные команды настройки BGP-маршрутизатора.

Пример включения BGP-маршрутизатора, который входит в AS с номером 65001.

```
(config)# router bgp 65001
(config —bgp—65001)#
```

Выключить BGP-маршрутизатор можно командой:

```
(config)# no router bgp <AS>
```

Конфигурация BGP-маршрутизатора при этом удаляется.

Для настройки маршрутизатора доступны несколько семейств адресов. Перейти к конкретному семейству адресов можно командой:

```
(config —bgp—65001)# address—family ipv4|ipv6 unicast|multicast
```

49.6.2.2 Идентификатор BGP-маршрутизатора (router-id)

При включении BGP-маршрутизатора создается идентификатор. Это IP-адрес, который совпадает с максимальным IP-адресом интерфейсов маршрутизатора. Если в маршрутизаторе не существует ни одного интерфейса, то идентификатору BGP-маршрутизатора присваивается значение 0.0.0.0 и его необходимо изменить вручную. Изменить идентификатор можно командой:

(config — bgp—65001)# bgp router—id <RID>

<RID> - IP-адрес, который будет идентифицировать BGP-маршрутизатор. Пример использования IP-адреса 192.168.1.1 в качестве идентификатора.

(config — bgp—65001)# bgp router—id 192.168.1.1

Удалить заданный идентификатор и вернуться к выбору по умолчанию можно командой:

(config — bgp—65001)# no bgp router—id

49.6.3 BGP-соединение

49.6.3.1 Создание BGP-соединения (neighbor)

BGP-соединение устанавливается между двумя соседями. Для установления BGP-соединения у обоих соседей в конфигурации должна быть команда:

(config — bgp—65001)# neighbor <ip>|<group> remote—as <AS>

<ip> - IP-адрес соседнего BGP-маршрутизатора. Для EBGP-соединения, этот адрес должен быть в непосредственно подключенной сети (можно отключить такую проверку командой disable-connected-check). Для IBGP-соединения, к этому адресу должен существовать маршрут.

<group> - имя группы BGP-маршрутизаторов.

<АБ> - номер AS, в которую входит соседний BGP-маршрутизатор или группа.

Пример настройки EBGP-соединения между соседом 1 из AS 65001 с IP-адресом 192.168.1.1 и соседом 2 из AS 65002 с IP-адресом 192.168.1.2:

Сосед— l(config — bgp—65001)# neighbor 192.168.1.2 remote—as 65002

Сосед—2(config —bgp—65002)# neighbor 192.168.1.1 remote—as 65001

Проверить возможность установления соединения можно пингом:

Сосед—1# ping 192.168.1.2 source 192.168.1.1

49.6.3.2 Удаление и административное выключение BGP-соединения (shutdown)

Удалить настройку BGP-соединения можно командой:

(config — bgp—65001)# no neighbor <ip>|<group> remote—as

При этом удаляются все остальные настройки, связанные с этим соседом. Для временной блокировки лучше использовать команду административного выключения:

(config—bgp—65001)# neighbor <ip>|<group> shutdown

Включить соседа обратно можно командой:

(config—bgp—65001)# no neighbor <ip>|<group> shutdown

49.6.3.3 Группы BGP-маршрутизаторов (peer-group)

Несколько BGP-маршрутизаторов можно объединить в единую группу, если они находятся в одной AS. В этом случае они будут использовать общие настройки группы. Создать группу можно командой:

(config—bgp—65001)# neighbor <group> peer—group

<group> - имя группы. Добавить BGP-маршрутизатор в группу можно командой:

(config—bgp—65001)# neighbor <ip> peer—group <group>

<ip> - IP-адрес соседа.

Пример создания группы mybgpgroup из трех BGP-маршрутизаторов, находящихся в AS 65003. Перед добавлением BGP-маршрутизаторов в группу следует сначала определить для группы номер удаленной AS, а затем добавить узлы.

(config—bgp—65001)# neighbor mybgpgroup peer—group (config—
bgp—65001)# neighbor mybgpgroup remote—as 65003 (config—bgp—
65001)# neighbor 192.168.3.1 peer—group mybgpgroup (config—
bgp—65001)# neighbor 192.168.3.2 peer—group mybgpgroup
(config—bgp—65001)# neighbor 192.168.3.3 peer—group mybgpgroup

49.6.3.4 Использование dummy-интерфейса (update-source)

При использовании dummy-интерфейса, BGP-маршрутизатор сообщает соседу IP-адрес, который не принадлежит ни одному физическому интерфейсу и, следовательно, не зависит от его состояния (dummy-интерфейс всегда активен). Для использования dummy-интерфейса нужно выполнить команду:

(config—bgp—65001)# neighbor <ip>|<group> update—source <ip—iface>|<iface>

<ip-iface> - IP-адрес dummy-интерфейса.

<iface> - имя dummy-интерфейса.

Пример использования dummy-интерфейса с именем dummyO и IP-адресом 192.168.1.1 в качестве источника BGP-обновлений.

(config)# interface dummy 0 (config—if—
dummy0)# ip address 192.168.1.1/24 (config—
if—dummy0)# enable
(config—bgp—65001)# neighbor 195.220.1.2 update—source dummy 0

Отменить использование dummy-интерфейса можно командой:

(config — bgp—65001)# no neighbor <ip>|<group> update—source

49.6.3.5 Отмена проверки подключенной сети для EBGP (disable-connected-check, enforce-multihop)

При EBGP-соединении BGP-маршрутизатор ищет соседа в одной из подключенных сетей. Например, при использовании dummy-интерфейса, который никуда не подключен, следует отключать такую проверку, либо использовать команду ebgp-multihop. Отключить проверку можно командой:

(config — bgp—65001)# neighbor <ip>|<group> disable—connected—check

У этой команды существует синоним, команда enforce-multihop, при выполнении которой в конфигурацию все равно запишется disable-connected-check.

49.6.3.6 Настройка TTL для EBGP (ebgp-multihop, ttl-security hops)

Допустим EBGP-соседи не находятся в одной сети. Тогда для установления BGP-соединения следует указать, что для достижения соседа требуется проходить несколько маршрутизаторов. Сделать это можно командой:

(config — bgp—65001)# neighbor <ip>|<group> ebgp-multihop [<hop_count>]

<hop_count> - не обязательный параметр устанавливает TTL в IP-датаграмме.

Пример увеличения TTL для соседа с IP-адресом 195.220.1.2.

(config — bgp—65001)# neighbor 195.220.1.2 ebgp-multihop 128

Отменить изменение TTL можно командой:

(config — bgp—65001)# no neighbor <ip>|<group> ebgp-multihop

Иногда, в целях безопасности, требуется явное указание расстояния до EBGP-соседа. Сделать это можно командой:

(config — bgp—65001)# neighbor <ip>|<group> ttl-security hops <n>

<n> - чисто маршрутизаторов на пути к соседу.

Внимание! Команда применима только для EBGP и при использовании заменяет команду neighbor ebgp-multihop. Запрещено одновременное использование с neighbor ebgp-multihop.

Пример контроля TTL для соседа с IP-адресом 195.220.1.1. IP-датаграммы, у которых TTL меньше 253, будут сброшены.

(config — bgp—65001)# neighbor 192.168.1.1 ttl-security hops 2

Отменить изменение TTL можно командой:

(config — bgp—65001)# no neighbor <ip>|<group> ttl-security hops

49.6.3.7 Контроль линка для EBGP (fast-external-failover)

В BGP-маршрутизаторе существует механизм, который немедленно обрывает BGP-соединение, если на интерфейсе пропало соединение. Этот механизм применяется только для EBGP-соединения и он включен по умолчанию. Если требуется отменить такое поведение, то сделать это можно командой:

(config — bgp—65001)# no bgp fast—external—failover

После отмены вместо механизма обнаружения падения соединения будут использоваться тай-меры hold и keepalive. Таким образом, если пропадет линк, BGP-соединение еще некоторое время будет установлено. Включить механизм обратно можно командой:

(config — bgp—65001)# bgp fast—external—failover

49.6.3.8 Аутентификация (password)

Включить аутентификацию по паролю (MD5) можно командой:

(config — bgp—65001)# neighbor <ip>|<group> password <string>

<string> - пароль для аутентификации соседа Аутентификация настраивается на обоих сосе-дях, устанавливающих BGP-соединение.

Пример настройки аутентификации по паролю lssap!

(config — bgp—65001)# neighbor 192.168.1.2 password lssap!

Отменить аутентификацию можно командой:

(config — bgp—65001)# no neighbor <ip>|<group> password

49.6.3.9 Пассивный режим BGP-соединения (passive)

Пассивный режим позволяет получать анонсы соседа только по его инициативе. Включить пассивный режим можно командой:

(config — bgp—65001)# neighbor <ip>|<group> passive

Отключить пассивный режим можно командой:

(config — bgp—65001)# no neighbor <ip>|<group> passive

49.6.3.10 IPv6 (activate)

Для разрешения обмена с адресами IPv6 перейти в подсекцию семейства адресов и выполнить команду **activate :**

(config — bgp—65001)# address—family ipv4 unicast
(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> activate

Эту команду не следует использовать для IPv4.

49.6.3.11 Журнал (log-neighbor-changes)

Включить протокол изменений состояния соседей можно командой:

(config — bgp—65001)# bgp log —neighbor—changes

Изменения будут записываться в журнал маршрутизатора. Посмотреть журнал можно командой:

show log router

Отменить ведение протокола можно командой:

(config — bgp—65001)# no bgp log —neighbor—changes

49.6.3.12 Изменение TCP-порта службы BGP (port)

BGP-соединение использует подключение к 179-му TCP-порту. Если требуется использовать подключение к другому порту, сделать это можно командой:

(config — bgp—65001)# neighbor <ip> port <n>

<n> - BGP-порт соседа.

Удалить настройку и вернуть значение по умолчанию можно командой:

(config — bgp—65001)# no neighbor <ip> port

49.6.3.13 Описание к настройкам BGP-соседа (description)

Для удобства администрирования можно задать описание для соседа или группы, при помощи команды:

(config — bgp—65001)# neighbor <ip>|<group> description <string>

<string> - Строка с описанием.

Удалить описание можно командой:

(config — bgp—65001)# no neighbor <ip>|<group> description

49.6.4 Таймеры

49.6.4.1 Hold и keepalive

Для поддержания BGP-соединения используется механизм отправки сигналов жизни keepalive. При установлении соединения BGP-маршрутизаторы обмениваются параметром hold, который определяет время ожидания прихода keepalive. Изменить значения по умолчанию можно командой:

(config — bgp—65001)# tinners bgp <keepalive> <hold>

<keepalive> - время отправки пакетов жизни, по умолчанию 60 секунд.

<hold> - время удержания BGP-соединения открытым до получения keepalive, по умолчанию 180 секунд.

Пример изменения таймеров keepalive на 30 секунд, hold на 90 секунд.

(config — bgp—65001)# tinners bgp 30 90

Удалить введенные параметры и вернуться к значениям по умолчанию можно командой:

(config — bgp—65001)#no timers bgp

Эти же параметры можно изменить для определенного соседа командой:

(config — bgp—65001)# neighbor <ip>|<group> timers <keepalive> <hold>

Пример изменения таймеров keepalive на 30 секунд, hold на 150 секунд для соседа с IP-адресом 192.168.1.2.

(config — bgp—65001)# neighbor 192.168.1.2 timers 30 150

Удалить установленные таймеры и вернуться к значениям по умолчанию для конкретного соседа можно командой:

(config — bgp—65001)# no neighbor <ip>|<group> timers

Посмотреть текущие таймеры можно командой:

show bgp neighbors [<ip>|<group>]

49.6.4.2 Интервал анонсирования (advertisement-interval)

BGP-маршрутизатор анонсирует пути с определенным интервалом (по умолчанию 30 секунд). Изменить этот интервал можно командой:

(config — bgp—65001)# neighbor <ip> advertisement—interval <t>

<t> - время в секундах (по умолчанию 30 секунд).

Пример изменения интервала анонсирования на 10 секунд для соседа с IP-адресом 192.168.1.2:

(config — bgp—65001)# neighbor 192.168.1.2 advertisement—interval 10

Посмотреть текущий интервал анонсирования можно командой:

show bgp neighbors [<ip>|<group>]

Удалить настроенные интервалы и вернуться к значению по умолчанию можно командой:

(config — bgp—65001)# no neighbor <ip> advertisement—interval

49.6.4.3 Интервал попыток подключения (timers connect)

BGP-маршрутизатор делает попытки установить BGP-соединение через определенный интервал (по умолчанию 120 секунд). Изменить этот интервал можно командой:

(config — bgp—65001)# neighbor <ip> tinners connect <t>

<t> - время в секундах (по умолчанию 120 секунд).

Пример настройки интервала 600 секунд для соседа с IP-адресом 192.168.1.2:

(config — bgp—65001)# neighbor 192.168.1.2 timers connect 600

Таймер постоянно уменьшается от установленного значения до нуля, достигнув нуля, вновь увеличивается до максимума и начинает уменьшаться. Посмотреть интервал можно (пока не установлено BGP-соединение) командой:

show bgp neighbors [<ip>]

Удалить настройки и вернуться к значению по умолчанию можно командой:

(config — bgp—65001)# no neighbor <ip> timers connect

49.6.5 Анонсирование

Команды анонсирования выполняются из подсекции семейства адресов Для перехода в подсекцию необходимо выполнить команду:

(config — bgp—65001)# address—family ipv4|ipv6 unicast|multicast

49.6.5.1 Анонсирование сети (network)

При первоначальной настройке BGP-маршрутизатор не анонсирует для соседей ничего, кроме своего идентификатора. Анонсировать определенную сеть можно командой:

(config — bgp—65001—af4—unicast)# network <ip/m> [(route—map <rmap_name>)]

<ip/m> - IP-адрес и маска анонсируемой сети.

route-map <rmap_name> - анонсирует сеть с параметрами карты маршрута.

Пример анонсирования сети 10.0.0.0/8.

(config — bgp—65001—af4—unicast)# network 10.0.0.0/8

Удалить анонсированную ранее сеть можно командой:

(config — bgp—65001—af4—unicast)# no network <ip/m>

49.6.5.2 Проверка наличия маршрута в анонсируемую сеть (import-check)

По умолчанию BGP-маршрутизатор не проверяет существование в таблице IP-маршрутизации маршрута к сети, которую он анонсирует соседям командой network. Проверку существования маршрута можно включить командой:

(config — bgp—65001)# bgp network import—check

Отменить эту проверку можно командой:

(config — bgp—65001)# no bgp network import—check

49.6.5.3 Перераспределение (redistribute)

Помимо явного анонсирования сетей при помощи команды network, возможно анонсировать сети путем перераспределения маршрутов из таблицы IP-маршрутизатора в BGP-маршрутизатор. Команда доступна из подсекции семейства адресов и имеет следующий вид:

(config — bgp—65001—af4—unicast)# redistribute kernel|connected|static|rip|ospf [metric <n>]
[route—map <rmap>]

redistribute kernel - анонсирует маршруты, используемые ядром linux.

redistribute connected - анонсирует маршруты интерфейсов, подключенных к коммутатору.

redistribute static - анонсирует статические маршруты, т.е. прописанные вручную администратором.

redistribute rip - анонсирует маршруты, полученные по RIP.

redistribute ospf - анонсирует маршруты, полученные по OSPF.

metric <n> - метрика, с которой будут анонсированы эти маршруты,

route-map <rmap> - анонсирует сеть с параметрами карты маршрута.

Посмотреть какие именно маршруты находятся в таблице IP-маршрутизатора можно командой:

show ip route

Пример перераспределения подключенных маршрутов:

(config — bgp—65001—af4—unicast)# redistribute connected

Удалить перераспределение можно командой:

(config — bgp—65001—af4—unicast)# no redistribute kernel|connected|static|rip|ospf

49.6.5.4 Суммарный путь (aggregate-address)

Суммарный путь для нескольких подсетей анонсируется соседям с целью уменьшения объемов передаваемой информации. Команда доступна из подсекции семейства адресов и имеет следующий вид:

(config — bgp—65001—af4—unicast)# aggregate—address <ip/m> [as—set] [summary—only]

<ip/m> - IP-адрес и маска сети, в которую входят подсети данного маршрутизатора.

as-set - создает новое поле AS_SET для суммарного пути, в которое записывает все AS, через которые проходят компоненты пути. Этот параметр требуется, если у путей в подсети различается поле AS_SEQ, которое при суммировании может приобрести нулевое значение, что может приводить к образованию петель.

summary-only - анонсирует только суммарный путь, пути в подсети не анонсируются.

Пример суммарного пути для сетей 192.168.1.0/25 и 192.168.1.128/25:

```
(config —bgp—65001—af4—unicast)# aggregate—address 192.168.1.0/24
```

В этом случае соседу анонсируется три пути.

```
B  192.168.1.0/24  via  195.220.1.1
B  192.168.1.0/25  via  195.220.1.1
B 192.168.1.128/25 via 195.220.1.1
```

Если требуется анонсировать только суммарный путь, то сделать это можно командой:

```
(config — bgp—65001—af4—unicast)# aggregate—address 192.168.1.0/24 summary—only
```

В этом случае соседу анонсируется только один суммарный путь.

```
B 192.168.1.0/24 via 195.220.1.1
```

Удалить суммарный путь можно командой:

```
(config — bgp—65001—af4—unicast)# no aggregate—address <ip/m>
```

49.6.5.5 Выборочный анонс подавленных путей (unsuppress-map)

Иногда следует разрешить некоторые пути из суммарного пути для определенного соседа. Команда доступна из подсекции семейства адресов и имеет следующий вид:

```
(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> unsuppress—map <rmap_name>
```

<rmap_name> - имя карты маршрутов, содержащей параметры разрешенных путей.

Пример разрешения подсети 192.168.1.128/25 для соседа с IP-адресом 172.16.0.2.

```
(config — bgp—65001—af4—unicast)# aggregate—address 192.168.1.0/24 summary—only
(config —bgp—65001—af4—unicast)# neighbor 172.16.0.2 unsuppress—map myunmap
(config)# router router—map myunmap permit 1
(config —route—map—myunmap)# match ip address myunmapacl
(config)# router accsss—list myunmapacl permit 192.168.1.128/25
```

В этом случае соседу анонсируется только два пути.

```
B 192.168.1.0 via 195.220.1.1
B 192.168.1.128/25 via 195.220.1.1
```

Удалить настройки можно командой:

```
(config — bgp—65001—af4—unicast)# no neighbor <ip>|<group> unsuppress—map <rmap_name>
```

49.6.5.6 Маршрут по умолчанию (default-originate)

Маршрут по умолчанию можно сообщить соседу и при этом его не нужно создавать в таблице IP-маршрутизации. Команда доступна из подсекции семейства адресов и имеет следующий вид:

(config—bgp—65001—af4—unicast)# neighbor <ip>|<group> default-originate [route—map <rmap_name>]

Пример анонсирования соседом 1 с IP-адреса 192.168.1.1 маршрута по умолчанию для соседа 2 с IP-адресом 192.168.1.2:

Сосед—l(config—bgp—65001)# neighbor 192.168.1.2 remote—as 65002
Сосед—l(config—bgp—65001)# address—family ipv4 unicast
Сосед—l(config—bgp—65001—af4—unicast)# neighbor 192.168.1.2 default-originate

Сосед—2(config—bgp—65002)# neighbor 192.168.1.1 remote—as 65001

У соседа 2 появится маршрут по умолчанию на шлюз с IP-адресом соседа, от которого пришел анонс.

B 0.0.0.0 via 192.168.1.1

49.6.6 Фильтрация анонсов

Команды фильтрации анонсов доступны из подсекции семейства адресов.

49.6.6.1 Distribute-list

Фильтрация при помощи distribute-list использует списки доступа маршрутизатора. Включить такую фильтрацию можно командой:

(config—bgp—65001—af4—unicast)# neighbor <ip>|<group> distribute-list <racl_name> | <racl_num> in|out

<racl_name>|<racl_num> - имя либо номер списка доступа маршрутизатора.

distribute-list in - фильтруются входящие анонсы, distribute-list out - фильтруются исходящие анонсы.

Пример использования списка доступа myacl, разрешающего только пути в сеть 10.0.0.0/8, для фильтрации анонсов, получаемых от соседа с IP-адресом 192.168.1.2:

(config)# router access—list myacl permit 10.0.0.0/8 (config—bgp—65001—af4—unicast)# neighbor 192.168.1.2 distribute-list myacl in

Отменить использование этого фильтра можно командой:

(config—bgp—65001—af4—unicast)#no neighbor <ip>|<group> distribute-list <racl_name> | <racl_num> in|out

49.6.6.2 Filter-list

Фильтрация при помощи filter-list использует списки доступа по AS-путям. Этот список применяет строку с регулярным выражением для поиска в атрибуте АБ_PATH. Включить такую фильтрацию можно командой:

(config —bgp—65001—af4—unicast)# neighbor <ip>|<group> filter—list <apath_name> in|out

<apath_name> - имя списка доступа по AS-путям.

Пример использования списка доступа myasacl, разрешающего только пути, проходящие через AS 65001, для фильтрации исходящих анонсов для соседа с IP-адресом 192.168.1.2.

(config)# router as—path access—list myasacl permit _65001_
(config —bgp—65001—af4—unicast)# neighbor 192.168.1.2 filter—list myasacl out

Отменить использование этого фильтра можно командой:

(config — bgp—65001—af4—unicast)#no neighbor <ip>|<group> filter—list <apath_name> in|out

49.6.6.3 Prefix-list

Фильтрация при помощи prefix-list использует префиксные списки. Эти списки похожи на списки доступа маршрутизатора. В отличие от списков доступа маршрутизатора, префиксные списки можно применять удаленно (outbond route filtering). Включить такую фильтрацию можно следующей командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> prefix—list <prlist_name> in|out

<prlist_name> - имя префиксного списка.

Пример использования списка доступа с myplist, разрешающего только пути в сеть 10.0.0.0/8, для фильтрации анонсов, получаемых от соседа с IP-адресом 192.168.1.2.

(config)# router access—list myplist permit 10.0.0.0/8
(config —bgp—65001—af4—unicast)# neighbor 192.168.1.2 distribute—list myplist in

Отменить использование этого фильтра можно командой:

(config — bgp—65001—af4—unicast)# no neighbor <ip>|<group> prefix—list <prlist_name> in|out

49.6.6.4 Maximum-prefix

Установить ограничение на количество префиксов (сетей), получаемых от соседа при переполнении памяти, можно командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> maximum —prefix <limit> <thresh>
[(restart <mins>)|warning—only]

<limit> - число получаемых префиксов.

<thresh> - процент от ограничения, при превышении которого выдавать предупреждение,

restart <mins> - рестарт BGP-соединения немедленно либо через несколько минут, если превышен лимит.

warning-only - не делать рестарт, а только выдавать предупреждение.

Отменить такое ограничение можно командой:

(config —bgp—65001—af4—unicast)# no neighbor <ip>|<group> maximum —prefix

49.6.6.5 Route-map

Фильтрацию при помощи карты маршрутов можно включить командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> route—map <rmap_name> in|out

in|out - применяет карту к входящим либо исходящим путям.

Отменить использование карты маршрутов можно командой:

(config — bgp—65001—af4—unicast)# no neighbor <ip>|<group> route—map <rmap_name> in|out

49.6.7 Замена полносвязного графа BGP

49.6.7.1 IBGP-конфедерация (confederation)

Конфедерации описаны в RFC 5065 «Autonomous System Confederations for BGP». Конфедерация под своим номером объединяет несколько AS. Для BGP-маршрутизаторов вне конфедерации, эти AS представляются, как единая AS. AS внутри конфедерации общаются между собой по EBGP.

Конфедерация позволяет упростить задачу настройки BGP-маршрутизаторов внутри одной организации. Без использования конфедерации требуется что бы организация, имеющая один публичный номер AS, обеспечила внутри своей AS полносвязное IBGP-соединение.

Для объединения в конфедерацию следует указать номер конфедерации, сделать это можно командой:

(config —bgp—65001)# bgp confederation identifier <AS>

<AS> - номер AS, которым будут представляться члены конфедерации.

Пример включения BGP-маршрутизатора из AS 65001 в конфедерацию с номером 1000.

(config —bgp—65001)# bgp confederation identifier 1000

Исключить узел из конфедерации можно командой:

(config —bgp—65001)# no bgp confederation identifier

Для обеспечения связей внутри конфедерации следует указать остальные AS, которые входят в конфедерацию. Сделать это можно командой:

(config —bgp—65001)# bgp confederation peers <AS>

Пример добавления в конфедерацию BGP-маршрутизаторов из AS 65002

(config —bgp—65001)# bgp confederation peers 65002

Если AS больше не член конфедерации, то удалить ее можно командой:

(config —bgp—65001)# no bgp confederation peers <AS>

49.6.7.2 IBGP отражатель маршрутов (cluster, route-reflector-client)

Отражатель маршрутов описан в RFC 4456 «BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)». Отражатель маршрутов, это роль BGP-маршрутизатора в BGP-кластере. Внутри одной AS BGP-маршрутизаторы можно разделить на кластеры (группы), в каждом кластере выбрать BGP-маршрутизатор на роль отражателя маршрутов. В этом случае полносвязное BGP-соединение требуется обеспечить только между отражателями. Остальные BGP-маршрутизаторы, члены кластера, будут получать маршруты через своего отражателя, т.е. являться его клиентами.

Чтобы настроить BGP-маршрутизатор, как отражатель маршрутов, следует присвоить ему идентификатор кластера. Сделать это можно командой:

(config — bgp—65001)# bgp cluster—id <id>

<id> - номер кластера.

Удалить номер кластера можно командой:

(config — bgp—65001)# no bgp cluster—id

Затем следует обозначить клиентов кластера. Сделать это можно командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> route—reflector—client

При этом не требуется, чтобы клиенты в кластере имели полносвязную топологию, так как отражатель передает все маршруты между всеми клиентами. Если же клиенты кластера имеют полносвязную топологию, то передачу маршрутов между клиентами одного кластера следует отключить на отражателе. Сделать это можно командой:

(config — bgp—65001)# no bgp client—to—client reflection

49.6.7.3 EBGP сервер маршрутов (route-server-client)

Сервер маршрутов является центром в топологии звезда, при которой EBGP-соседи используют его, как транзитный узел, для обмена между собой. Для включения централизованного обмена на сервере следует объявить соседей, как Route Server Client. Сделать это можно командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> route—server—client

49.6.8 Перемещение BGP-путей в IP-маршрутизатор

49.6.8.1 Атрибуты пути

Пути имеют атрибуты, которые разделены на 4 категории:

1. Хорошо известные, обязательные (Well-known mandatory) —должны распознаваться и присутствовать во всех анонсах: [14]

14 AS_PATH (Autonomous system path) - определяют автономные системы, через которые доступна сеть назначения;

- NEXT_HOP - определяет IP-адрес пограничного маршрутизатора, который должен рассматриваться, как шлюз к сети назначения в таблице маршрутизации;
- ORIGIN - определяет происхождение пути.

2. Хорошо известные, не обязательные (Well-known discretionary) —должны распознаваться, но наличие в анонсах не обязательно:
   - LOCAL_PREF (Local preference) - используется чтобы сообщить соседям внутри своей автономной системы степень предпочтения пути;
   - ATOMIC_AGGREGATE - используется для информирования соседей о суммарном пути.
3. Дополнительные пересылаемые (Optional transitive) — могут не распознаваться, но должны передаваться соседям:
   - AGGREGATOR - содержит номер последней AS, и IP-адрес BGP-маршрутизатора, который сформировал суммарный путь.
4. Дополнительные не пересылаемые (Optional non-transitive) — могут не распознаваться и отбрасываться:
   - MULTI_EXIT_DISC (Multi-exit discriminator, MED) - может использоваться при выборе одного из нескольких путей к соседней автономной системе.

49.6.8.2 Алгоритм выбора лучшего пути

1. Weight. Лучшим считается путь с наибольшим значением веса.
2. LOCAL_PREF. Если вес путей одинаков, то выбирается путь с наибольшим значением атрибута LOCAL_PREF.
3. Локальные пути. Если атрибуты LOCAL_PREF одинаковы, пути объявленные командами network, redistribute и aggregate-address предпочитаются над путями, полученными от соседей.
4. AS_PATH. Если нет локальных путей, то выбирается путь с самым коротким атрибутом AS_PATH.
5. ORIGIN. Если все пути имеют одинаковую длину атрибута AS_PATH, то выбирается путь с меньшим атрибутом ORIGIN (IGP < EGP < Incomplete).
6. MED. Если атрибут ORIGIN одинаков, то выбирается путь с наименьшим атрибутом MED.
7. EBGP или IBGP. Если атрибут MED одинаков, то предпочтение отдается маршрутам, полученным по EBGP, над маршрутами, полученными по IBGP.
8. Метрика маршрута до next hop. Чем меньше, тем предпочтительнее.
9. Маршрут первый появившийся в таблице (самый старый).

49.6.8.3 Вес пути (Weight)

Weight, это локальный атрибут, который не передается соседям. По умолчанию, значение веса 32768 для локальных путей, которые созданы на данном роутере, и 0 - для всех остальных путей. Если необходимо устанавливать определенный вес для пути, то сделать это можно командой:

(config —bgp—65001—af4—unicast)# neighbor <ip>|<group> weight <n>

<n> - число определяющее вес пути, по умолчанию 0.

Пример увеличения веса путей, полученных от соседа с IP-адресом 192.168.1.2.

(config — bgp—65001—af4—unicast)# neighbor 192.168.1.2 weight 40000

Посмотреть вес можно командой:

show bgp

Вернуть значения по умолчанию можно командой:

(config — bgp—65001—af4—unicast)# no neighbor <ip>|<group> weight

49.6.8.4 Атрибут LOCAL_PREF (local-preference)

Атрибут присваивается маршрутам, полученным через EBGP, и локальным по команде network, атрибут предается только по IBPG. Установить определенное значение можно командой:

(config — bgp—65001)# bgp default local —preference <n>

<n> новое значение атрибута, по умолчанию 100.

Вернуть значение по умолчанию можно командой:

(config — bgp—65001)# no bgp default local —preference

49.6.8.5 Атрибут AS_PATH

49.6.8.5.1 Разрешить принимать пути с номером собственной AS (allowas-in) По умолчанию для исключения петель маршрутизатор отбрасывает путь, полученный по EBGP, если видит в AS_PATH номер своей AS. Можно отменить это правило командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> allowas—in [<n>]

Вернуть поведение по умолчанию можно командой:

(config — bgp—65001—af4—unicast)# no neighbor <ip>|<group> allowas—in

49.6.8.5.2 Изменить номер собственной AS для отдельных соседей (local-as) Можно указать другую локальную AS для отдельного EBGP-соседа, отличную от той, в которую входит данный маршрутизатор. Сосед при этом должен указать другую AS в команде network remote-as. Сделать это можно командой:

(config — bgp—65001)# neighbor <ip>|<group> local—as <AS> [no—prepend]

<AS> - номер AS отличный от собственной.

no-prepend - пути, полученные через EBGP, не предваряются номером local-as при распространении по IBGP.

Отменить замену номера можно командой:

(config — bgp—65001)# no neighbor <ip>|<group> local—as

49.6.8.5.3 Проверить номер первой AS (enforce-first-as) Принимать пути только от EBGP-соседей, чей номер AS стоит первым в AS_PATH. Включить такую проверку можно командой:

(config — bgp—65001)# bgp enforce—first—as

Отменить проверку можно командой:

(config — bgp—65001)# no bgp enforce—first—as

49.6.8.5.4 Удалить частные AS (remove-private-as) Номера частных автономных систем с 64512 по 65535 используются в частных сетях и не используются интернет-провайдерами. Можно удалять их из получаемых анонсов от EBGP-соседей. Включить удаление можно командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> remove—private—AS

Отменить удаление можно командой:

(config — bgp—65001—af4—unicast)# no neighbor <ip>|<group> remove—private—AS

49.6.8.5.5 Длина атрибута AS_PATH (confed, ignore) По умолчанию AS конфедераций не влияют на выбор «лучшего» маршрута. Если требуется учитывать также AS конфедераций, сделать это можно командой:

(config — bgp—65001)# bgp bestpath as—path confed

Если требуется отменить сравнение путей по длине атрибута AS_PATH, сделать это можно командой:

(config — bgp—65001)# bgp bestpath as—path ignore

49.6.8.6 Атрибут HEXT_HOP (next-hop-self)

По умолчанию, когда путь, полученный по EBGP, анонсируется IBGP-соседу, атрибут NEXT_HOP (шлюз) не изменяется. В случае если этот шлюз недоступен, его нужно подменить на доступный IP-адрес соседа. Сделать это можно командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> next—hop—self

По умолчанию, когда путь анонсируется EBGP-соседу, атрибут NEXT_HOP всегда меняется на IP-адрес соседа, который анонсирует маршрут. Отменить подмену можно командой:

(config — bgp—65001—af4—unicast)# no neighbor <ip>|<group> next—hop—self

49.6.8.7 Атрибут MED (always-compare-med, confed, missing-as-worst, deterministic-med)

По умолчанию MED сравнивается только для путей из одной и той же AS, т.е. для путей с одинаковой первой AS в атрибуте AS_PATH. Если требуется сравнивать MED для любых путей, сделать это можно командой:

(config — bgp—65001)# bgp always—compare—med

По умолчанию, MED не сравнивается для путей из конфедераций. Если требуется учитывать также конфедерации, сделать это можно командой:

(config — bgp—65001)# bgp bestpath med confed

По умолчанию, если атрибут MED отсутствует в полученном пути, то ему присваивается значение 0 (самый высокий приоритет). Если требуется присвоить отсутствующему атрибуту самый низкий приоритет (4294967295), то сделать это можно командой:

(config — bgp—65001)# bgp bestpath med missing—as—worst

Можно изменить алгоритм выбора лучшего пути, таким образом, что все пути в одну сеть будут сравниваются друг с другом, несмотря на разные AS этих путей и порядок получения. Сделать это можно командой:

(config — bgp—65001)# bgp deterministic—med

49.6.8.8 Сравнить по идентификаторам (compare-routerid)

Полностью одинаковые пути EBGP не сравниваются, а лучшим выбирается первый полученный (самый старый). Можно заменить такой выбор на выбор пути с наименьшем идентификатором. Сделать это можно командой:

(config — bgp—65001)# bgp bestpath compare-routerid

49.6.8.9 Контроль атрибутов через карту маршрутов

49.6.8.9.1 Вес пути (set weight) Установить вес можно следующей командой:

(config — route—map—myrmap)# set weight <n>

<n> - вес пути

49.6.8.9.2 Атрибут LOCAL_PREF (set local-preference) Установить атрибут можно следующей командой:

(config—route—map—myrmap)# set local—preference <n>

<n> - новое значение атрибута

49.6.8.9.3 Атрибут AS_PATH (match/set as-path) Критерий отбора по спискам доступа по AS-путям можно установить следующей командой:

(config — route—map—myrmap)#match as—path <as_path_acl>

Изменить атрибут AS_PATH, добавив либо удалив номера AS можно следующей командой:

set as—path prepend|exclude <as_numl> [<as_num2>] [<as_num3>] ...

prepend - добавить номера AS.

exclude - удалить номера AS.

49.6.8.9.4 Атрибут ORIGIN (match/set origin) Критерий происхождения маршрута:

(config—route—map—myrmap)# match origin egp|igp|incomplete

Установить происхождение маршрута: (config—route—

map—myrmap)# set origin egp|igp|incomplete

49.6.8.9.5 Атрибуты AGGREGATOR и ATOMIC_AGGREGATE (set aggregator, atomic—aggregate) При формировании суммарного маршрута BGP-маршрутизатор добавляет к пути атрибуты AGGREGATOR и ATOMIC_AG GREG ATE. Установить атрибут aggregator можно следующей командой:

(config—route—map—myrmap)# set aggregator as <as_num> <ip>

<as_num> - номер AS, обычно последняя AS.

<ip> - IP-адрес, обычно адрес BGP-маршрутизатора сформировавшего маршрут.

Установить атрибут atomic-aggregate можно следующей командой:

(config—route—map—myrmap)# set atomic—aggregate

49.6.8.9.6 Атрибут Ordinator-ID (set originator-id) При работе BGP-маршрутизаторов в кластере к путям, которые проходят через отражатель маршрутов, добавляется атрибут Ordinator-ID. Данный атрибут - это IP-адрес - идентификатор маршрутизатора источника маршрута. Атрибут нужен для исключения петель. BGP-маршрутизатор отбрасывает анонсы со своим Ordinator-ID. Установить атрибут originator ID можно следующей командой:

(config —route—map—myrmap)# set originator—id <ip>

49.6.8.10 Сохранение атрибутов (attribute-unchanged)

При анонсе путей, которые были получены по BGP, BGP-маршрутизатору можно запретить менять исходные атрибуты пути. Сделать это можно командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> attribute—unchanged [as—path]
[next—hop] [med]

Параметры as-path, next-hop и med - определяют, какие именно атрибуты оставить в неизменном виде.

Отменить сохранение атрибутов можно командой:

(config — bgp—65001—af4—unicast)# no neighbor <ip>|<group> attribute—unchanged

49.6.8.11 Административная дистанция (AD)

AD используется для изменения приоритета путей, полученных от разных протоколов. Работает после выбора лучшего пути до помещения пути в таблицу маршрутизации. Чем меньше AD, тем приоритетнее путь. Значения AD: подключенный интерфейс 0, статический маршрут 1, EBGP 20, OSPF 110, RIP 120, IBGP 200.

Если требуется изменить значение AD, которые будут использоваться для всех соседей, то сделать это можно командой:

(config —bgp—65001—af4—unicast)# distance bgp <ext> <int> <local>

<ext> - AD для путей, полученных через EBGP, (по умолчанию 20).

<int> - AD для путей, полученных через IBGP, (по умолчанию 200).

<local> - AD для путей, настроенных вручную, через команду network, (по умолчанию 200).

Пример задания AD 200 для любых путей из BGP:

(config — bgp—65001—af4—unicast)# distance bgp 200 200 200

Возврат значений по умолчанию выполняется командой:

(config — bgp—65001—af4—unicast)# no distance bgp

Если требуется настроить значения AD для путей от конкретного соседа, это выполняется командой:

(config — bgp—65001—af4—unicast)# distance (<n> <ip/m> [<racl>])

<n> - значение AD.

<ip/m> - IP-адрес соседа.

<racl> - список доступа, который определяет, каким конкретно путям присваивать AD.

Пример указания AD 200 для путей, полученных от соседа с IP-адресом 195.220.1.2/32. Предварительно создан список доступа myadracl, разрешающий сеть 10.1.2.0/24. AD изменяется для путей в эту сеть.

```
(config)# router access—list myadracl permit 10.1.2.0/24
(config —bgp—65001—af4—unicast)# distance 200 195.220.1.2/32 myadracl
```

Для удаления изменений AD и возврата значений по умолчанию используется команда:

(config — bgp—65001—af4—unicast)# no distance <n> <ip/m>

49.6.8.12 Backdoor

Иногда требуется уменьшить приоритет пути, полученного по EBGP (AD 20), если есть путь в ту же сеть, но полученный, например по OSPF (AD 110). Сделать это можно командой, которая устанавливает административную дистанцию 200 вместо 20 для сети, полученной по EBGP:

(config — bgp—65001—af4—unicast)# network <ip/m> backdoor

<ip/m> - IP-адрес и маска сети, получаемой по EBGP, для которой требуется уменьшить приоритет. Данная сеть не анонсируется по BGP другим соседям.

49.6.9 Сообщества

49.6.9.1 Атрибуты сообществ (send-community)

Сообщества описаны в RFC 1997 «BGP Communities Attribute» и RFC 4360 «BGP Extended Communities Attribute». Предназначены для облегчения управления анонсами на основе политик. Сообщества добавляют атрибут COMMUNITY к пути. Вид атрибута AS:Number, где AS - номер AS, Number - номер политики. По номеру политики можно устанавливать различные атрибуты, например LOCAL_PREF. Некоторые сообщества имеют зарезервированные имена:

internet - анонсировать этот путь в интернет. Любой путь принадлежит этому сообществу.

local-AS - это сообщество запрещает передачу путей за пределы собственной AS.

no-advertise - запрещает анонсы любому соседу.

no-export - запрещает анонсы EBGP-соседям.

Для того чтобы использовать атрибуты требуется разрешить пересылку и прием их для соседей. Сделать это можно командой:

(config —bgp—65001—af4—unicast)# neighbor <ip>|<group> send—community
standard | extended | both

standard, extended и both - определяет, какие списки сообществ отправлять и принимать. Списки сообществ бывают со стандартными (standard) и расширенными (extended) атрибутами; в каждом из списков можно применять регулярные выражения.

49.6.9.2 Стандартный список (community-list)

Создать стандартное сообщество можно командой:

(config)# router community—list <1..99>|(standard <name>) permit|deny <com_name>

<1..99> - задает номер списка сообществ.

standard <name> задет имя списка сообществ.

<com_name> - имя сообществ, может иметь вид <NN>:<NN> либо одно из стандартных названий internet, local-AS, no-advertise или no-export

Можно создать список сообществ с регулярным выражением:

(config)# router community—list <100..500>|(expanded <name>) permit|deny <regexp>

<regexp> - регулярное выражение.

49.6.9.3 Расширеный список (extcommunity-list)

Создать расширенный список сообществ можно командой:

(config)# router extcommunity—list <1..99>|(standard <name>) permit|deny <([rt
<ip>:NN|NN:NN] [soo <ip>:NN|NN:NN])>

rt <ip>:NN|NN:NN - это (Route Target) идентификатор маршрутизатора источника пути с этим сообществом.

soo <ip>:NN|NN:NN - это (Site of Origin) идентификатор сайта источника пути.

Можно создать список сообществ с регулярным выражением.

(config)# router extcommunity—list < 100..500>|(expanded <name>) permit|deny <regexp>

49.6.9.4 Фильтрация при помощи карт маршрутов

Критерий сравнения атрибутов пути со стандартным списком задается при помощи команды:

(config—route—map—myrmap)# match community <num> | <comlist_name> [exact—match]

exact-match - точное совпадение.

Критерий сравнения атрибутов пути с расширенным списком списком задается при помощи команды:

(config—route—map—myrmap)# match extcommunity <num>|<ecomlist_name> [exact—match]

Установка атрибутов стандартного сообщества осуществляется при помощи команды:

(config—route—map—myrmap)# set community
<NN>[:<NN>]|internet|local—AS|no—advertise|no—export|additive|none ...

Установка атрибутов расширенного сообщества осуществляется при помощи команды:

(config—route—map—myrmap)# set extcommunity [rt <ip>:NN|NN:NN] [soo <ip>:NN|NN:NN] ...

Удаление атрибутов стандартного или расширенного сообщества осуществляется при помощи команды:

(config—route—map—myrmap)# set comm—list <num>|<comlist_name> delete

49.6.10 Дополнительные возможности BGP-маршрутизатора

49.6.10.1 Согласование возможностей (capability)

При согласовании соединения BGP-маршрутизаторы обмениваются информацией о своих возможностях. За каждой хорошо известной возможностью закреплен определенный код. Например:

1 - Multiprotocol Extensions for BGP-4.

2 - Route Refresh Capability for BGP-4.

3 - Outbound Route Filtering Capability.

4 - Multiple routes to a destination capability.

5 - Extended Next Hop Encoding.

64 - Graceful Restart Capability.

67 - Support for Dynamic Capability (capability specific).

По умолчанию включен минимальный набор хорошо известных возможностей. Для проверки точного совпадения возможностей данного маршрутизатора с соседним можно использовать команду:

(config — bgp—65001)# neighbor <ip> strict—capability—match

Некоторые старые версии BGP не умеют согласовывать возможности. Для отключения согласования своих возможностей с соседом можно использовать команду:

(config — bgp—65001)# neighbor <ip>|<group> dont—capability—negotiate

Можно также игнорировать согласованные возможности и принудительно использовать все свои возможности. Сделать так можно командой:

(config — bgp—65001)# neighbor <ip>|<group> override—capability

Включить возможность динамического согласования можно командой:

(config — bgp—65001)# neighbor <ip>|<group> capability dynamic

49.6.10.2 Удаленное применение фильтров (outbound route filtering)

Если сосед поддерживает возможность удаленного применения фильтров, то можно настроить префиксные списки, для отправки и применения их на стороне соседа. Таким образом сокращается объем анонса. Включить такую возможность можно командой:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> capability orf prefix—list
send | receive | both

49.6.10.3 Мягкая перезагрузка (graceful-restart)

Механизм «Graceful-restart» описан в RFC 4724 «Graceful Restart Mechanism for BGP». Этот механизм позволяет сохранять состояние BGP-соединений и не трогать IP-маршрутизацию, в процессе перезапуска BGP-маршрутизатора, что снижает нагрузку и позволяет избавиться от временных петель в маршрутизации. Включить механизм «Graceful Restart» можно командой:

(config — bgp—65001)# bgp graceful —restart [stalepath—time <t>]

<t> - время в секундах, которое BGP-маршрутизатор ждет с момента получения сигнала о перезапуске, прежде чем сбросить BGP-соединения. По умолчанию 360 секунд.

Выключить этот механизм можно командой:

(config — bgp—65001)# no bgp graceful —restart [stalepath—time]

49.6.10.4 Мягкое применение политик без обрыва сессий (soft-reconfiguration)

При изменении настроек, например входящих фильтров, BGP требуется заново получить от соседа пути, чтобы по-новому их отфильтровать. Сделать это можно, принудительно очистив таблицу BGP командой clear bgp, в этом случае информация удаляется, соединения обрываются

и устанавливаются заново. Можно воспользоваться более мягким вариантом clear bgp soft in, при котором изменения применяются без обрыва сессий. Для того чтобы работал такой вариант BGP требуется хранить информацию о всех путях от соседа (не отфильтрованных) в памяти. При перезагрузке фильтры применятся к путям в памяти, соединение не разорвется.

Для того чтобы включить данный механизм нужно выполнить команду:

(config — bgp—65001—af4—unicast)# neighbor <ip>|<group> soft—reconfiguration inbound

После этого можно пользоваться clear bgp soft in.

49.6.10.5 Быстрое применение политик с принудительным обновлением анонсов (route refresh)

Более совершенная возможность заменяющая «soft reconfiguration» позволяет принудительно запросить свежий анонс у соседа либо отправить анонс соседу без разрыва сессии.

Чтобы использовать возможность не требуется дополнительных команд конфигурации. Принудительно запросить анонс можно командой:

clear bgp in

Отправить анонс можно командой:

clear bgp out

49.6.10.6 Подавление мигающих путей (dampening)

Механизм защиты от мигающих (хлопающих) путей описан в RFC 2439 «BGP Route Flap Damping». Когда путь анонсируется и быстро исчезает из анонса, то данному пути начисляется штраф 1000 очков и путь запоминается, как «мигающий». Если он снова появился и исчез, то еще 1000 очков и т.д. за каждое мигание. Если число очков превысило настроенный предел, то путь помечается, как «подавленный». Подавленные пути не анонсируются и не передаются в 1Р-маршрутизатор.

Включить этот механизм можно командой:

(config —bgp—65001)# bgp dampening [<halflife> [<reuse_start> <suppress_thresh> <suppress_dur>]]

<halflife> - время в минутах, после которого штраф уменьшается наполовину, (по умолчанию 15 минут).

<reuse_start> - размер штрафа, ниже которого путь перестаёт быть подавленным, (по умолчанию 750 очков).

<suppress_thresh> - размер штрафа, выше которого путь подавляется, (по умолчанию 2000 очков).

<suppress_dur> - время в минутах, в течение которого путь подавляется, обычно в четыре раза больше, чем halflife, (по умолчанию 60 минут).

Пример настройки механизма защиты от мигающих путей со значениями по умолчанию:

(config —bgp—65001)# bgp dampening

Выключить механизм можно командой:

(config—bgp—65001)# no bgp dampening

49.7 Карты маршрутов

49.7.1 Описание карт маршрутов

Карты маршрутов позволяют изменять атрибуты путей и маршруты по определенным критериям. Они обычно применяются для управления маршрутами в протоколах динамической маршрутизации, чаще всего для BGP.

Карта маршрутов - это набор правил. Каждое правило имеет свой порядковый номер и политику. Политика может быть либо разрешающей вносить изменения и передавать маршрут на дальнейшую обработку, либо запрещающей обрабатывать маршрут.

Правило содержит разделы:

- Описание (description) задает комментарий для правила и служит для удобства;
- Критерии (match) задают условия, при которых срабатывает правило. Если маршрут попал под заданные критерии, то при разрешающей политике этого правила, выполняются остальные разделы: установки (set), вызов другой карты (call) и действие (action). При запрещающей - маршруты, попавшие под правило, отбрасываются. Если маршрут не попал под критерии, то рассматриваются критерии правила со следующим порядковым номером;
- Установки (set) изменяют атрибуты путей и маршруты;
- Вызов другой карты (call) позволяет выполнить правила из другой карты маршрутов;
- Действие (action) определяет поведение после выполнения правила. Если действие отсутствует, то выполнение карты завершается. В качестве действия можно задать переход на другое правило (goto).

Если маршруты не попали ни под одно правило, то они отбрасываются. Чтобы разрешить дальнейшую обработку таких маршрутов, последним в карте должно идти пустое правило с разрешающей политикой.

49.7.2 Создание правил

Создание правил route-map.

Каждое правило в карте маршрутов изменяется отдельно в режиме редактирования. Перейти в режим редактирования правила можно командой:

(config)# router route—map <rmap_name> permit|deny <seq>

<rmap_name> - имя карты маршрутов, в которую заносится правило,

permit - разрешающая политика правила,

deny - запрещающая политика правила.

<seq> - порядковый номер правила. Рекомендуется задавать порядковый номер через десятки т.е. 10, 20, 30 и т.д. Если в дальнейшем понадобиться вставить новое правило между существующими, можно будет использовать номера 11, 12, 22 и пр. и не придется переписывать всю карту.

Пример создания разрешающего правила под номером 10 для карты с именем myrmap:

```
(config)# router route—map myrmap permit 10
(config—route—map—myrmap)#
```

Удалить правило можно командой:

```
(config)# no router route—map <rmap_name> permit|deny <seq>
```

49.7.3 Просмотр правил

Посмотреть правила, созданные в определенной карте, можно командой:

```
#  show router route—map <mnap_name>
```

Команда выдает список правил, разделенный на четыре секции: ZEBRA, RIP, OSPF и BGP. В каждой секции показан один и тот же набор правил. Критерии и установки правил (match и set), могут меняться в зависимости от секции. Например, критерии и установки, специфические для BGP, будут показаны только в BGP-секции.

Пример вывода правил карты маршрутов:

```
(config—route—map—myrmap)# do show
call myrmap2
description "first rule"
match ip address myacl
set as—path prepend 65010
set metric 1
# show router route—map myrmap
ZEBRA:
route—map myrmap, permit, sequence 10
  Description:
    "first rule"
  Match clauses:
  ip address myacl
  Set clauses:
  Call clause:
    Call myrmap2
  Action:
    Exit routemap
RIP:
```

```
 route—map myrmap, permit, sequence 10
   Description:
     "first rule"
   Match clauses:
   ip address myacl
   Set clauses:
   metric 1
   Call clause:
     Call myrmap2
   Action:
     Exit routemap
OSPF:
 route—map myrmap, permit, sequence 10
   Description:
     "first rule"
   Match clauses:
   ip address myacl
   Set clauses:
   metric 1
   Call clause:
     Call myrmap2
   Action:
     Exit routemap
BGP:
 route—map myrmap, permit, sequence 10
   Description:
     "first rule"
   Match clauses:
   ip address myacl
   peer 192.168.0.3
   Set clauses:
   metric 1
     as—path prepend 65010
   Call clause:
     Call myrmap2
   Action:
     Exit routemap
```

49.7.4 Описание к правилу

Для удобства администрирования можно задать описание к правилу при помощи команды

(config—route—map—myrmap)# description <string>

<string> - строка с описанием.

Удалить описание можно командой:

(config—route—map—myrmap)# no description

49.7.5 Критерии сравнения

49.7.5.1 Интерфейс

Для RIP, OSPF, BGP.

Сравнение интерфейса, с которого доступен маршрут.

(config—route—map—myrmap)# match interface <iface>

<iface> - название интерфейса, например ethernetO.

49.7.5.2 IP-адрес назначения

Для: RIP, OSPF, BGP

Сравнение IP-адреса сети назначения по списку доступа маршрутизатора, либо по префиксному списку:

(config—route—map—myrmap)# match ip address <racl_num>|<racl_name>|(prefix—list <prlist_name>)

<racl_num>|<racl_name> - номер либо имя списка доступа маршрутизатора.

<prlist_name> - имя префиксного списка.

49.7.5.3 IP-адреса шлюза

Для RIP, OSPF

Сравнение IP-адреса шлюза по списку доступа, либо по префиксному списку:

(config—route—map—myrmap)# match ip next—hop <racl_num>|<racl_name\>|(prefix—list <prlist_name>)

49.7.5.4 IP-адрес маршрутизатора, анонсировавшего маршрут

Для BGP.

IP-адрес маршрутизатора, анонсировавшего маршрут:

(config—route—map—myrmap)# match ip route—source <racl_num>|<racl_name>|(prefix—list <prlist_name>)

49.7.5.5 Метрика маршрута (metric). RIP, BGP

(config—route—map—myrmap)# match metric <n>

<n> - метрика маршрута.

49.7.5.6 Тег маршрута

Для RIP.

(config—route—map—myrmap)# match tag \<n\>

<n> - тег маршрута.

49.7.6 Установки

49.7.6.1 Метрика маршрута

Для OSPF, BGP.

Устанавливает метрику, либо изменяет существующую

set metric [+|-]<seq>

<seq> - изменение метрики

49.7.6.2 Источник маршрута

Для ZEBRA.

IP-адрес источника маршрута

set src <ip>

49.7.6.3 Адрес шлюза

Для RIP, BGP.

set ip next-hop <ip>

49.7.6.4 Тег маршрута

Для RIP.

set tag <n>

49.7.7 Вызов другой карты маршрутов

Внутри правила можно вызвать правила из другой карты маршрутов. Сделать это можно командой:

(config — route—map—myrmap)# call <rmap_name>

Удалить такой вызов можно командой:

(config —route—map—myrmap)# no call

49.7.8 Переход на другое правило при выполнении условий

Переход на другое правило при выполнении условий (on-match next, goto, continue)

Если совпадений с критериями нет, то просматривается правило со следующим порядковым номером. При выполнении критериев правила, выполняются указанные изменения и просмотр карты маршрутов закачивается. Если требуется продолжить просмотр, следует использовать команду on-match.

Переход на следующее правило при выполнении критериев можно задать командой:

(config —route—map—myrmap)# on—match next

При выполнении этой команды в конфигурацию вносится строка «оп-match goto N», где N - число на единицу большее порядкового номера правила, например для десятого правила N будет 11.

У этой команды существует синоним, команда continue. При выполнении этой команды в конфигурацию все равно запишется «оп-match goto N».

Явно указать, на какое правило переходить, можно при помощи команды:

(config —route—map—myrmap)# on—match goto <n>

<n> - номер правила.

Если правила с таким номером не существует, то переход осуществляется на первое правило с номером большим, чем указанный. Например, есть правила с номерами 10, 20 и 30. Если из десятого правила указать переход на 11-е, то переход произойдет на 20-е.

Внимание! Нельзя переходить на правила выше текущего, например, из десятого правила нельзя переходить на девятое.

У этой команды существует синоним, команда continue <N>. При выполнении этой команды в конфигурацию все равно запишется «оп-match goto N».

50. MPLS - многопротокольная коммутация по меткам

50.1 Введение

Dionis-NX имеет базовую поддержку MPLS-маршрутизации. MPLS работает поверх ip-маршрутизации. т.е. для работы MPLS необходимо вначале настроить статическую или динамическую ip-маршрутизацию.

В рамках архитектуры MPLS различают следующие типы устройств:

- LSR-маршрутизатор, поддерживающий коммутацию по меткам и традиционную IP-маршрутизацию. (Маршрутизатор *заменяет(swap)* или снимает(pop) метки с пакетов).
- LER-маршрутизатор, подключённый к устройствам, не осуществляющим коммутацию по меткам. (Маршрутизатор назначает(push) или снимает(pop) метки с пакетов).
- MPLS-domain - MPLS-домен - группа соединенных устройств осуществляющих коммутацию по меткам, находящихся под единым административным подчинением и функционирующих в соответствии с единой политикой маршрутизации. MPLS домен образуется LSR-ами, а на границе домена размещаются устройства LER.

Для включения поддержки MPLS-маршрутизации необходимо включить MPLS как глобально на маршрутизаторе, так и на необходимых интерфейсах:

```
adm@DionisNX(config)# mpls ip

adm@DionisNX(config)# interface ethernet0
adm@DionisNX(config — if—ethernet0)# mpls ip
adm@DionisNX(config — if—ethernet0)# exit

adm@DionisNX(config)# interface ethernet1
adm@DionisNX(config — if—ethernet1)# mpls ip
adm@DionisNX(config —if—ethernet1)# exit
```

50.2 Статическая MPLS - маршрутизация

Настройка происходит в два этапа:

1. Настройка LER-маршрутизатора (push);
2. Настройка LSR-маршрутизатора (swap/pop).

Настройка LER-маршрутизатора

```
adm@DionisNX(config)# ip route 192.168.16.0/24 10.1.1.1 label 17
```

Настройка LSR-маршрутизатора

Замена метки:

```
adm@DionisNX(config)# mpls Isp 17 10.1.1.2 18
```

Снятие метки:

```
adm@DionisNX(config)# mpls Isp 17 10.1.1.2 implicit—null
```

50.3 Динамическая MPLS - маршрутизация

В Dionis-NX динамическое назначение меток происходит при помощи протокола LDP.

Для включения динамической MPLS-маршрутизации необходимо:

1. Настроить динамическую ip-маршрутизацию (например OSPF).
2. Включить поддержку MPLS глобально и на необходимых интерфейсах.
3. Включить поддержку LDP-протокола на необходимых интерфейсах.
4. Указать адрес TCP-соединения для LDP-протокола (адрес одного из интерфейсов, на котором включена поддержка LDP-протокола).

Чтобы войти в режим настроек LDP-протокола в необходимо выполнить команду:

```
adm@DionisNX(config)# router Ipd
```

Для включения поддержки LDP-протокола на интерфейсе необходимо выполнить команду:

```
adm@DionisNX(config)# router Ipd
adm@DionisNX(router— Idp)# address—family ipv4
adm@DionisNX(router— Idp—af4)# iface ethernet 0
```

Для настройки адреса TCP-соединения для LDP-протокола необходимо выполнить команду:

```
adm@DionisNX(config)# router Ipd
adm@DionisNX(router— Idp)# address—family ipv4
adm@DionisNX(router— Idp—af4)# discovery transport—address 192.168.1.1
```

51. VRF - Virtual Routing and Forwarding

51.1 Введение

VRF - технология, позволяющая реализовывать на базе одного физического маршрутизатора несколько виртуальных - каждый со своей независимой таблицей маршрутизации. О основном данная технология применяется для изоляции трафика в сетях провайдера.

Каждый такой виртуальный маршуртизатор — практически является отдельным VPN. Их таблицы маршрутизации, FIB, список интерфейсов и прочие параметры не пересекаются — они строго индивидуальны и изолированы. Ровно так же они обособлены и от самого физического маршрутизатора. Но между ними возможна коммуникация.

Различают два вида VRF: VRF-Lite (VPN без MPLS) и MPLS-L3VPN.

На данный момент в DionisNX реализован поддержка обоих видов:

- VRF-Lite: доступна как статическая маршрутизация, так и динамическая (с использованием OSPF или BGP).
- MPLS-L3VPN: динамическая маршрутизация на основе протокола BGP.

51.2 Базовая настройка

Базовая настройка VRF в DionisNX заключается в создании виртуального интерфейса vrf и добавления к нему других интерфейсов. В данном примере создается виртуальный интерфейс vrf 1 и к нему добавляется реальный интерфейс ethernet 0:

```
adm@DionisNX(config)# interface ethernet 0
adm@DionisNX(config — if—ethernet0)# ip address 192.168.16.2/24
adm@DionisNX(config — if—ethernet0)# enable
adm@DionisNX(config —if—ethernet0)# exit
adm@DionisNX(config)# interface vrf 1
adm@DionisNX(config — if—vrfl)# slave ethernet 0
adm@DionisNX(config —if—vrfl)# enable
```

Примечание: Имя vrf интерфейса имеет значение только в пределах одного устройства.

51.3 VRF-Lite

При создании интерфейса vrf автоматически создается виртуальная таблица маршрутизации. Для редактирования маршрутов в данной таблице необходимо зайти в секцию **router vrf <N>,** где **N** соответствует номеру созданного vrf интерфейса.

```
adm@DionisNX(config)# router vrf 1 adm@DionisNX(config—
vrfl)# ip route 10.0.0.0/8 192.168.16.254
```

Для просмотра таблицы маршрутизации для конкретного vrf необходимо выполнить команду:

adm@DionisNX# show router vrf 1 ip route

Связь между различными vrf и глобальной таблицей маршрутизации возможна через параметр **nexthop-vrf** при создании/редактировании маршрутов. Пример использования маршрутов из глобальной таблицы маршрутизации в таблице маршрутизации vrf 1:

adm@DionisNX(config—vrfl)# ip route 10.0.0.0/8 192.168.16.254 nexthop-vrf main

Пример использования маршрутов из vrf 2 таблицы маршрутизации в таблице маршрутизации vrf 1:

adm@DionisNX(config—vrfl)# ip route 10.0.0.0/8 192.168.16.254 nexthop-vrf 2

51.3.1 Пример использования VRF-Lite совместно с OSPF

Ниже приведена схема, реализующая изоляцию сети NET1(NET1_1 + NET1_2) от сети NET2(NET2_1 + NET2_2). Так сеть NET1 использует для передачи своего трафика маршруты VRF1, а сеть NET2 - VRF2. Между узлами настроена динамическая маршрутизация OSPF для каждого VRF. Таким образом на узле запущено одновременно два OSPF процесса (для vrfl и для vrf2).

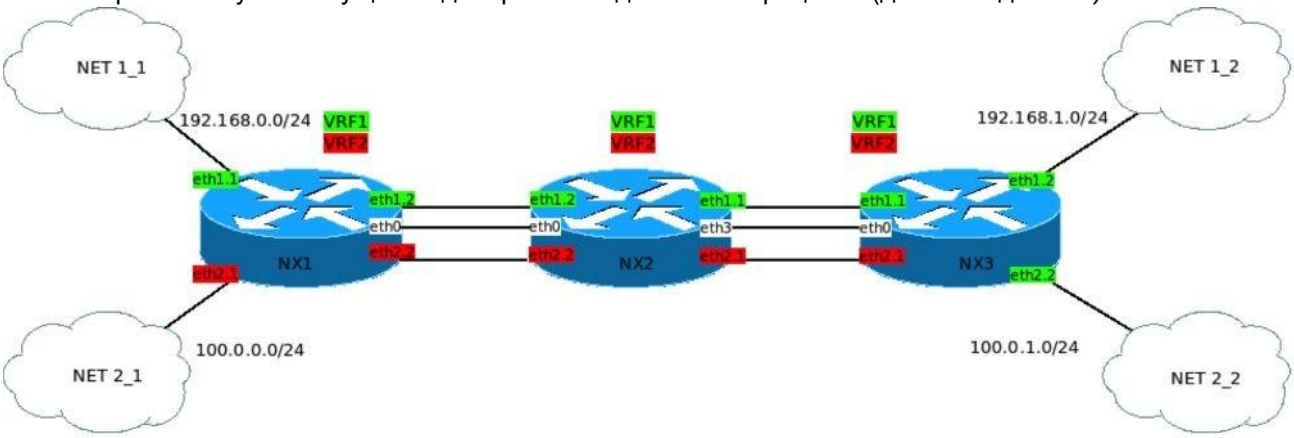

Рис. 51.1: Схема 1: vrf-lite + OSPF

Примеры конфигураций узлов для данной схемы приведены в Приложении.

51.4 MPLS-L3VPN

MPLS L3VPN позволяет избавиться от некоторых этапов настройки VRF-Lite:

- Настройка VRF на каждом узле между точками подключения
- Настройка отдельных интерфейсов для каждого VRF на каждом узле.

Настройка отдельных процессов IGP для каждого VRF на каждом узле.

- Необходимость поддержки таблицы маршрутизации для каждого VRF на каждом узле.

Процесс настройки можно описать в следующем порядке:

- Настройка базовой связности магистральной сети: IP-адреса, IGP.
- Включение MPLS и LDP
- Создание VRF и привязка к интерфейсам.
- Настройка протокола маршрутизации с CE.
- Настройка BGP и MBGP

Терминология:

CE — Customer Edge router — граничный маршрутизатор клиента, который подключен в сеть провайдера.

PE — Provider Edge router — граничный маршрутизатор провайдера. К нему подключаются CE. На PE начинаются и заканчиваются VPN. На нём расположены интерфейсы, привязанные к VPN. Также PE назначает и снимает сервисные метки. PE являются Ingress LSR и Egress LSR. PE должны знать таблицы маршрутизации каждого VPN, т.к. они принимают решение о том, куда посылать пакет, как в пределах провайдерской сети, так и в плане клиентских интерфейсов.

P — Provider router — транзитный маршрутизатор, который не является точкой подключения — пакеты VPN проходят через него без каких-либо дополнительных обработок, т.е просто коммутируются по транспортной метке. P нет нужды знать таблицы маршрутизации VPN или сервисные метки. На P нет интерфейсов привязанных к VPN.

Для связи PE и P роутеров в MPLS-L3VPN используется семейство адресов ipv4/ipv6 vpn в секции router bgp.

Пример настройки PE роутера для связи с P роутером:

```
router bgp 5226
  neighbor 2.2.2.2 remote—as 5226
  !
  address—family ipv4 unicast
  no neighbor 2.2.2.2 activate
  exit
  !
  address—family ipv4 vpn
  neighbor 2.2.2.2 activate
  exit
```

Настройка анонсирования VPN маршрутов на PE роутере происходит в секции address-family ipv4/ipv6 unicast соответствующего VRF:

```
router bgp 5224 vrf 1
```

```
  address—family ipv4 unicast
label vpn export 527
rd vpn export 5227:1
rt vpn both 52:27
export vpn
import vpn
exit
```

Описание команд для анонсирования VPN маршрутов

1.

label vpn export <auto|label>

Назначение сервисной метки для определения принадлежность к конкретному VPN.

2.

rd vpn export <AA:NN>

Добавление к анонсируемому маршруту Route Distinguisher для различия маршрутов разных VPN. (В частности, чтобы анонсировать одинаковые префиксы сети от разных клиентов)

3.

rt vpn import <AA:NN>

Импорт Route Target - для определения к какому VFR относится маршрут

rt vpn export <AA:NN>

Экспортирование Route Target - для определения к какому VFR относится маршрут

rt vpn both <AA:NN>

Импорт и экспорт Route Target.

Таким образом на "отдающей" стороне назначается **rt vpn export,** а на "принимающей" должен быть соответствующий **rt vpn import.**

4.

export vpn

Экспорт маршрутов из address-family unicast в address-family vpn

5.

import vpn

Импорт маршрутов из address-family vpn в address-family unicast

51.4.1 Пример реализации MPLS-L3VPN

Ниже приведена схема, реализующая изоляцию сети C3PO от сети R2D2. Причем префиксы сетей у клиентов C3PO-1 и R2D2-1 одинаковые.

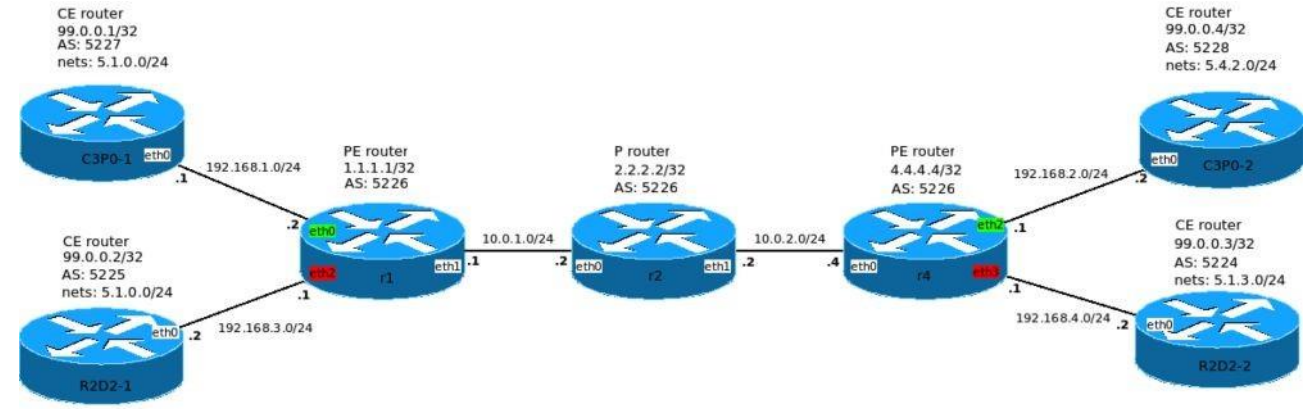

Рис. 51.2: Схема 2: MPLS-L3VPN

Примеры конфигураций узлов для данной схемы приведены в Приложении.

52. Криптография

52.1 Ключ доступа

Для начала работы с криптографическими средствами Dionis-NX необходимо инициализировать датчик случайных чисел (ДСЧ), а также создать ключ доступа (КД).

Начальное заполнение ДСЧ доставляется на внешнем носителе в одном из форматов:

- random.ini - формат хранения симметричных ключей Dionis (необходимы файлы gk.db3, uz.db3, random.ini);
- gk.db3 - формат хранения симметричных ключей Dionis, вариант ДСРФ (необходимы файлы gk.db3, uz.db3);
- PKCS#15 - формат хранения ключевой информации PKCS#15.

Ключ доступа используется для защиты секретов системы, хранящихся на внутреннем носителе, а именно:

- Начальное заполнение ДСЧ для следующей перезагрузки (защита шифрованием);
- Узлы замены для симметричных ключей Disec (защита шифрованием);
- Симметричные ключи Disec (защита шифрованием);

После генерации ключ доступа должен быть сохранён на внешнем носителе или в постоянной памяти LCD-индикатора. В случае сохранения на внешнем носителе, данный носитель потребуется при «холодном» перезапуске системы (poweroff/включение). В случае «тёплого» перезапуска (reboot) носитель с КД не потребуется, потому что КД также сохраняется в оперативной памяти LCD-индикатора.

ВАЖНО: В случае невозможности чтения КД с внешнего носителя из-за технической неисправности следует полностью очистить все данные на носителе без возможности восстановления (или полностью уничтожить носитель). Далее необходимо сгенерировать новый ключ доступа и заново установить в систему все ключи.

ВАЖНО: Утеря или компрометация КД означает компрометацию ключей, установленных в данную систему. В этом случае необходимо объявить соответствующие ключи Disec недействительными. Далее необходимо сгенерировать новый ключ доступа и установить в систему новые ключи.

Действуют следующие правила: [15]

15 Один ключ доступа может относиться только к одному узлу Dionis-NX;
- Один узел Dionis-NX может одновременно иметь только один ключ доступа;
- Ключ доступа может быть сохранён только один раз и на одном носителе (флэш, дискета, LCD, USB-токен/смарт-карта) после генерации или замены;
- На одном носителе может быть только один ключ доступа.

52.1.1 Состояние КД

Чтобы выяснить текущее состояние ключа доступа, нужно выполнить команду (в привилегированном режиме):

show crypto access key status

Возможные состояния:

- no key;
- not stored;
- ok.

no key - КД не сгенерирован или не загружен в систему. ДСЧ не инициализирован. Работа крипто-системы невозможна.

not stored - сгенерирован новый КД, но **не** сохранён на **внешнем** носителе. Необходимо выполнить команду'crypto access key store'. (См. ниже).

ok - КД загружен в систему. ДСЧ инициализирован.

52.1.2 Генерация нового КД и инициализация ДСЧ

Вставьте внешний носитель с начальным заполнением ДСЧ. (При необходимости начальное заполнение ДСЧ может быть сгенерировано с помощью команды "crypto random-ini create". См. раздел "Генерация начального заполнения ДСЧ" ниже).

Поддерживаемые внешние носители:

- floppyO - дискета;
- flashN[.N] - внешний флэш-носитель (пример: flashO, flash 1.2);
- token - USB-токен/смарт-карта.

Перейдите в привилегированный режим.

Если на внешнем носителе находится несколько контейнеров РКСБ#15, то можно выполнить команду просмотра содержимого внешнего носителя для выяснения имени контейнера:

show crypto access key random—inis flashO:

Данная команда выведет доступные контейнеры в корневой директории флэш-накопителя. Примерный формат вывода:

```
subdirl/
subdir2/

random.ini old—dionis
filel       pkcsl5
file2       pkcsl5
```

или

```
subdirl/
subdir2/


gk.db3        old—dionis—kb2
filel         pkcsl5
file2         pkcsl5
```

где subdirl, subdir2 - поддиректории; old-dionis, old-dionis-kb2, pkcsl5 - форматы контейнеров; random.ini, gk.db3, filel, file2 - имена файлов.

Для просмотра содержимого поддиректорий можно выполнять команды типа:

```
# show crypto access key random—inis flash0:/subdirl/subdir2/subdir3...
```

В случае использования USB-токена/смарт-карты потребуется ввести пин-код пользователя.

ВНИМАНИЕ: При многократном вводе ошибочного пин-кода смарт-карта может заблокироваться. Количество попыток ввода зависит от производителя и настройки смарт-карты.

Для генерации КД следует ввести команду:

```
# crypto access key init flashO:
```

Данная команда выполнит попытку чтения контейнера random.ini/gk.db3 (также необходим файл uz.db3).

Для считывания начального значения ДСЧ из контейнера РКСБ#15 необходимо выполнить команду с точным указанием имени файла контейнера. Например:

```
# crypto access key init flash0:/subdirl/file.pl5
```

Если контейнер PKCS#15 защищён паролем, то потребуется ввести пароль.

Для считывания начального значения ДСЧ из токена указывать имя РКСБ#15 файла не обязательно:

```
# crypto access key init token:
```

ВАЖНО: Если в системе уже присутствовал старый КД, то операция генерации нового КД перезапишет начальное заполнение ДСЧ, сохранённое на **внутреннем** носителе. Новое заполнение ДСЧ будет зашифровано на новом КД. Таким образом, данная операция является необратимой, и использование старого КД становится невозможным. После генерации нового КД необходимо также заново импортировать все необходимые секреты. Для плановой замены КД следует использовать команду crypto access key replace (см. ниже).

Кроме этого, в целях безопасности будет обновлено начальное заполнение ДСЧ на **внешнем** носителе.

Далее обязательно нужно выполнить пункт «Сохранение КД».

52.1.3 Сохранение КД

Если операция генерации прошла успешно, то ключ доступа находится в состоянии «not stored», и его необходимо сохранить на внешнем носителе **или** в постоянной памяти LCD-индикатора.

Для сохранения в ПЗУ LCD выполните команду:

```
# crypto access key store led
```

Для сохранения КД на внешнем носителе **извлеките** носитель с начальным заполнением ДСЧ, вставьте носитель для хранения ключа доступа и выполните команду:

```
# crypto access key store flashO
```

(или "floppyO" для сохранения на дискете).

ВАЖНО: Ключ доступа сохраняется в корневой директории на внешнем носителе в файле «acc-key». Если такой файл уже существует, то он будет заменён (с предварительным уведомлением).

Ключ доступа можно защитить паролем. При сохранении КД будет предложено ввести этот пароль. Если защита КД не требуется, то надо два раза нажать Enter. Если ключ защитить паролем, то его придётся вводить при каждом «холодном» перезапуске. При «тёплом» перезапуске пароль не требуется, так как в ОЗУ LCD ключ доступа сохраняется в открытом виде.

Сохранение КД на USB-токене/смарт-карте выполняется командой:

```
# crypto access key store token
```

Ключ доступа, сохранённый на токене, защищается пин-кодом пользователя токена.

Далее необходимо выполнить пункт «Загрузка КД».

52.1.4 Загрузка КД

Сначала необходимо добавить команду «crypto access key load» в сохранённую конфигурацию (startup-config), чтобы ключ доступа загружался каждый раз при перезапуске системы. Для этого нужно выполнить следующие команды:

```
# configure
(config)# crypto access key load
(config)# do copy running—config startup-config
```

Команда «crypto access key load» осуществляет поиск КД на внешних носителях в следующем порядке: все флэш-носители, дискета, ОЗУ LCD, ПЗУ LCD, USB-токен/смарт-карта. Если требуется иной порядок поиска КД, то можно явно указать список устройств. Например:

```
(config)# crypto access key load Icdram token flashO.l flash 1 Icdrom
```

Примечание: При загрузке КД из токена потребуется ввести пин-код пользователя.

При загрузке КД осуществляется попытка расшифровать начальное заполнение ДСЧ, сохранённое на внутреннем носителе. Неудачное расшифрование означает несоответствие данного

КД данному узлу Dionis-NX. При удачном расшифровании происходит инициализация ДСЧ, формируется новое начальное заполнение ДСЧ, которое зашифровывается на КД и помещается на внутренний носитель (для следующей перезагрузки).

Чтобы проверить, успешно ли загрузился КД, нужно выполнить команду «show crypto access key status» (см. выше).

Если по каким-то причинам требуется не осуществлять загрузку КД при перезапуске системы, то нужно удалить команду «crypto access key load» из сохранённой конфигурации, выполнив команды:

```
(config)# no crypto access key load
(config)# do copy running—config startup—config
```

52.1.5 Удаление КД

Если по каким-то причинам требуется удалить КД из **оперативной памяти** системы, следует выполнить команду:

```
#  crypto access key clear memory
```

Данная команда удаляет ключ доступа из памяти системы и из ОЗУ LCD.

Следующие команды удаляют КД с **внешних носителей** (соответственно ПЗУ LCD, флэш, дискета, токен/смарт-карта):

```
#  crypto access key clear led
#  crypto access key clear flashO
#  crypto access key clear floppyO
#  crypto access key clear token
```

Для **безопасного** удаления ключей с внешних носителей рекомендуется использовать команду "clear removable" (см. раздел "Обслуживание").

52.1.6 Плановая замена КД

Для замены ключа доступа необходимо, чтобы старый ключ доступа был загружен в память (см. «crypto access key load» выше).

Для замены КД в ПЗУ LCD-индикатора выполните команду:

```
#  crypto access key replace led
```

Для замены КД с сохранением на внешнем носителе вставьте носитель со старым КД и выполните команду:

```
#  crypto access key replace flashO
```

(или floppy0 для дискеты, token для USB-токена/смарт-карты).

В случае использования токена КД защищается пин-кодом пользователя токена.

В случае использования флэш/дискеты будет предложено защитить новый КД паролем. Если защита не нужна, нажмите 2 раза Enter.

При замене КД перешифровываются все секреты системы на новом КД.

Если по каким-то причинам замена КД не удалась, то осуществляется откат, и старый КД остаётся в силе.

52.1.7 Генерация начального заполнения ДСЧ

Команда «crypto random-ini create» создаёт начальное заполнение датчика случайных чисел и сохраняет его на внешний носитель. Далее данное начальное заполнение может быть использовано для инициализации крипто-системы с помощью команды «crypto access keyinit».

Формат команды:

crypto random —ini create pkcsl5 <внешний_носитель>

где: <носитель> - внешний носитель («flashO:», «flash 1.1:», «floppy0:», и т. д.).

Данная команда создаёт в корневой директории внешнего носителя РКСБ15-контейнер с именем «random-ini.p15», содержащий начальное заполнение ДСЧ. Если файл с таким именем уже существует, то он перезаписывается.

52.1.8 Плановое обновление начального заполнения ДСЧ

Для обеспечения качества псевдослучайных последовательностей, выдаваемых программным ДСЧ, периодически необходимо инициализировать ДСЧ повторно с помощью нового начального заполнения, доставляемого на внешнем носителе. Периоды повторной инициализации ДСЧ определяются регламентом.

Повторная инициализация ДСЧ внешним начальным заполнением осуществляется командой:

crypto random update <носитель>[:/<путь_к_файлу_рскз15>]

Начальное значение ДСЧ может доставляться в форматах "random.ini", ДСРФ, PKCS15 (см. выше). (См. также описание команд 'crypto access key init' и 'show crypto access key random-inis' выше).

52.2 Туннели Disec

Dionis-NX имеет возможность создавать туннели DISEC.

52.2.1 Инициализация DISEC

Перед созданием туннеля необходимо инициализировать подсистему DISEC.

Перед инициализацией подсистемы DISEC необходимо загрузить ключ доступа (подробнее см. Криптография. Ключ доступа):

```
#  crypto access key init flashO
# crypto access key store led
(config)# crypto access key load
```

Первой командой производится инициализация ключа доступа (КД), данные для КД считываются с внешнего ключевого носителя (ВКН), в данном случае с флэшки. Второй командой КД сохраняется в памяти LCD. И, наконец, третья команда определяет, что нужно каждый раз при загрузке системы считывать КД с LCD, куда он ранее был сохранен (см.вторую команду).

После того, как КД был успешно проинициализироваван, можно импортировать ключ DISEC:

```
#  crypto disec import key flashO:/
```

Если команда успешно выполнилась, будет выведена информация об импортированном ключе:

```
Info: key (serial:55; cn:l) successfully imported
```

В данном случае ключ серии 55 с локальным крипто-номером 1 был успешно импортирован при инициализации DISEC.

Можно импортировать ключи, указав путь хранения ключей на конкретном устройстве, например:

```
#  crypto disec import key flashO.l:/path/to/keys
```

Также можно осуществить множественный импорт ключей, указав путь базовой директории на устройстве, например:

```
#  crypto disec import key flashO. l:/keys/16
```

Команда произведет импорт ключей (если они есть), находящихся в директориях, которые находятся в директории /keys на устройстве flashO. 1. Директории с именами km_k,dbl,db2 при этом будут пропущены.

52.2.2 Создание туннеля

Перед описанием процесса создания туннеля DISEC (далее туннеля) введем необходимые понятия, и сопроводим их краткими описаниями:

1. имя туннеля: задает имя туннеля, нужно для идентификации туннеля и для большей на-

---

16 local: задает IP-адрес локального конца туннеля;

- remote: задает IP-адрес удаленного конца туннеля;
- id: целое число (до 5 цифр), идентифицирующее туннель; значение этого параметра должно совпадать на обоих концах туннеля;
- serial: номер серии ключей - целое десятичное число, равное номеру серии ключей, используемой в данной криптографической сети;
- local-cn: целое число (до 5 цифр), равное номеру данного узла в криптографической сети;

глядности;

2. параметры туннеля:

    • IP-адреса концов туннеля;

- криптографические параметры туннеля;

3. правила отбора в туннель: задают правила для исходящего сетевого трафика, по которым он попадает в туннель;
4. приоритет правил отбора:
    - чем ниже численно приоритет правиладем выше в списке правил стоит данное правило;
    - правила просматриваются сверху вниз;
    - при попадании исходящей датаграммы в правило, просмотр нижележащих правил данного туннеля прекращается: датаграмма попала в туннель;
5. приоритет туннеля:
    - определяет приоритет набора правил туннеля;
    - чем ниже численно приоритет туннелядем выше в списке туннелей стоит данной туннель;
    - наборы правил туннелей просматриваются сверху вних;
    - при попадании исходящей датаграммы в одно из правил в наборе правил туннеля, просмотр нижележащих туннелей прекращается.

Будем считать, что DISEC инициализирован с поддержкой криптографии. Для создания туннеля следует выполнить команды:

```
(config)# crypto disec conn tl
(config—disec—tl)# local 1.1.1.1
(config—disec—tl)# remote 2.2.2.2
(config—disec—tl)# id 1
(config—disec—tl)# serial 55
(config—disec—tl)# local—cn 1
(config—disec—tl)# remote—cn 2
(config—disec—tl)# alg both
```

Туннель добавится в конец списка уже имеющихся туннелей под очередным номером.

Если необходимо поместить туннель в список под другим номером, при создании туннеля можно использовать следующую команду:

```
(config)# 1 crypto disec conn t2
```

Данная команда создаст туннель t2 под номером 1. Ранее созданный туннель tl переместится ниже под номером 2.

Рассмотрим по порядку параметры туннеля, которые необходимо указать при его создании: ·

- remote-cn: целое число (до 5 цифр), равное номеру в криптографической сети того узла, с которым будет выполняться обмен информацией по данному туннелю;
- alg both: алгоритм трансформации данных в туннеле; возможные значения:
  - compression: только сжатие данных;
  - encryption: только зашифрование данных;
  - both: и сжатие, и зашифрование данных;
  - попе: никакой трансформации данных не производится.

Чтобы удалить ненужный более туннель, следует использовать команду:

(config)# no crypto disec conn tl

Этой командой удаляется туннель tl.

Чтобы удалить все туннели, следует использовать команду:

(config)# no crypto disec conn *

Настройка IPv6-туннелей аналогична настройке IPv4-туннелей.

Создание IPv6-туннеля и редактирование его параметров:

(config)# 1 crypto disec соппб tlv6 (config—disec—
соппб—tlv6)# local 2001:db8:: 10:10 (config—disec—
соппб—tlv6)# remote 2001:db8:: 12:11 (config—
disec—соппб—tlv6)# id 2
(config—disec—соппб—tlv6)# serial 33
(config—disec—соппб—tlv6)# local—cn 3
(config—disec—соппб—tlv6)# remote—cn 2
(config—disec—соппб—tlv6)# alg both

Удаление IPv6 туннеля:

(config)# no crypto disec соппб tlv6

52.2.3 Правила отбора туннеля

Правила отбора определяют, какой именно сетевой трафик будет попадать в туннель. Правила отбора туннеля просматриваются сверху вниз: самое приоритетное правило имеет номер 1, следующее правило имеет номер 2 и т.д. Как только для данной датаграммы будет найдено соответствующее правило отбора, принимается решение инкапсулировать датаграмму в туннель. Важную роль в том, в какой именно туннель попадет датаграмма, играет приоритет туннеля. Чем ниже номер туннелядем более приоритетен набор правил отбора данного туннеля по отношению к наборам других туннелей. Рассмотрим пример:

1 tunnell
  1 rulela
  2 rulelb
2 tunnel2
  1 rule2a

2 rule2b
3 rule2c

В данном примере условными обозначениями определено 2 туннеля tunnel1 и tunnel2, в каждом из которых свой набор правил. При принятии решения, каким именно туннелем пересылать исходящую датаграмму и нужно ли ее вообще пересылать каким-либо туннелем, правила отбора анализируется в следующем порядке: rulela, rulelb, rule2a, rule2b, rule2c.

При попадании в одно из первых двух правил датаграмма будет пересылаться через tunnel1, при попадании в одно из следующих трех правил - через tunnel2, иначе - датаграмма не будет пересылаться через эти DISEC-тунели.

Теперь перейдем непосредственно к созданию правил отбора.

Формат задания правила следующий: **permit|deny [PROTO] src <SRCIP> dst <DSTIP> [sport <S1> [S2]] [dport <D1> [D2]] [remark <REMARK>]**

Рассмотрим аргументы команды задания правила:

- permit - разрешающее правило: трафик, попадающий в правило, будет обрабатываться данным туннелем;
- deny - исключающее правило: трафик, попадающий в правило, не будет обрабатываться данным туннелем;
- PROTO - протокол, например IP,TCP,UDP,ICMP и др. или номер протокола; по-умолчанию: протокол IP;
- SRCIP - адрес сети или узла отправителя датаграммы;
- DSTIP - адрес сети или узла получателя датаграммы;
- S1,S2 - начальный и, возможно,конечный порт отправителя датаграммы; если указан S2, то значения S1,S2 задают интервал портов; по-умолчанию: любой порт;
- D1,D2 - начальный и,возможно,конечный порт получателя датаграммы; если указан S2, то значения S1,S2 задают интервал портов; по-умолчанию: любой порт;
- REMARK - необязательная пометка(любая строка без пробелов) правила.

Порты S1/S2,D1/D2 можно задать, только если PROTO указывает на tcp- или udp-протокол.

Примеры:

```
(crypto—disec—tl)# permit tcp src 1.1.1.0/24 dst 192.168.2.2/32 sport 20 80 dport 100 200
        remark GOOD
(crypto—disec—tl)# deny src 2.2.0.0/16 dst 192.168.2.2/32 remark BETTER
(crypto—disec—tl)# 1 permit icmp src 2.2.0.0/16 dst 192.168.2.2/32 remark BEST
(crypto—disec—tl)# no 2
(crypto—disec—tl)# no all
```

Первые три команды задают правила отбора. Четвертая команда удаляет правило под номером 2, помеченное как GOOD. Пятая команда удаляет все правила.

Логика настройки правил отбора IPv6-туннелей практически идентична настройке IPv4-туннелей за исключением используемых адресов для параметров "src" и "dst" (используются IPv6-адреса).

52.2.4 Включение и выключение туннеля

Вышеописанные параметры и правила отбора в туннель не будут действовать, пока вы не включите туннель:

(config—disec)# crypto disec enable conn t1

Если пристутсвуют лицензионные ограничения на количество включенных туннелей, вы не сможите включить больше определенного числа туннелей. Узнать число доступных к включению туннелей можно по команде show crypto disec license.

Когда туннель включен вы можете менять его параметры и правила отбора. Если эти параметры имеют смысл, они будут автоматически применены к данному туннелю.

Чтобы выключить туннель, следует использовать команду:

(config—disec)# crypto disec disable conn t1

Чтобы включить все туннели, следует использовать команду:

(config—disec)# crypto disec enable conn *

Чтобы выключить все туннели, следует использовать команду:

(config—disec)# crypto disec disable conn *

В случае, если какой-либо туннель А не сможет отключиться, будет предпринята попытка включения успешно отключенных туннелей, если таковые были до момента сбоя в отключении туннеля А. Таким образом, в случае успешности данной команды ВСЕ туннели будут отключены. В случае неуспешности - ничего не изменится.

Чтобы включить определенные туннели, заданные по ID выполните команду:

(config—disec)# crypto disec enable id IDSTR

Параметр IDSTR имеет формат <N|N-M>[,{N|N-M}] , т.е. это набор из 1 или более подстрок, разделенных запятыми, где каждая подстрока - это число (ID туннеля) или интервал чисел (ID туннелей),заданный через дефис, причем левая граница интервала должна быть меньше либо равна правой. Неверные подстроки пропускаются.

Пример: команда crypto disec enable id 1,4-7,9-9,23-32,„5 - попытается включить туннели со следующими ID: 1,4,5,6,7,9, если указанные ID заданы в туннелях.

Чтобы выключить определенные туннели, заданные по ID выполните команду:

(config—disec)# crypto disec disable id IDSTR

Включение и выключение IPv6-туннелей идентичны тем же операциям с IPv4-туннелями:

(config—disec)# crypto disec епаЬleб id IDSTR

и

(config—disec)# crypto disec disable id IDSTR

52.2.5 Копирование и перемещение туннеля

crypto disec copy <OLD> <NEW> [id <ID>] [PRF] [force] [rules]

Данная команда осуществляет копирование существующего туннеля в новый вместе со всеми параметрами и, возможно, правилами отбора. Новый туннель в случае успешного копирования будет находиться в состоянии **выключен.**

Параметры:

- OLD : старое имя туннеля;
- NEW : новое имя туннеля;
- ID : id, присваиваемый новому туннелю;
- PRF : приоритет, под которым следует создать новый туннель;
- force : если туннель NEW уже существует, он будет отключен и его параметры станут равными параметрам туннеля OLD, т.е. произойдет замена туннеля;
- rules : копировать также и правила туннеля.

crypto disec move <NAME> <PRF>

Данная команда осуществляет перемещение существующего туннеля NAME: туннелю присваивается новый приоритет PRF. Состояние туннеля (включен/выключен) сохраняется.

Для IPv6-туннелей команда будет иметь вид:

crypto disec move6 <NAME> <PRF>

52.2.6 Работа с ключами

В данном разделе описывается просмотр, удаление и добавление ключей и абонентов.

52.2.6.1 Генерация ключей

Команда «crypto matrix create» используется для генерации матрицы симметричных ключей и записи её на внешний носитель. Матрица записывается в формате, пригодном к использованию в туннелях disec.

Формат команды:

crypto matrix create <серия> <размер_матрицы> <носитель>

где:

- <серия> - номер серии ключей;
- <размер_матрицы> - количество абонентов;
- <носитель> - внешний носитель («flashO:», «flash 1.1:», «floppyO:», и т. д.).

После генерации и записи на внешний носитель матрица ключей удаляется из оперативной памяти.

52.2.6.1.1 Формат матрицы на внешнем носителе В корневой директории носителя создаётся файл «series», содержащий в текстовом виде серию ключей, а также директории вида «1», «2», «3», ..., «NNN», где NNN - размер матрицы. Каждая директория «1\1» содержит строчку из матрицы ключей для соответствующего абонента N. Содержимое директории «N» должно быть записано в корневую директорию другого внешнего носителя, который будет передан соответствующему абоненту disec. Содержимое директории «N» имеет формат, пригодный для импортирования строчки ключей с помощью команды «crypto disec import key» (содержит файлы «gk.db3», «random.ini», «uz.db3» и директорию «km_k» с ключевой информацией).

ВНИМАНИЕ: Перед записью матрицы ключей команда «crypto create matrix» удаляет с внешнего носителя **всё** старое содержимое.

52.2.6.2 Просмотр ключей

Чтобы посмотреть установленные в систему крипто-ключи, следует использовать команду:

```
#  show crypto disec keys
```

Пример вывода команды:

```
Installed keys:
Serial: 55 , locals: 1
```

Из вывода следует, что установлен один ключ с локальным криптономером 1 серии 55.

Чтобы посмотреть установленные в систему крипто-ключи определенной серии, следует использовать команду:

```
#  show crypto disec key 55
```

Пример вывода команды:

```
Installed keys for serial 55: 1
```

Из вывода следует, что установлен один ключ с локальным криптономером 1 для серии ключей 55.

52.2.6.3 Просмотр абонентов

Чтобы посмотреть доступность абонента с крипто-номером 2 для доступа по ключу с серией 55 и крипто-номером 1, следует выполнить команду:

```
# show crypto disec abonent 55 1 2
```

Если абонент доступен, выдача команды будет следующей:

```
Access to abonent 2 for key (sn = 55;loc=1), check status: GRANTED
```

Если абонент не доступен, выдача команды будет следующей:

```
Access to abonent 2 for key (sn = 55;loc=1), check status: DENIED
```

52.2.6.4 Добавление и удаление ключей

В инициализированную подсистему DISEC вы можете добавлять новые и удалять старые крипто-ключи.

Чтобы добавить новый ключ, нужно вставить ВКН, например, флэшку, хранящую новый крипто-ключ, и выполнить команду:

```
# crypto disec import key flash0:/
```

Чтобы удалить установленный ключ, следует использовать команду:

```
# crypto disec remove key 55 1
```

Команда удалит ключ 1 серии 55.

Чтобы удалить все установленные ключи, следует использовать команду:

```
# crypto disec remove key all
```

52.2.6.5 Полное удаление DISEC

Полное удаление из системы всей информации, относящейся к DISEC, состоит в следующих шагах:

- удаление туннелей командой no crypto disec conn <NAME> (режим configure);
- удаление ключей командой: crypto disec remove key <SERIAL> <LOCAL> (режим enable);
- окончательная очистка от DISEC: crypto disec cleanup (режим enable).

Примечание: Для безопасного удаления ключей с внешних носителей рекомендуется использовать команду "clear removable" (см. раздел "Обслуживание").

52.2.6.6 Плановая смена ключей

При плановой смене ключей (далее ПС) производится замена сетевых ключей одной из серий на ключи новой серии, которая может либо уже иметься в системе, либо быть импортирована с внешнего носителя.

ПС должна быть выполнена на всех узлах, входящих в криптографическую сеть с данной серией ключей.

Рассмотрим процесс ПС на одном из концов туннеля на примере.

Предположим, что у нас имеются следующие входные данные: [17]

---

17 туннель Disec с именем: TUN;

- туннель TUN использует серию ключей с номером SER0;
- туннель TUN использует локальный криптономер LOCO;
- туннель TUN использует удаленный криптономер REM0;
- туннель TUN может быть включен или выключен.

Задача: для туннеля TUN осуществить ПС:

- заменить серию ключей SER0 на серию ключей SER1;
- криптономер LOC0 на LOC1;
- криптономер REM0 на REM1;
- новые криптономера LOC1 и/или REM1 могут остаться прежними, в этом случае LOC1 = LOC0 и/или REM1 = REM0.

В данном случае алгоритм смены ключей следующий:

1. В случае, если производится удаленная ПС, то необходимо войти в систему, на которой нужно осуществить ПС, по защищенному туннелю. Например, если это туннель Disec, то нужно иметь для этого туннеля специальную серию ключей, используемую только для процедуры ПС.
2. В случае, если производится локальная ПС, то защищенный туннель для ПС не нужен - администратор просто заходит в систему локально, используя свое имя пользователя и пароль.
3. После входа в систему (локально или удаленно), необходимо удостовериться в наличии новой серии ключей SER1, выполнив следующую команду:

```
DionisNX# show crypto disec key SERI
```

Выдача команды покажет, какие локальные криптономера доступны для данной серии ключей. Необходимо удостовериться, что новый локальный криптономер LOCI туннеля TUN перечислен в выдаче данной команды.

Если LOCI не найден в выдаче команды, значит ПС не будет выполнена успешно.

4. Далее необходимо проверить, возможен ли доступ по данной серии ключей к нужному удаленному абоненту REM1, выполнив следующую команду:

```
DionisNX# show crypto disec abonent SERI LOC1 REM1
```

Если удаленный абонент является доступным по данной серии ключей, то выдача данной команды будет следующей:

```
Access to abonent REM1 for key (sn = SERl;loc=LOC1): GRANTED.
```

Если удаленный абонент является недоступным по данной серии ключей, то выдача данной команды будет следующей:

```
Access to abonent REM1 for key (sn = SERl;loc=LOC1): DENIED.
```

Если удаленный абонент недоступен (команда выдала DENIED), значит ПС не будет выполнена успешно.

5. Затем следует войти в настройки туннеля TUN и выполнить следующие команды:

```
DionisNX# configure
DionisNX(config)# crypto disec conn TUN
DionisNX(config—disec—TUN)# serial SERI
```

Если криптономер LOCI не равен LOC0, то дополнительно необходимо будет выполнить следующую команду:

```
DionisNX(config—disec—TUN)# local—cn LOC1
```

Если криптономер REM1 не равен REM0, то дополнительно необходимо будет выполнить следующую команду:

```
DionisNX(config—disec—TUN)# remote—cn REM1
```

6. На этом плановая смена ключей завершена. Пункты 3 и 4 алгоритма не обязательны и нужны исключительно для проверки того, что туннель будет корректно работать на новой серии ключей.

После проверки связи со всеми удаленными узлами криптографической сети, необходимо удалить старую серию ключей следующей командой:

```
DionisNX# crypto disec remove key SER0 LOC
```

Эту команду следует повторить для каждого локального криптономера в серии SER0 до полного удаления старой серии ключей SER0 из системы.

52.2.6.7 Действия при неплановой смене ключа доступа

Если ключ доступа после импорта ключей был изменен способом, отличным от «Плановой замены КД», то для корректной работы туннелей необходимо: [18]

---

18 удалить установленные ключи;

- выполнить crypto disec cleanup;
- вновь осуществить импорт нужных ключей.

52.2.6.8 Удаление абонентов

Чтобы заблокировать возможность криптографической связи с абонентом, определяемым удаленным крипто-номером, следует удалить его из локального ключа:

```
# crypto disec remove abonent 55 1 10
```

Команда навсегда блокирует абонента 10 для ключа 1 серии 55. Чтобы разблокировать абонента, необходимо удалить и снова добавить ключ 1 серии 55 с ВКН.

52.2.7 Работа с туннелями

52.2.7.1 Просмотр туннелей

Чтобы посмотреть таблицу имеющихся туннелей, следует использовать команду:

```
# show crypto disec conns
```

Пример выдачи команды:

```
[#]NAME          ID   SRC           DST SN                LOC REM   A B
tl               4    192.168.2.1   192.168.2.2 -          —   —     N
#t2              6    192.168.4.1   192.168.4.2 55         1   2     E N
```

Рассмотрим столбцы таблицы:

- [#]NAME - имя туннеля; если перед именем стоит знак #, значит туннель выключен;
- ID - идентификатор туннеля;
- SRC - адрес локального конца туннеля;
- DST - адрес удаленного конца туннеля;
- SN - номер серии ключей;
- LOC - локальный крипто-номер туннеля;
- REM - удаленный крипто-номер туннеля;
- A - тип трансформации данных в туннеле: E - шифрование, C - компрессия, B - шифрование и компрессия, N - нет трансформации;
- B - туннель заблокирован: У - да, N - нет.

Чтобы посмотреть информацию по конкретному туннелю, следует использовать команду:

```
#   show crypto disec conn t1
```

и для IPv6-туннелей:

```
#   show crypto disec conn6 t1
```

В дополнение к информации, которая была рассмотрена для предыдущей команды, будут выведены правила данного туннеля, например:

```
[#]NAME ID SRC DST SN                                  LOC REM A B
idl 11 192.168.0.5 192.168.0.4 1                         2 3 E N
1 permit src 0.0.0.0/0 dst 192.168.32.0/24
2 permit src 0.0.0.0/0 dst 192.168.16.0/24
3 permit src 0.0.0.0/0 dst 192.168.3.0/24
```

```
[#]NAME ID SRC      DST
        SN      LOC     REM
A B
idl     11      192.168.0.5
                192.168.0.4     1
2       3 EN
```

```
1  permit src 0.0.0.0/0 dst 192.168.32.0/24
2  permit src 0.0.0.0/0 dst 192.168.16.0/24
3  permit src 0.0.0.0/0 dst 192.168.3.0/24
```

52.2.7.2 Просмотр правил отбора

Чтобы посмотреть информацию по конкретному туннелю, следует использовать команду:

```
# show crypto disec rules id1
```

Пример выдачи команды:

```
1  permit src 0.0.0.0/0 dst 192.168.32.0/24
2  permit src 0.0.0.0/0 dst 192.168.16.0/24
3  permit src 0.0.0.0/0 dst 192.168.3.0/24
```

52.2.7.1 Блокирование туннеля

Иногда бывает необходимо полностью заблокировать трафик через включенный туннель. Это делается следующей командой:

```
(config—disec—tl)# block
```

Чтоб разблокировать трафик через туннель, следует использовать команду:

```
(config—disec—tl)# no block
```

52.2.7 Прочая работа с DISEC

52.2.8.1 Привязка процедур шифрования к работе на определенных ядрах процессора

Для балансировки шифрования по ядрам процессора используйте команду:

```
(config)# crypto disec affinity 0—1,3
```

где в качестве параметров выступает сумма диапазонов и номеров ядер процессора.

52.2.8.2 Синхронный и асинхронный режим обработки пакетов

По умолчанию ядро пытается распараллеливать обработку входящих и исходящих DISEC-пакетов. Зашифрование/дешифрование пакетов производится на разных процессорах/ядрах. В результате пакеты могут отправляться/попадать в систему фактически не в том порядке, в котором они были отправлены/получены. Если такое поведение неприемлемо, то можно включить синхронный режим обработки:

```
(config)# crypto disec mode sync
```

Вернуть поведение по умолчанию (асинхронный режим) можно с помощью команды:

```
(config)# no crypto disec mode
```

Или:

```
(config)# crypto disec mode async
```

52.2.8.3 Инкапсуляция DISEC в UDP

Для инкапсуляции датаграмм DISEC в UDP-датаграммы в дополнение к основным параметрам туннеля следует использовать следующую команду:

(config—disec—tl)# encap sport 500 dport 600

Параметры sport и dport - номер UDP-порта отправителя и получателя, соответственно, они не обязательны. По умолчанию равны 500.

52.2.8.4 Блокирование DISEC трафика

Чтоб полностью заблокировать трафик через все включенные туннели, следует использовать команду:

(config—disec)# crypto disec block

Чтоб разблокировать трафик через все включенные туннели, следует использовать команду:

(config—disec)# no crypto disec block

Трафик будет разблокирован только для тех туннелей, которые не заблокированны индивидуально командой **block.**

52.2.8.5 Просмотр состояния

Чтобы посмотреть версию подсистемы DISEC, следует использовать команду:

show crypto disec version

Чтобы посмотреть статистику обработки пакетов через подсистему DISEC, следует использовать команду:

show crypto disec statistic

и для 1Ру6-туннелей:

show crypto disec statistic6

Рассмотрим пример выдачи данной команды:

TUNNEL	XP_OUT XP_IN XP_FWD XS_OUT XS	_IN
tun 1	17:49:20 17:49:20 17:49:20 1235557106676	108258340540
tun2	17:49:20 17:49:20 17:49:20 1302550192329	609146127844

Рассмотрим столбцы данной таблицы:

- TUNNEL : имя туннеля;
- XP_OUT : последнее время попадания исходящей датаграммы в туннель;
- XP_FWD : последнее время попадания в туннель входящей датаграммы, не предназначенной для текущей системы;
- XP_IN : последнее время попадания в туннель входящей датаграммы, предназначенной для текущей системы;

- XS_OUT : число байт исходящего трафика, прошедшего через туннель;
- XS_IN : число байт входящего трафика, прошедшего через туннель.

Чтобы посмотреть лицензионные ограничения на подсистему DISEC, следует использовать команду:

```
# show crypto disec license
```

В настоящее время поддерживается только лицензионное ограничение на количество включенных туннелей DISEC.

52.2.8.3 Режим отладки

При возникновении проблем в работе каких-либо команд подсистемы DISEC можно включить режим отладочных сообщений:

```
(config—disec)# crypto disec debug
```

Более глубокий режим отладки можно включить при помощи команды:

```
(config—disec)# crypto disec debug trace
```

52.3 Туннельные интерфейсы Ditun (ditun, ditap, ip6ditun, ip6ditap)

Туннельные интерфейсы Ditun, позволяют создавать криптографически защищенные каналы связи, трафик в которые отбирается по правилам маршрутизации, в отличии от туннельных соединений Disec, трафик в которые отбирается по правилам отбора.

52.3.1 Создание интерфейса

Внимание! Для включения в интерфейсе ditun режима шифрования в Dionis-NX должен быть предварительно создан ключ доступа и загружены ключи абонентов Disec (см. Туннели Disec, работа с ключами). Для работы интерфейса в открытом режиме (без шифрования) это не является необходимым.

Для создания интерфейса, из режима конфигурации выполните команду:

```
(config)# interface ditun 0
```

В качестве номера интерфейса (в примере - 0) может быть выбрано любое число (натуральное или 0). После этого, в режиме конфигурации интерфейса необходимо задать идентификатор туннеля (id), локальный (local) и удаленный (remote) IP-адреса - концы туннеля, например:

```
(config — if— ditunO)# id 1
(config —if—ditunO)# local 1.1.1.1
(config —if—ditunO)# remote 2.2.2.2
(config—if—ditunO)# enable
```

Параметр enable - делает созданный интерфейс активным.

При использовании схемы Disec необходимо дополнительно ввести параметры аналогичные параметрам туннелей Disec, алгоритм (alg), номер серии ключей (serial), локальный (local-cn) и удаленный (remote-cn) крипто-номера, например:

```
(config—if—ditun0)# id 1
(config—if—ditun0)# alg encrypt
(config—if—ditun0)# local 1.1.1.1
(config—if—ditun0)# remote 2.2.1.2
(config—if—ditun0)# serial 1
(config—if—ditun0)# local—cn 1
(config—if—ditun0)# remote—cn 1
(config—if—ditun0)# enable
```

Интерфейс будет создан в тот момент, когда будет задана минимально необходимая информация.

Для создания IPv6-интерфейса, из режима конфигурации выполните команду:

```
(config)# interface ip6ditun 0
```

Настройка IPv6-интерфейсов аналогична настройке IPv4-интерфейсов за исключением используемых адресов (используются IPv6-адреса).

```
(config—if— ip6ditun0)# id 10 (config—if—
ip6ditun0)# alg encrypt (config—if—
ip6ditun0)# local 2001:db8::2 (config—if—
ip6ditun0)# remote 2001:db8::3 (config—
if—ip6ditun0)# serial 7
(config—if—ip6ditun0)# local—cn 1
(config—if—ip6ditun0)# remote—cn 3
(config—if—ip6ditun0)# enable
```

Для удаления IPv6-интерфейса, из режима конфигурации выполните команду:

```
(config)# no interface ip6ditun 0
```

52.3.2 Настройка интерфейса

Настройка автоматического назначения идентификатора туннеля. Возможно использовать автоматическую генерацию id туннеля, с помощью команды:

```
(config —if—ditunO)# id auto
```

При этом, в качестве id будет выбран:

Номер интерфейса (если интерфейс не в режиме шифрования);
Локальный крипто-номер + удаленный крипто-номер.

Какой именно id туннеля используются можно узнать с помощью команды show interface ditun.

Настройка инкапсуляции UDP. Возможно использовать инкапсуляцию в UDP, если провайдер не пропускает протокол инкапсуляции IP в IP. Для создания туннеля с инкапсуляцией в протокол UDP, необходимо выполнить команду encap, с указанием портов концов туннеля (отправитель и получатель), например:

```
(config — if— ditun0)# encap 505 505
```

Значение удаленного порта можно не указывать, если он совпадает с портом источника.

Настройка NAT-traversal. Возможно работать через NAT, когда адрес/порт удаленного конца туннеля заранее неизвестен. Для этого, на одном конце туннеля можно указать значение "any" для параметров удаленного порта и удаленного адреса. В этом случае, туннельный интерфейс будет ожидать входящий пакет от любого IP-адреса с любого порта. Значения порта и адреса, полученные из первого входящего пакета, будут автоматически применены и интерфейс сможет передавать данные. Пример:

```
(config — if— ditun0)# remote any
(config —if—ditun0)# encap 505 any
```

Какой именно IP адрес и порт используются при отправке датаграмм можно узнать с помощью команды show interface ditun.

Настройка keepalive. Интерфейсы ditun поддерживают механизм пинг-проб, при этом, отсутствие ответной пробы меняет состояние несущей и переводит интерфейс в режим no-carrier. Настройки проб аналогичны настройкам проб в GRE туннелях, но имеют небольшие отличия, например:

```
(config —if—ditun0)# keepalive 5 0 src 10.0.0.1 dst 10.0.0.2
(config—if—ditun0)# link—detect
```

В данном примере src и dst устанавливают IP-адрес источника и IP-адрес назначения пинг-проб и являются опциональными. Если они не указаны явно, IP-адресам источника и назначения присваиваются значения IP-адресов концов туннеля.

Дополнительные настройки. Возможно задать IP-адрес (или несколько адресов) интерфейсу ditun и использовать его в маршрутизации/NAT/фильтрах, а также делать другие действия, которые являются допустимы по отношению к интерфейсу, например:

```
(config —if—ditun0)# ip address 10.0.0.1/24
(config —if—ditun0)# ttl 32 (config—if—
ditun0)# ip access—group wan in (config —if—
ditun0)# do show interface ditun 0
(config —if—ditun0)# do tcpdump ditun0 numeric
```

Как и в случае с любыми другими интерфейсами, вы можете объединять ditun интерфейсы в bond интерфейс, настраивать политику QoS, применять NAT и фильтры и т.д.

Вы можете менять режим работы обработки пакетов (асинхронный/синхронный) с помощью команды: crypto disec mode, как описано в главе "Туннели Disec".

Настройка IPv6-интерфейсов аналогична настройке IPv4-интерфейсов за исключением используемых адресов (используются IPv6-адреса).

52.3.3 Маршрутизация

Для направления трафика в туннель используются правила маршрутизации. Правила могут быть заданы как и относительно самого интерфейса, так и относительно адреса удаленного интерфейса ditun (не путать с адресом удаленного конца туннеля!), например:

(config)# ip route default ditun 0
(config)# ip route 192.168.0.0/24 10.0.0.2

При этом во втором случае (маршрутизация через адрес удаленного ditun интерфейса) и примененных настройках: keepalive и link-detect, в случае недостижении пинг пробами удаленной стороны, маршрут будет динамически деактивирован (а при возобновлении прохождения проб - активирован снова).

При нахождении ditun интерфейса в составе bond интерфейса, попадание трафика в туннель определяется правилами маршрутизации для bond интерфейса и режимом работы bond интерфейса.

52.3.4 Удаление интерфейсов

Для удаления интерфейса выполните команду no interface ditun, например:

(config)# no interface ditun 0

52.3.5 Интерфейсы ditap

Интерфейсы ditap/ip6ditap в отличии от ditun/ip6ditun работают не на сетевом, а на канальном уровне модели OSI. Имеют свой MAC-адрес и обладают возможностями реального сетевого интерфейса. Настройка ditap/ip6ditap полностью аналогична настройке ditun/ip6ditun интерфейсов. Например:

(config—if— ip6ditap0)# id 3
(config—if— ip6ditap0)# alg encrypt
(config—if— ip6ditap0)# local 2001:db8:: 10:9
(config—if— ip6ditap0)# remote 2001:db8:: 10:3
(config—if—ip6ditap0)# serial 1
(config—if— ip6ditap0)# local—cn 2
(config—if— ip6ditap0)# remote—cn 2
(config—if—ip6ditap0)# enable

53. VRRP-кластер

VRRP (Virtual Router Redundancy Protocol) — сетевой протокол, предназначенный для увеличения доступности маршрутизаторов, выполняющих роль шлюза по умолчанию. Это достигается путём объединения группы маршрутизаторов в один виртуальный маршрутизатор и назначения им общего IP-адреса, который и будет использоваться как шлюз по умолчанию для компьютеров в сети.

53.1 Основные понятия

- VRRP-маршрутизатор (VRRP Router) — маршрутизатор, на котором работает протокол VRRP. Он может участвовать в одном или более виртуальных маршрутизаторах;
- Виртуальный маршрутизатор (Virtual Router, VR) — абстрактный объект, которым управляет VRRP. Выполняет роль маршрутизатора по умолчанию для компьютеров в сети. Фактически, виртуальный маршрутизатор — это группа интерфейсов маршрутизаторов, которые находятся в одной сети и разделяют Virtual Router Identifier (VRID) и виртуальный IP-адрес;
- Владелец IP-адреса (IP Address Owner) — VRRP-маршрутизатор, который использует IP-адрес, назначенный виртуальному маршрутизатору, как реальный IP-адрес, присвоенный интерфейсу;
- VRRP-объявление (ADVERTISEMENT) — сообщения, которые отправляет Master-маршрутизатор;
- Виртуальный IP-адрес (Virtual IP address) — это IP-адрес, присвоенный интерфейсу одного из маршрутизаторов, которые составляют Virtual Router. Используется также название — основной IP-адрес (Primary IP Address). В VRRP-объявлениях в качестве адреса отправителя всегда используется виртуальный IP-адрес;
- Virtual Router Master или VRRP Master router — VRRP-маршрутизатор, который отвечает за отправку пакетов, отправленных на IP-адрес, который ассоциирован с виртуальным маршрутизатором, и за ответы на ARP-запросы, отправленные на этот адрес. Если владелец IP-адреса доступен, то он всегда становится Master;
- Virtual Router Backup или VRRP Backup router — это группа маршрутизаторов, которые находятся в режиме ожидания и готовы взять на себя роль VRRP Master router, как только текущий VRRP Master router станет недоступным;
- Виртуальный MAC-адрес (Virtual MAC) — 0000:5E00:01xx, где xx — номер группы VRRP.

53.2 Настройка кластера

Для настройки VRRP в режиме configure для всех участников кластера необходимо перейти в режим конфигурации VRRP:

```
Router(config)# service vrrp
```

В этом режиме можно активировать и деактивировать службу VRRP с помощью команд enable/disable, а также создавать и удалять участников (экземпляры) кластера и группы.

53.2.1 Создание кластера

Для функционирования кластера необходимо создать хотя бы одного участника кластера.

Для этого необходимо создать instance (экземпляр) кластера:

```
Router(service—vrrp)# instance outside
```

outside здесь любое удобное администратору имя. При этом осуществляется переход в режим настройки данного экземпляра участника кластера.

В этом режиме осуществляется связь виртуального IP-адреса с участником кластера и задаются другие параметры:

Команда	Назначение
description текст	Описание instance для администратора
id <идентификатор>	Задание номера группы для кластера
iface <интерфейс>	Привязка к интерфейсу
ip address <адрес/маска>	Задание виртуального IP-адреса
ip6 address <адрес/маска>	Задание виртуального IPv6-aflpeca
src-ip <адрес>	Задание IP-адреса, используемого в пакетах
src-ip6 <адрес>	Задание IPve-адреса, используемого в пакетах
adv-interval <секунды>	Интервал advertisement-оповещений
garp-delay <секунды>	Время в секундах для перехода в состояние Master
priority <приоритет>	Приоритет данного instance в кластере
state <master|backup>	Первоначальное состояние данного instance
password <пароль>	Защита сообщений паролем
vmac	Включить режим подмены виртуального MAC-адреса
preempt	Перехватывать роль у instance с низким приоритетом
preempt-delay <секунды>	Задержка для перехвата роли

Для всех описанных команд существуют аналоги с префиксом "no", которые используются для удаления соответствующей настройки. Например:

```
Router(service—vrrp—outside)# no vmac
```

Для удаления экземпляра следует пользоваться командой:

```
Router(service—vrrp)# no instance outside
```

При наличии хотя бы двух маршрутизаторов с созданными instance на них, принадлежащих одной группе и разделяющий одинаковый IP-адрес, кластер (после включения его командой enable) может выполнять роль виртуального маршрутизатора с заданным виртуальным IP-адресом. При наличии одного маршрутизатора с созданным instance, маршрутизатор также будет выполнять роль виртуального маршрутизатора, но без функций резервирования.

53.2.2 Создание групп синхронизации

По умолчанию, если участник кластера с ролью backup перестает получать сообщения от master, то он переходит в режим master и становится владельцем виртуального IP-адрес. Иногда бывает необходимым рассматривать группу виртуальных IP-адресов как одно целое. Это означает, что при сбое любого члена группы, должно происходить резервирование. Для этого существует понятие группы синхронизации участников кластера:

```
Router(service—vrrp)# group myrouter
```

При этом происходит переход в режим конфигурации группы, в котором можно добавлять и удалять instance с помощью команд member и no member, например:

```
Router(service—vrrp—myrouter)# member outside
Router(service—vrrp—myrouter)# no member outside
```

Группа рассматривается кластером как единое целое, и при сбое любого из instance группы, происходит смена роли (один из backup становится master).

Для удаления группы, воспользуйтесь командой:

```
Router(service—vrrp)# no group myrouter
```

53.2.3 Диагностика

Для просмотра сообщений от службы VRRP следует пользоваться командой show service vrrp log из режима enable:

```
Router# show service vrrp log
```

Кроме стандартных параметров для этой команды, существует параметр states, которые позволяет просмотреть только смену состояний (ролей).

Для просмотра текущей информации о состоянии кластера, следует пользоваться командой:

```
Router# show service vrrp state
```

Для получения более подробной и, наоборот, выборочной информации, можно воспользоваться командами: show service vrrp state all, show service vrrp state sgroups (информация по группам), show service vrrp state topology.

54. Отказоустойчивый кластер

Типичная схема использования отказоустойчивого кластера на основе маршрутизаторов Dionis-NX представлена на рисунке:

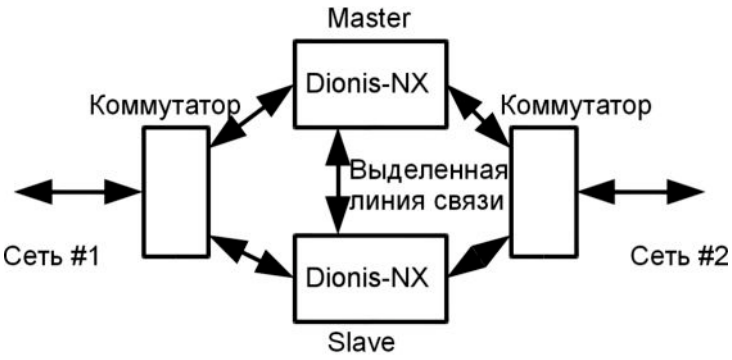

Рис. 54.1: Отказоустойчивый кластер

Предположим, маршрутизатор используется для связи сетей #1 и #2. При выходе из строя маршрутизатора связь между сетями будет утеряна. Чтобы этого не произошло, используется резервирование. Вместо одного маршрутизатора, устанавливается два одинаковых маршрутизатора. В каждый момент времени активен только один из них. Второй находится в резерве. Один из маршрутизаторов считается основным (master), второй резервным (slave). Когда работает основной маршрутизатор, резервный блокирует все свои интерфейсы, кроме одного служебного. Резервный маршрутизатор связан с основным специальной выделенной линией связи (dedicated link). Резервный маршрутизатор прослушивает выделенную линию связи и получает от основного маршрутизатора всю информацию, характеризующую состояние компонента TCP/IP, и специально сформированные «пакеты жизни» (heartbeat или advertizing message), которые служат признаком того, что основной маршрутизатор работоспособен. Если пакетов нет слишком долго, считается, что основной маршрутизатор вышел из строя. При этом резервный маршрутизатор временно становится основным (temp master), разблокирует свои интерфейсы и берет на себя все функции по обработке трафика. Если, после проведения ремонта или замены, основной маршрутизатор становится снова доступен и начинает генерировать пакеты жизни, резервный маршрутизатор возвращается в состояние ожидания.

Кроме обмена пакетами жизни, выделенная линия связи между основным и резервным маршрутизатором используется для синхронизации настроек маршрутизаторов и обмена информацией о текущих соединениях.

54.1 Требования к оборудованию

Для организации отказоустойчивого кластера на основе системы Dionis-NX рекомендуется использовать два одинаковых маршрутизатора, с одинаковым количеством интерфейсов и производительностью. Наличие выделенной линии связи основного маршрутизатора с резервным является обязательным условием. Также обязательным условием является установка на оба маршрутизатора одинаковой версии ОС Dionis-NX.

53.3 Подготовка к организации кластера

Перед объединением двух маршрутизаторов в кластер, на каждый из них необходимо установить одинаковую версию системы Dionis-NX и выполнить первичную настройку. Первичная настройка необходима для организации выделенной линии связи.

Перед настройкой параметров кластера, необходимо убедиться в том, что интерфейсы на каждом маршрутизаторе пронумерованы единообразно (п. 5.2. Настоятельно рекомендуется нумеровать все интерфейсы маршрутизаторов кластера единым образом во избежание путаницы, так как впоследствии настройки резервного маршрутизатора будут синхронизированы с настройками основного. Важно заметить, что нумерация интерфейсов входит в состав локальных настроек и не входит в состав конфигурации системы, и поэтому, не будет синхронизирована. На каждом маршрутизаторе нумерацию нужно провести отдельно.

После этого необходимо выбрать интерфейс для организации выделенной линии связи. На основном и резервном маршрутизаторе это должны быть одноименные интерфейсы.

Необходимо выбрать подсеть для организации выделенной линии связи. Подсеть служит для уникальной идентификации кластера. Для кластеров, находящихся в одной физической сети, выбранные подсети выделенной линии должны отличаться.

Для минимальной предварительной настройки основного маршрутизатора (master), необходимо выполнить следующие команды в режиме конфигурирования кластера:

```
Router(config)# interface ethernet 2
Router(config — if— ethernet2)# ip address 192.168.10.1/24
Router(config — if— ethernet2)# enable
Router(config)# cluster
Router(config—cluster)# aux interface ethernet 2
Router(config—cluster)# aux peer—addr 192.168.10.2/24
Router(config—cluster)# enable
```

В данном примере «ethernet 2» и «192.168.10.0/24» - выбранные для организации выделенной линии связи интерфейс и подсеть, соответственно. Локальный адрес для выделенного интерфейса задается в настройках этого интерфейса. В данном примере это "192.168.10.1/24". Выделенный интерфейс должен иметь строго один IP-адрес и быть активным (enable). В настройках кластера указывается выделенный интерфейс и IP-адрес резервного маршрутизатора (slave). Этот адрес должен быть установлен на выделенном интерфейсе резервного маршрутизатора. Команда "enable" запускает кластер в работу.

Для минимальной предварительной настройки резервного маршрутизатора, необходимо в режиме конфигурирования выполнить следующие команды :

```
Router(config)# interface ethernet 2
Router(config — if— ethernet2)# ip address 192.168.10.2/24
Router(config — if— ethernet2)# enable
Router(config)# cluster
Router(config—cluster)# slave
Router(config—cluster)# aux interface ethernet 2
Router(config—cluster)# aux peer—addr 192.168.10.1/24
Router(config—cluster)# enable
```

Команда «slave» в данном примере говорит о том, что данный маршрутизатор является резервным. По-умолчанию, маршрутизатор считается основным. Обратите внимание, что на выделенный интерфейс устанавливается IP-адрес, упомянутый на основном маршрутизаторе при настройке кластера как"аих peer-addr". При настройке кластера на резервном маршрутизаторе в поле"аих peer-addr" указывается IP-адрес выделенного интерфейса на основном маршрутизаторе.

Сформированную таким образом конфигурацию следует сохранить в startup-config. После этого маршрутизаторы можно объединять в кластер. Интерфейсы выделенной линии связи должны быть соединены друг с другом напрямую. Кластер практически готов к работе. Всю остальную настройку можно выполнить позже.

53.4 Настройки кластера

Все настройки кластера производятся в режиме конфигурирования кластера (config-cluster).

Чтобы указать маршрутизатору, является ли он основным или резервным, используется команда «slave». Для резервного маршрутизатора «slave» должен быть установлен, для основного - сброшен. Перевести резервный маршрутизатор в режим основного можно с помощью команды:

```
Router(config—cluster)# no slave
```

Настройка интерфейса и подсети для организации выделенной линии связи между маршрутизаторами внутри кластера описана в п. 54.2.

Для кластера может быть настроен интервал между посылками пакета жизни 54 основным маршрутизатором и тайм-аут, после которого резервный маршрутизатор должен считать, что основной маршрутизатор вышел из строя.

```
Router(config—cluster)# advert 1000
Router(config—cluster)# timeout 3000
```

Времена задаются в миллисекундах. В данном примере периодичность посылки пакетов жизни равна одной секунде, а тайм-аут, после истечения которого резервный маршрутизатор станет временно основным, равен трем секундам. В этом примере резервный маршрутизатор станет временно основным, если он не получил три подряд пакета жизни от основного маршрутизатора кластера. По умолчанию эти времена (advert и timeout) равны 500 мс и 1500 мс, соответственно.

При изменении состояния машин в кластере, т.е. когда резервный маршрутизатор становится активным вместо основного, либо основной возвращается в активное состояние, вытесняя резервный, машина, которая стала активной, посылает в сеть специальные пакеты, предназначенные для сетевых коммутаторов. Эти пакеты выходят через все интерфейсы системы. Дело в том, что сетевой коммутатор поддерживает внутреннюю таблицу соответствия своего физического порта и MAC-адресов, доступных через этот порт. Так как одноименные интерфейсы на двух машинах кластера имеют одинаковые MAC-адреса, но подключены к разным портам коммутатора, то при переходе одного из маршрутизаторов в активное состояние, коммутатору надо как можно быстрее дать знать, что порт, соответствующий MAC-адресу теперь другой. Иначе, пакеты, предназначенные для активного маршрутизатора будут попадать в порт, соединенный с пассивным маршрутизатором. Маршрутизатор, став активным, посылает в сеть служебные пакеты со своим MAC-адресом. Коммутатор, получив такой пакет, узнает, что данный MAC-адрес доступен теперь через другой порт и меняет записи в своих внутренних таблицах. Траффик начинает уходить к

активному маршрутизатору. Так как переход из активного состояния в пассивного и обратно не происходит мгновенно, может понадобиться послать несколько служебных пакетов. Для управления генерацией служебных пакетов существуют настройки кластера.

```
Router(config—cluster)# mac—advert 100
Router(config—cluster)# mac—advert retry 20
```

Директива "mac-advert" задает период выдачи служебных пакетов в сеть (в миллисекундах). Рекомендуется задавать небольшой период. В этом случае восстановления нормальной работы кластера после перехода управления от одной машины к другой произойдет быстрее.

Директива "mac-advert retry" определяет сколько служебных пакетов будет сгенерировано. В данном примере будет выдано 20 пакетов через каждый активный сетевой интерфейс маршрутизатора. Существует значение "indefinite", при котором пакеты будут генерироваться постоянно.

При загрузке маршрутизатора есть определенный промежуток времени, когда активные сетевые интерфейсы согласуют параметры соединения с удаленной стороной (negotiation). Это процесс асинхронный и может занимать достаточно длительное время. Кроме этого и сама активация интерфейсов в системе - процесс асинхронный. Предположим в кластере резервный маршрутизатор является активным и в это время загружается основной маршрутизатор и перехватывает управление. Если ко времени перехвата управления интерфейсы не готовы к полноценной работе, то проходящий через маршрутизатор траффик будет потерян. Чтобы этого не происходило существует настройка кластера "delay". Она определяет сколько времени после загрузки основной маршрутизатор не будет перехватывать управление, т.е. время на приведение интерфейсов в рабочее состояние.

```
Router(config—cluster)# delay 20
```

В примере задержка устанавливается в 20 секунд. Конкретное время необходимой задержки сильно зависит от конфигурации маршрутизатора, количества интерфейсов. Определяется в каждом конкретном случае экспериментально.

Основной и резервный маршрутизатор обмениваются информацией об активных соединениях (conntrack), чтобы резервный маршрутизатор (а в случае перезагрузки основного, то и основной) «подхватил» уже установленные соединения автоматически. Если в каких то случаях пересылка информации о соединениях в реальном времени является нежелательной, вы можете отключить ее командой:

```
Router(config—cluster)# no conntrack service
```

Команда conntrack service снова включит передачу информацию о соединениях.

Для синхронизации часов реального времени между основным и резервным маршрутизаторами, предусмотрен специальный механизм. Основной маршрутизатор может посылать резервному через выделенную линию связи текущее время. Период синхронизации можно задать.

```
Router(config—cluster)# time—sync 10
```

В примере период синхронизации времени установлен в 10 минут.

Чтобы запустить кластер, необходимо выполнить следующую команду:

```
Router(config—cluster)# enable
```

Чтобы остановить кластер, необходимо выполнить следующую команду:

```
Router(config—cluster)# disable
```

После запуска кластера администратор может менять его настройки, но эти изменения не будут сразу применены. Чтобы применить введенные настройки, необходимо остановить кластер и вновь его запустить. Если администратор изменил настройки при работающем кластере, он будет предупрежден об этом с помощью значка »~» в приглашении командной строки в режиме конфигурирования кластера:

```
Router(config—cluster)^#
```

Знак »~» означает рассинхронизацию между настройками работающего кластера и текущими настройками в running-config.

54.4 Получение информации о кластере

Получить информацию о текущем состоянии кластера можно с помощью следующей команды, выполненной из привилегированного режима:

```
Router# show cluster

Mode                : master
State               : active
Interface           : ethernet2
Master IP-address : 192.168.30.49/30
Slave IP-address : 192.168.30.50/30
Advert period       :  500 msec
Advert timeout : 1500 msec
MAC—advert period : 100 msec
MAC—advert retry : 20
Starting delay      :  3 sec
Time sync           :  1 min
ConnTrack           :  on
```

Свойство "Mode" показывает текущий режим маршрутизатора:

- master - основной маршрутизатор;
- slave - резервный маршрутизатор;
- slave (temp master) - резервный маршрутизатор, получивший управление из-за неработоспособности основного.

Свойство «State» состояние маршрутизатора:

active - в активном состоянии, т.е. выполняет свои функции согласно своему текущему режиму;
pause - функционирование временно приостановлено командой "cluster pause".

Свойства «Interface», «Master IP-address», "Slave IP-address" отображают выделенный интерфейс и его сетевые адреса на основном и резервном маршрутизаторах соответственно.

Свойства «Advert period» и «Advert timeout» отображают период посылки пакета жизни основным маршрутизатором и тайм-аут, по истечении которого основной маршрутизатор считается неработоспособным.

Свойства "MAC-advert period", "MAC-advert retry" отображают настройки отправки специальных пакетов для коммутаторов. Соответственно период отправки и количество пакетов.

Свойство "Starting delay" отображает задержку перехвата управления основным маршрутизатором при загрузке.

Свойство «ConnTrack» отображает включен ли режим обмена информацией об установленных соединениях.

54.5 Пауза в работе кластера

Иногда требуется принудительно перевести один из маршрутизаторов кластера в неактивное состояние. Для этого используется команда enable-режима "cluster pause".

При выполнении команды "cluster pause" на основном маршрутизаторе, он переходит в пассивный режим. Не посылает пакетов жизни, блокирует работу всех сетевых интерфейсов, кроме выделенной линии связи с резервным маршрутизатором.

При выполнении команды "cluster pause" на резервном маршрутизаторе, он также переходит в пассивный режим, т.е. не отслеживает тайм-аут при приеме пакетов жизни от основного маршрутизатора и, соответственно, не может перехватить управление.

Чтобы восстановить нормальную работу кластера, используется команда enable-режима "cluster resume".

54.6 Синхронизация настроек между маршрутизаторами

Так как маршрутизаторы в кластере взаимозаменяемы, то их системные настройки должны быть идентичны. Для этого предусмотрена возможность синхронизации настроек. После настройки системы на основном маршрутизаторе, конфигурация может быть полностью перенесена на резервный маршрутизатор.

Для удобства последующей синхронизации необходимо однократно выполнить ряд действий на основном маршрутизаторе (предполагается, что кластер запущен) в привилегированном режиме:

```
Router# cluster key generate
Router# cluster key export
```

Эти команды предназначены для шифрования соединения между основным и резервным маршрутизаторами. Первая команда генерирует ключ для шифрования, а вторая - передает этот ключ на резервный маршрутизатор. При выполнении второй команды будет запрошен пароль администратора на резервном маршрутизаторе. После выполнения этих действий последующее

взаимодействие между основным и резервным маршрутизаторами не потребует ручного ввода пароля администратора.

Для синхронизации системной конфигурации необходимо выполнить следующие действия из привилегированного режима:

```
Router# cluster sync
```

При этом конфигурация (с добавлением служебной информации) основного маршрутизатора будет скопирована на резервный маршрутизатор. Так как заранее неизвестно, какие настройки были изменены, для применения новых настроек требуется перезагрузка резервного маршрутизатора. Поэтому, если настройка на основном маршрутизаторе завершена, то можно использовать команду:

```
Router# cluster sync reboot
```

При этом, после синхронизации настроек, резервный маршрутизатор будет автоматически перезагружен, чтобы применилась новая конфигурация системы.

Для того, чтобы администратор с основного маршрутизатора мог зайти на резервный маршрутизатор или с резервного на основной, предусмотрена команда:

```
Router# cluster connect
```

К имени хоста в приглашении командной строки на резервном маршрутизаторе будет добавлена строка "[slave]", на основном "[master]" - для того, чтобы администратор легко мог отличить основной маршрутизатор от резервного.

```
Router[slave]#
```

54.7 Дополнительные команды

В разделе «Синхронизация настроек между маршрутизаторами» были указаны команды для исключения ручного ввода команд при взаимодействии маршрутизаторов внутри кластера. Однако при замене одного из маршрутизаторов в кластере или в случае, если маршрутизаторы исключаются из кластера и становятся самостоятельными, может потребоваться очистка некоторых настроек.

Так на основном маршрутизаторе сохраняются параметры для взаимодействия с резервным. Поэтому при замене резервного маршрутизатора необходимо очистить параметры, которые соответствовали старому маршрутизатору, иначе взаимодействие с новым резервным маршрутизатором будет заблокировано. Для этого используется команда привилегированного режима:

```
Router# clear cluster known —hosts all
```

На резервном маршрутизаторе также хранятся параметры основного маршрутизатора. Поэтому, при замене основного маршрутизатора, на резервном следует выполнить команду:

```
Router—slave# clear cluster authorized—keys all
```

388
RU.HKBT.70007-01 90
RU.HKBT.70007-01 90

55. Балансировка прерываний

При высокой сетевой нагрузке на маршрутизатор становится актуальной проблема балансировки прерываний.

При приеме и отправке траффика сетевые интерфейсы вырабатывают аппаратные прерывания. Каждое аппаратное прерывание привязано к определенному процессорному ядру. Соответствующий обработчик прерывания выполняется на том ядре, к которому привязано аппаратное прерывание. При высокой сетевой нагрузке, время обработки принимаемых и отправляемых пакетов становится существенным. Процессорное ядро может быть полностью, на 100% загружено обработкой прерываний, т.е. перегружено. Когда процессорное ядро перегружено, траффик, который уже невозможно обработать, уничтожается (drop).

Строго говоря современные сетевые карты вырабатывают довольно мало прерываний. Может вырабатываться одно прерывание на несколько пакетов. Однако вместо слишком частых прерываний, когда на каждый принятый пакет вырабатывалось свое прерывание, используется механизм NAPI. Этот механизм временно отключает аппаратные прерывания для конкретной сетевой карты, но при этом после обработки пакета, обработчик самостоятельно проверяет не приняты ли сетевой картой еще новые пакеты (polling). Если пакеты приняты, то новая порция пакетов обрабатывается сразу же. Прерывания вновь будут включены только при отсутствии вновь принятых пакетов.

Одна сетевая карта может иметь несколько аппаратных очередей. Каждая очередь использует свое прерывание. Сетевые пакеты попадают в одну из аппаратных очередей в зависимости от хеш-функции, расчитанной по некоторым полям сетевого пакета. Обычно это адреса и порты отправителя и адресата. Т.е. один сетевой поток будет попадать в одну аппаратную очередь сетевой карты. Аппаратные очереди созданы для того, чтобы весь траффик от сетевой карты обрабатывался не на одном процессорном ядре, а распределялся на разные ядра.

Однако когда сетевых карт много, очередей много и высокая сетевая загрузка, распределение потоков по очередям не решает проблемы. Отдельные процессорные ядра становятся перегружены, а другие ничего не делают. Решить проблему можно динамической балансировкой прерываний (балансировкой обработки прерываний). Балансировщик должен снимать прерывания с перегруженных процессорных ядер и переносить на менее загруженные. В ОС Dionis-NX есть такой балансировщик. Его параметры настраиваются в секции конфигурирования "service balance-irq".

Балансировщик через определенные отрезки времени анализирует загрузку процессорных ядер. Если в системе есть перегруженные прерываниями ядра - состояние считается ненормальным, если нет - нормальным. В ненормальном состоянии балансировщик ищет наиболее загруженное ядро, выбирает одно из прерываний, привязанных к этому ядру и переносит его на другое, наименее загруженное ядро.

Промежуток времени (период) между анализом загруженности ядер зависит от того, есть ли в системе перегруженные процессоры или нет. Если перегруженные процессоры есть, то балансировщик считает, что период нужно сократить, чтобы быстрее выйти из ненормального состояния. Если перегруженных процессоров нет, то период можно увеличить. Короткий период можно задать опцией "interval active" (по-умолчанию 2 сек), длинный - опцией "interval passive" (по-умолчанию 5 сек).

```
DionisNX(config—service—balance—irq)# interval active 2
DionisNX(config—service—balance—irq)# interval passive 5
```

Уровень загруженности ядра прерываниями (irq, softirq), при котором ядро считается перегруженным, можно задать опцией "threshold" (по-умолчанию 99%).

```
DionisNX(config—service—balance—irq)# threshold 99
```

Выбрав самое перегруженное ядро, балансировщик выбирает одно из прерываний, чтобы перенести его на другое ядро. Есть несколько стратегий выбора прерывания для переноса:

- прерывание, обработчик которого был вызван наибольшее количество раз за последний период (max);
- прерывание, обработчик которого был вызван наименьшее количество раз за последний период (min);
- случайный выбор прерывания (rnd).

С началом использования механизма NAPI, прямая зависимость количества вызовов обработчика и общим вкладом конкретного прерывания в общую загруженность ядра - исчезла. Поэтому рекомендуемая стратегия - это случайный выбор прерывания (rnd). Эта стратегия используется по-умолчанию. Надо заметить, что прерывания, обработчик которых не был вызван ни разу за последний период - игнорируются и не могут быть выбраны для переноса на другое ядро. Стратегию выбора прерывания для переноса можно выбрать опцией "strategy".

```
DionisNX(config—service—balance—irq)# strategy rnd
```

При выборе балансировщиком ядра, на которое следует перенести прерывание, выбирается наименее загруженное ядро в системе. Однако если в системе не осталось свободных ядер и все ядра довольно сильно загружены, то перенос может не иметь смысла. Опция "load-limit" задает уровень загруженности ядра при котором перенос на него прерываний считается бессмысленным. По умолчанию этот уровень загруженности - 95%.

```
DionisNX(config—service—balance—irq)# load —limit 95
```

При необходимости некоторые процессорные ядра можно принудительно исключить из списка кандидатов при выборе целевого ядра для переноса на него прерывания.

```
DionisNX(config—service—balance—irq)# exclude—cpus 0,1,3—7
```

В данном примере из списка исключаются ядра с номерам 0, 1 и весь диапазон от 3-го ядра до 7-го.

На некоторых мощных маршрутизаторах может использоваться технология NUMA (Non Uniform Memory Access). Эта технология применяется на многопроцессорных (не путать с многоядерными) системах. Каждый процессор имеет "собственную" оперативную память, доступ к которой осуществляется быстро. Доступ к оперативной памяти других процессоров тоже возможен, но он производится с помощью обращения одного процессора к другому по специальным линиям связи. Процессор - "владелец" памяти сам получает значения из своей памяти и пересылает запросившему процессору. Это тяжелая и длительная операция. Кроме того, в многопроцессорных системах контроллеры шин, таких как PC1e, часто привязаны к одному из процессоров. Соответственно память именно этого процессора будет использована DMA-устройствами на этой шине. Все ядра такого процессора будут считаться локальными для устройств, подключенных по шине. Прерывания от устройств могут обрабатывать как локальные, так и не локальные ядра, однако запрос на прерывания так-же придется передавать по специальным линиям связи от локального

процессора к удаленному. Если прерывание от устройства обрабатывает нелокальное ядро, то ему придется обращаться к памяти другого процессора, что очень долго.

Таким образом, обработка прерываний на нелокальных ядрах очень неэффективна. По-умолчанию балансировщик не использует нелокальные ядра для перемещения на них прерываний. Если администратор все-таки хочет использовать нелокальные ядра, то можно использовать опцию "non-local-cpus".

DionisNX(config-service-balance-irq)# non-local-cpus

Но даже в этом случае, балансировщик отдает предпочтение локальным ядрам и использует нелокальные только в том случае, когда загрузка всех локальных ядер выше, чем "load-limit".

392

RU.HKBT-70057-01 90

56. Обновление системы

Администратор имеет возможность производить обновление системы Dionis-NX. Обновление может быть локальным, если администратор имеет физический доступ к оборудованию, или удалённым, если оборудование физически недоступно. Работы по установке и управлению обновлениями администратор должен производить из командной строки в привилегированном режиме.

56.1 DIP-пакеты

Обновления системы Dionis-NX предоставляются в виде DIP-пакетов. Аббревиатура DIP расшифровывается как «Dionis Package». DIP-пакет - это файл с именем вида «dionisnx-1.0-0.x86_64.dip», где «1.0» - версия системы, «0» - номер редакции (релиз), «x86_64» - архитектура целевой платформы. Пакет содержит информацию о предоставляемой системе - версия, дата создания, характеристики и т.д., ядро системы ОС Dionis-NX, образ корневой файловой системы Dionis-NX.

DIP-пакет привязан к конкретному экземпляру оборудования, для которого он был создан. Он не может быть установлен на другой маршрутизатор. Для маршрутизатора вычисляется идентификатор оборудования (Platform ID). Для каждого экземпляра маршрутизатора этот идентификатор имеет уникальное значение. Администратор может узнать идентификатор текущей платформы с помощью команды привилегированного режима:

```
Router# show version

Platform ID: 4F7F-7879-F676-E85B-5369
```

56.2 Инфраструктура DIP

На маршрутизаторе может быть одновременно установлено несколько экземпляров операционной системы Dionis-NX. Это могут быть и разные версии ОС, и несколько экземпляров системы одной версии. Возможность использования нескольких версий ОС нужна для безопасного обновления системы, а также для организации отката (fallback) к работоспособному экземпляру системы при возникновении сбоев в работе текущей системы.

Все установленные экземпляры операционной системы (будем называть их пакетами ОС или OS package) доступны только для чтения. Дополнительные данные для операционных систем (конфигурация, настройки и т.д.) хранятся отдельно и доступны для чтения и записи. Эти данные хранятся в области внутреннего диска маршрутизатора, называемом «слот данных» (data slot). Одновременно на диске может существовать несколько слотов данных.

Обычно каждый пакет ОС связан (bind) со своим слотом данных, где он и хранит данные. Однако пакеты и слоты данных не связаны друг с другом жестко. Могут существовать пакеты ОС, не имеющие своего слота данных, а также слоты данных, не привязанные ни к одному пакету.

К примеру, вновь установленное обновление ОС не имеет своего слота данных. Пакет получит слот данных либо автоматически при загрузке, либо администратор вручную свяжет этот пакет с уже существующим слотом данных. В случае загрузки операционной системы, не имеющей на текущий момент своего слота данных, новый слот данных будет создан автоматически и привязан к загружаемой системе. Таким образом, пакет может существовать без слота данных в пассивном режиме, но не может без него работать.

Каждый установленный пакет ОС идентифицируется уникальным именем. Каждый слот данных также идентифицируется уникальным именем. Используя эти уникальные имена, администратор системы может производить различные действия над пакетами ОС и слотами данных. Операции над текущим (активным) пакетом ОС и активным слотом данных ограничены, так как невозможно, к примеру, удалить текущий слот данных, не нарушив работу маршрутизатора.

Команды для пакетов ОС:

Команда	Назначение
os install <dip-pkg>	Установка нового пакета ОС. Источником является DIP-пакет
os remove <os>	Удаление существующего пакета ОС. Невозможно для активного пакета
os rename <old> <new>	Переименование существующего пакета ОС. Меняется уникальное имя пакета в системе
os export <os-name>	Экспорт существующего пакета ОС. Будет создан DIP-пакет
os bind <os> <dataslot>	Привязка пакета ОС к существующему слоту данных. Невозможно для активного пакета ОС. Невозможно для уже привязанного слота данных
os bind <os>	Отвязывание пакета ОС от слота данных. Невозможно для активного пакета ОС
show os	Получение списка установленных пакетов ОС
show os info	Получение подробной информации об установленных пакетах ОС

Команды для слотов данных:

Команда	Назначение
os data create <name>	Создание нового пустого слота данных
os data clone <old> <new>	Клонирование слота данных. Создается новый слот, дублирующий содержимое исходного слота
os data remove <name>	Удаление существующего слота данных. Невозможно для слотов, привязанных к какому-либо пакету ОС
os data rename <old> <new>	Переименование существующего слота данных
os data backup <name> <path>	Создание резервной копии данных на основании существующего слота данных. Создается файл - резервная копия

Команда	Назначение
os data restore <name> <path>	Восстановление слота данных на основании резервной копии данных. Невозможно для активного слота данных
schedule backup <path>	Безопасное создание резервной копии активного слота с перезагрузкой
schedule restore <path>	Восстановление текущего слота данных на основании резервной копии
schedule migrate <os>	Миграция на другой пакет ОС с сохранением текущего слота данных. Требуется перезагрузка
schedule rebind <dataslot>	Миграция на другой слот данных с сохранением текущей ОС. Требуется перезагрузка
show os data	Получить список существующих слотов данных

Команды загрузки системы:

Команда	Назначение
boot default <os>	Задать пакет ОС, который будет загружаться по умолчанию
boot fallback <os>	Задать пакет ОС, который будет загружен в случае необходимости отката к предыдущей версии (fallback).
boot experimental <os>	Установить для пакета ОС признак того, что эта ОС является экспериментальной
show boot	Получить текущую конфигурацию загрузчика

Общие операции:

Команда	Назначение
show os summary	Получить сводку состояния DIP-инфраструктуры

56.3 Установка обновления

Для начала установки DIP-пакет обновления должен быть скопирован в локальную файловую систему маршрутизатора. Это может сделать администратор с помощью команд привилегированного режима «copy» или «ssh get» (п. 34.4). В случае локального обновления источником пакета обновления будет служить флеш-диск. В случае удаленного копирования (с помощью команды «ssh get»), между рабочим местом администратора и маршрутизатором должен быть установлен доверенный канал передачи информации. Копирование обновлений без установления доверенного канала передачи информации не допускается.

Локальными хранилищами файлов на диске маршрутизатора являются пространства имен «file:» и «share:». Хранилище «file:» доступно только из текущей загруженной версии системы Dionis-NX. Каждая установленная версия системы Dionis-NX имеет собственное хранилище «file:».

недоступное для других версий. Хранилище «share:» доступно для всех установленных систем. Это хранилище может быть использовано для передачи данных между разными версиями установленных ОС.

Копирование может быть выполнено при помощи команды:

```
Router# copy flash0.1:/dionisnx—1.0—0.x86_64.dip file:
```

После этого можно начать установку обновления:

```
Router# os install file:/dionisnx—1.0.1.x86_64.dip
```

Если операция прошла успешно, на машине будет установлено две системы Dionis-NX. Список установленных систем можно получить с помощью команды:

```
Router# show os
```

Можно получить подробную информацию о конкретной установленной системе при помощи команды:

```
Router# show os info dionisnx—1.0—0
```

Любой установленный пакет ОС может быть переименован. Новое название должно быть уникально:

```
Router# os rename dionisnx—1.0—0 mysystem
```

После этого пакет ОС в системе идентифицируется новым именем «mysystem».

Если какой-либо пакет ОС устарел и не используется, его можно удалить:

```
Router# os remove dionisnx—0.9—0
```

56.4 Параметры загрузки

После установки обновления нужно указать первичному загрузчику, какую из установленных систем следует загружать по умолчанию. Следующая команда покажет текущие установки первичного загрузчика:

```
Router# show boot
0   dionisnx—1.0—0 (D) (F) (C)
1   dionisnx—1.0—1
```

Первое поле - порядковый номер установленной системы (начиная с 0). Второе - идентификатор установленной системы. Отображаемые в строке признаки имеют следующее значение:

- (D) - (default). После перезагрузки данная система будет загружена по-умолчанию;
- (F) - (fallback). При возникновении проблем с загрузкой системы по-умолчанию (помеченной флагом »(D)»), произойдет откат к системе, помеченной флагом »(F)» (резервная система);
- (C) - (current). Текущая система, т.е. система, загруженная сейчас;

- (E15) - (experimental). Система загружена в «экспериментальном» режиме. Описание экспериментального режима работы ОС приведено в данном разделе ниже. Число после символа «E» означает количество минут до перезагрузки;
- (d) - (user default). Система была помечена администратором, как система по умолчанию, но по какой-либо причине произошел откат к резервной системе.

Указать загрузчику, какая система является загружаемой по умолчанию, а какая является резервной, можно следующими командами:

```
Router# boot default dionisnx—1.0—1
Router# boot fallback dionisnx—1.0—0
```

Типичные параметры первичного загрузчика при локальном обновлении системы:

```
Router# show boot
0  dionisnx—1.0—0 (F) (C)
1  dionisnx—1.0—1 (D)
```

Старая система становится резервной. По умолчанию загружается новая система.

Для удаленного обновления предусмотрен дополнительный механизм, обеспечивающий доступ администратора к системе при возникновении проблем со вновь установленной системой - работа в экспериментальном режиме. Администратор может пометить систему, как «экспериментальную». При загрузке экспериментальной системы будет взведен специальный таймер и, по истечении указанного тайм-аута, маршрутизатор будет автоматически перезагружен. После перезагрузки экспериментальной системы произойдет автоматический откат к резервной системе. Механизм работы в экспериментальном режиме позволяет защититься от неверных сетевых настроек в новой системе, при которых удаленный администратор потеряет возможность входа в систему. Если новая система загрузилась успешно и доступна, администратор может дать команду для снятия экспериментального режима. После этого таймер будет остановлен и автоматической перезагрузки не произойдет.

Установка экспериментального режима:

```
Router# boot experimental dionisnx—1.0—1 15
```

Последним параметром задается время в минутах до автоматической перезагрузки. После этой команды, параметры первичного загрузчика будут выглядеть так:

```
Router# show boot
0  dionisnx—1.0—0 (F) (C)
1  dionisnx—1.0—1 (D) (E15)
```

Предположим, что экспериментальная система загрузилась успешно и доступна. Администратор может войти в систему и узнать текущий статус системы и время, оставшееся до автоматической перезагрузки:

```
Router# show boot experimental
```

Далее администратор может снять экспериментальный режим и сделать новую систему "системой по умолчанию":

```
Router# no boot experimental dionisnx—1.0—1
Router# boot default dionisnx—1.0—1
```

Кроме экспериментального режима, который касается всей ОС, существует экспериментальный режим только для файлов конфигурации. Экспериментальные файлы конфигурации описаны в отдельном разделе.

56.3 Конфигурация системы и данные

Каждая установленная система имеет (или получит при первой загрузке) выделенную область для хранения своей конфигурации и данных - слот данных. Так как для каждой системы конфигурация хранится отдельно, то откат на резервную систему восстановит также и резервную конфигурацию.

Типичная задача при установке обновления - миграция существующей конфигурации и данных в новую систему. Чтобы узнать для каких установленных систем уже существует область хранения данных, администратор может выполнить команду:

```
Router# show os data
```

Для получения более подробной информации используется следующая команда:

```
Router# show os summary
```

Для копирования данных из области хранения старой системы в область хранения новой системы, необходимо выполнить команду:

```
Router# os data clone dionisnx—1.0—0 dionisnx—1.0—1
```

Предполагается что dionisnx-1.0-0 - это существующая и настроенная система, а dionisnx-1.0-1 - вновь установленная система. При таком копировании следует иметь в виде что, в принципе, возможна ситуация, когда формат команд новой и старой версий ОС отличается. В этом случае необходимо внести соответствующие изменения в скопированные данные.

Если какой-то слот данных больше не нужен (например, соответствующая система устарела и удалена), данные и конфигурация могут быть стерты с диска:

```
Router# os data remove dionisnx—1.0—0
```

Может быть создан новый пустой слот данных. Это может понадобиться, если администратор желает восстановить данные, используя ранее созданную резервную копию (backup):

```
Router# os data create my_new_slot
```

Для более удобной идентификации слот данных может быть переименован. Новое имя должно быть уникально в системе (относительно других слотов данных).

```
Router# os data rename my_new_slot new_name
```

Где «my_new_slot» - существующий слот данных, а «new_name» - его новое имя.

56.4 Привязка данных

Слот данных может быть привязан к установленному пакету ОС. Это означает, что при загрузке этой ОС для хранения конфигурации и данных будет использован именно привязанный слот данных.

Привязка слота данных к пакету ОС производится следующей командой:

```
Router# os bind dionisnx—1.0—0 data~l
```

Где «dionisnx-1.0-0» - установленный пакет ОС, а «data~l» - имя существующего слота данных.

Если загружается система, не имеющая привязанного слота данных, то новый слот будет создан автоматически и автоматически же привязан к текущему пакету ОС.

Операция по привязке слота данных «os bind» не может быть выполнена для слота данных, который уже привязан к какому либо пакету ОС. Такой слот необходимо сначала отвязать и только потом использовать:

```
Router# os bind dionisnx—1.0—0
```

Данная команда (без указания слота данных) отвяжет пакет ОС от слота данных.

Текущая активная система не может быть привязана или отвязана от слота данных "на лету". Также недопустимы операции над текущим слотом данных. Для операций над текущим пакетом ОС и текущим слотом данных смотрите раздел «Миграция ОС».

Привязки слотов данных можно узнать с помощью команды «show os summary»:

```
Router# show os summary
Installed OSes:
mysystem {dionisnx—1.0—0} [data~l] (D) (C)
anothersys {dionisnx—0.9—0} [anotherdata] (F)
newsys {dionisnx—1.0—1}
Data slots:
   data~l [mysystem] (C)
   anotherdata [anothersys]
   not—binded—data
```

В выводе этой команды сначала перечислены установленные пакеты ОС. Первое поле - идентификатор (имя) системы. Поле в фигурных скобках показывает версию системы. Поле в квадратных скобках указывает на привязанный слот данных. После установленных пакетов ОС перечислены существующие слоты данных. Первое поле - имя слота данных. Поле в квадратных скобках - пакет ОС, к которому привязан слот.

56.5 Миграция ОС

Часто возникает ситуация, когда администратор хочет установить новую версию ОС Dionis-NX, но использовать текущую конфигурацию. Проблема в том, что текущая конфигурация и данные

хранятся в активном слоте данных. Используемый в данный момент слот данных нельзя привязать к вновь установленной системе с помощью команды «os bind», так как слот данных уже имеет привязку, а отвязать слот от работающей системы невозможно. Таким образом, данная задача решается только с использованием перезагрузки работающей системы.

```
Router# schedule migrate dionisnx—1.0—1
```

Команда планирует миграцию на указанный пакет ОС в ходе следующей перезагрузки. Команду «reboot» для начала перезагрузки администратор должен ввести вручную. На этапе ранней загрузки текущий слот данных будет отвязан от текущей системы и привязан к новой. После этого будет загружена новая система со старым слотом данных.

Существует и обратная задача. Когда требуется мигрировать на другой слот данных, но используя текущий пакет ОС. Эта задача также решается через перезагрузку.

```
Router# schedule rebind data_new
```

Команда планирует привязку слота данных «data_new» к текущему пакету ОС. После перезагрузки будет загружена старая система с новым слотом данных.

56.6 Экспериментальный файл конфигурации

Операция копирования вновь созданного файла конфигурации в startup-config иногда может быть опасной. Так, при удаленном управлении маршрутизатором возможна ситуация, когда при перезагрузке системы с новыми непродуманными настройками, будет утеряна связь. В итоге администратор потеряет возможность удаленного управления. Чтобы этого не допустить существует специальный режим экспериментального конфигурационного файла.

В режиме экспериментального конфигурационного файла администратор указывает отдельно сохраненный файл конфигурации с новыми настройками и планирует его использование при следующей перезагрузке системы.

```
adm@DionisNX# schedule experimental—config file://new_config 15
```

После перезагрузки будет использован указанный файл конфигурации. Однако, через определенный тайм-аут (в данном примере указан тайм-аут в 15 минут), система будет перезагружена с использованием старого корректного файла конфигурации. Таким образом при неправильных новых настройках, система восстановит прежнее состояние и доступность по сети.

В случае, если новый файл конфигурации корректен, то администратор может удаленно войти в систему и выключить режим экспериментального файла конфигурации.

```
adm@DionisNX# clear experimental—config
```

При выключении экспериментального режима отключается таймер, приводящий к перезагрузке.

Также администратор может посмотреть текущее состояние экспериментального режима.

```
adm@DionisNX# show experimental—config
```

Команда покажет включен ли экспериментальный режим и, если включен, время оставшиеся до перезагрузки системы.

Надо помнить, что отключение экспериментального режима не заменяет startup-config новым файлом конфигурации автоматически. Поэтому для завершения операции по переходу на новый файл конфигурации, администратор должен выполнить следующую команду:

```
adm@DionisNX# copy running—config startup-config
```

или

```
adm@DionisNX# copy file://new_config startup-config
```

56.7 Резервная копия пакета ОС

Администратор может создать DIP-пакет из уже установленного пакета ОС. Полученный пакет может использоваться в целях резервного копирования.

```
Router# os export dionisnx—1.0—0 file:
```

Корневая файловая система, ядро и дополнительная информация будут завёрнуты в DIP-пакет, который, в свою очередь, будет помещен в локальное хранилище файлов «file:» с именем dionisnx-l.0-0.x86_64.dip. Созданный DIP-пакет будет привязан к данному экземпляру оборудования и не может быть установлен на другую машину.

401 402

RU.HKBT.70007-01 90

RU.HKBT.70007-01 90

57. Обслуживание

57.1 Резервное копирование

В системе Dionis-NX предусмотрено резервное копирование данных. Слот данных содержит текущие настройки системы, данные и файлы системы протоколирования.

Из-за непредсказуемости изменений и состояния файлов, создание полной резервной копии во время штатной работы системы не гарантирует целостность данных. По тем же причинам восстановление системы из полной резервной копии "на лету" невозможно. Для восстановления системы требуется перезагрузка.

57.1.1 Создание резервной копии

Для удобства администрирования все же предусмотрено создание резервной копии работающей системы "на лету". Однако надо помнить, что создание резервной копии в этом случае не гарантирует целостность данных, так как сохранение копий файлов процесс не мгновенный и в момент сохранения одного файла, другие (еще не сохраненные) могут изменяться. Чаще всего это не опасно, так как основные параметры конфигурации системы представлены всего несколькими небольшими файлами, хотя для больших файлов журналирования проблема актуальна.

Создание резервной копии слота данных производится командой:

```
Router# os data backup data~l share:
```

Где «data~l» - идентификатор слота данных (это может быть как текущий слот данных, так и любой другой), «share:» - место, куда сохранить резервную копию. Резервная копия может быть сохранена в пространство «share:», "file:", а также на флеш-носитель (например flash0.1:). По умолчанию в резервную копию попадут только основные настройки системы. При создании резервной копии предусмотрены следующие опции, меняющие такое поведение:

- files - Сохранять файлы из пространства file:;
- log - Сохранять файлы журналирования;
- old-backups - В пространстве file: могут уже находиться резервные копии с расширением "dbu". Опция позволяет включить их в новую резервную копию. По умолчанию такие файлы будут проигнорированы для экономии дискового пространства;
- name <имя> - Даёт возможность задать имя файла - резервной копии;
- desc <текст> - Даёт возможность задать описание резервной копии.

Для надежного создания резервной копии с гарантируемой целостностью, предусмотрена команда «schedule backup». Чтобы запланировать создание резервной копии раздела данных во время следующей перезагрузки, необходимо в привилегированном режиме (enable - режим) ввести следующую команду:

```
# schedule backup flash0.1
```

Третий аргумент «flash0.1» определяет носитель, на котором будет сохранена резервная копия. Данная запись означает, что резервная копия будет сохранена на первом найденном флеш-диске (отсчет ведется от нуля) и на первом разделе этого диска. При вводе третьего аргумента можно нажать кнопку «TaЬ», чтобы увидеть доступные в данный момент носители. Для этой команды предусмотрены те же опции что и для команды создания резервной копии "на лету".

Резервная копия представляется в виде файла backup-dionisnx-<версия>-<дата>-<время>.dbu. Файл содержит сжатый образ раздела данных и текстовый файл описания резервной копии. В общем случае резервная копия может представляться в виде нескольких файлов. Это произойдет в случае превышения файлом размера в 2Гб. Разбиение по 2Гб позволяет хранить большие резервные копии на носителях с файловой системой FAT32.

57.1.2 Просмотр доступных резервных копий

Для просмотра уже существующих на носителе резервных копий используется команда привилегированного режима:

```
Router# show backup share:
```

В данном случае на экран будет выведен список резервных копий, содержащихся в пространстве «share:». Пример вывода:

```
Profile : share:/backup—dionisnx—1.0—0—121105—104228.dbu
System ID : dionisnx—1.0—0
Description : Testing
Date/Time : 2012.11.05 10:42:28
```

57.1.3 Восстановление из резервной копии

Для восстановления раздела данных из резервной копии используется команда привилегированного режима:

```
Router# schedule restore flash0.1:/backup—dionisnx—1.0—0—121105—104228.dbu
```

Третий аргумент указывает файл резервной копии. После этой команды необходимо произвести перезагрузку системы. В процессе перезагрузки старый слот данных будет очищен и на его место будет установлены файлы из резервной копии. После копирования файлов из образа система Dionis-NX продолжит загружаться. После окончания загрузки это будет уже восстановленная из резервной копии система. Вторичная перезагрузка не требуется.

Также можно производить восстановление данных "на лету" в неактивные слоты данных. Восстановление из резервной копии в текущий слот данных (активный) невозможно.

```
Router# os data restore data~l flash0.1:/backup—dionisnx—1.0—0—121105—104228.dbu
```

Где «data~l» - имя слота данных, в который будут записаны восстановленные данные.

При любом способе восстановления надо помнить, что данные, содержащиеся в целевом слоте данных, будут уничтожены и на их место будут записаны данные из резервной копии.

57.2 Проверка файловых систем

Несмотря на то, что файловая система EXT3 (системный раздел и раздел данных системы Dionis-NX) является журналируемой, в некоторых нештатных ситуациях требуется принудитель- ная проверка файловой системы. Такая проверка по умолчанию будет проводиться каждый месяц, либо после каждых 30 случаев монтирования файловой системы. Дополнительно администратор может запланировать принудительную проверку файловых систем с помощью команды:

```
# schedule fsck
```

Во время следующей загрузки системы, файловые системы будут принудительно проверены на ошибки. Смонтированные файловые системы на работающей системе не могут быть проверены в силу технологических особенностей процесса. Поэтому для проверки требуется перезагрузка.

57.3 Безопасная очистка внешнего носителя

Если внешний носитель (флеш, дискета) содержит конфиденциальную ключевую информа-цию, и возникает необходимость безопасно её удалить без возможности восстановления, то это можно сделать с помощью команды 'clear removable'. Данная команда заполняет всё пространство носителя случайными данными. Для дальнейшего использования данного носителя его необхо-димо будет отформатировать с помощью команды 'format' (см. ниже).

Формат команды безопасной очистки внешнего носителя:

```
clear removable <flashN>|<floppyN> [repeat <n>]
```

Если указан параметр "repeat", то процедура очистки будет выполнена указанное число раз.

57.4 Форматирование внешнего носителя

Форматирование внешнего носителя выполняется с помощью команды:

```
format <flashN>|<floppyN>
```

При форматировании флеш-носителя создаётся один раздел, занимающий всё пространство носителя. На носителе создаётся файловая система FAT.

57.5 Сброс паролей в начальное значение

Если системные пароли были по какой-то причине утеряны, они могут быть сброшены в свои начальные значения. Начальное значение пароля для учетной записи консольного доступа cli - cli. Для администратора adm начальное значение пароля - adm. Для других учетных записей пароли не могут быть сброшены.

Сброс паролей может быть произведен с помощью сервисного (установочного) флеш-диска. Для этого необходимо загрузиться с сервисного флеш-диска и выбрать пункт меню "Обслуживание системы -> Сброс паролей в начальное значение".

Плата "Сторож" в рабочем режиме (режим JL) предотвращает загрузку с внешних носителей, соответственно получение физического доступа к системе (без вскрытия корпуса) не означает возможность сброса паролей.

58. Приложение

58.1 Примеры конфигураций

58.1.1 Конфигурация по-умолчанию

Конфигурация по-умолчанию содержит следующие настройки:

- Временная зона соответствует Москве;
- Имя хоста задано как DionisNX;
- Включены настройки TCP/IP-стека по умолчанию;
- Запрещена маршрутизация некорректных пакетов;
- Настроен один интерфейс со статическим адресом 192.168.1.1/24;
- Созданы (но не применены) классы QoS, соответствующие классам изделия Dionis-LX;
- Сервис протоколирования настроен по умолчанию;
- Минимально настроен (но выключен) сервис DNS;
- Минимально настроен (но выключен) сервис NTP;
- Включен сервис SSH для оператора cli;
- Включена маршрутизация пакетов.

```
timezone MSK—3
!
hostname DionisNX
!
ip path — mtu—discovery
ip tcp ecn server—mode
ip tcp selective—ack
ip tcp syncookies
ip tcp timestamps
ip tcp window—scaling
!
ip access—group no—invalid forward
ip access—list no—invalid
deny state invalid
!
 ip class—map prt0
   match tos 0/0xe0
!
ip class—map prtl
 match tos 0x20/0xe0
!
ip class—map prt2
 match tos 0x40/0xe0
```

```
 ip class—map prt3
match tos 0x60/0xe0
 !
 ip class—map prt4
match tos 0x80/0xe0
 !
 ip class—map prt5
match tos 0xa0/0xe0
 !
 ip class—map prt6
match tos 0xc0/0xe0
 !
 ip class—map prt7
match tos 0xe0/0xe0

 ip        policy—map        prio
class    prt0    rate    1kbit    ceil      10000mbit    priority    7
class    prt1    rate    1kbit    ceil      10000mbit    priority    6
class    prt2    rate    1kbit    ceil      10000mbit    priority    5
class    prt3    rate    1kbit    ceil      10000mbit    priority    4
class    prt4    rate    1kbit    ceil      10000mbit    priority    3
class    prt5    rate    1kbit    ceil      10000mbit    priority    2
class    prt6    rate    1kbit    ceil      10000mbit    priority    1
class prt7 rate 1kbit ceil              10000mbit priority 0
!
 interface ethernet 0
ip address 192.168.1.1/24
enable

 service log
log
  trace all
  size 262144 131072
alert beep

 controller serial 0
listen
  speed 115200

 service dns
view  default
zone .
auto static

 service ntp
  server     0.ru.pool.ntp.org
server       1.ru.pool.ntp.org
server 2.ru.pool.ntp.org
```

```
 server 3.ru.pool.ntp.org
!
 service ssh
enable
!
 ip forwarding
```

58.1.2 Пример файерволла

```
 timezone MSK—4
session timeout adm none
!
 hostname raul
ip resolver domain cuba.int
ip resolver nameserver 192.168.33.254
!
 ip path—mtu—discovery
ip tcp ecn server—mode
ip tcp selective—ack
ip tcp syncookies
ip tcp timestamps
!
 ip access—list inl6
  deny tcp dst 192.168.33.254 dport 22
permit dst 192.168.33.0/24
permit dst 192.168.32.0/24
permit dst 192.168.16.0/24
deny
 !
 ip access—list int33
  deny tcp dport 3127 dst 192.168.33.254
 !
 ip access—list outside
permit dst 195.220.32.68 tcp dport 22 syn
deny tcp syn
permit state established
permit state invalid
permit state related
deny
 !
 ip nat—list masq
  nat tcp dport 22 dnat  ip  192.168.33.160  port  22
nat src 192.168.33.0/24 snat ip 195.220.32.68
 !
 ip nat—list masql6
```

```
 nat src 192.168.33.0/24 masquerade
 !
 ip nat—list squid
 exclude in tcp dport 80 dst 192.168.33.254
 nat tcp dport 80 src 192.168.33.0/24 redirect port 3127
 !
 ip nat—list squid—test
 nat tcp dport 80 src 192.168.33.22/32 redirect port 3127
 !
 interface ethernet 0
ip address 195.220.32.68/27
ip access—group outside in
ip nat—group masq
enable
 !
 interface ethernet 1
ip address 192.168.16.58/24
ip access—group inl6 in
enable
 !
 interface ethernet 2
ip address 192.168.33.254/24
ip access—group int33 in
ip nat—group squid
enable

 ip route 0.0.0.0/0 195.220.32.65
 ip route 192.168.0.0/24 192.168.16.1
 ip route 192.168.32.0/24 192.168.16.1

 service log
log
 alert beep
 size 262144 131072
trace
 !
 service dns
 acl Cuba 192.168.33.0/24
 acl  netl6  192.168.16.0/24
 acl net32 192.168.32.0/24
 acl nets netl6 net32 cuba localips
 log all info
 allow query nets
 allow query—cache nets
 allow recursion nets
 allow transfer none
 limit cache—size 10000000
 limit journal—size 10000000
```

```
  listen localips
notify no
view default
zone .
     auto weekly
    zone forward 16.168.192.in—addr.arpa.
    forwarders 192.168.16.3 192.168.16.4
    zone forward bubblegum.int.
    forwarders 192.168.16.3 192.168.16.4
    zone forward bubblegum.net.
     forwarders 192.168.16.3 192.168.16.4 192.168.16.1
    zone forward bubblegum.ru.
     forwarders 192.168.16.3 192.168.16.4 192.168.16.1
    zone master 33.168.192.in—addr.arpa.
    ttl 604800
     soa master raul.cuba.int. admin root@raul.cuba.int. refresh 604800 retry 86400 expire 2419200
    negttl 604800
    ns raul.cuba.int.
    ptr 1 fidel.cuba.int.
    ptr 160 havana.cuba.int.
    ptr 254 raul.cuba.int.
    ptr 3 pkunistan.cuba.int.
    ptr 6 vova—ipsec.cuba.int.
    update
    zone master cuba.int.
     ttl 604800
     soa master raul admin root@raul.cuba.int negttl 604000
a 192.168.33.1 fidel
a 192.168.33.160 havana
a 192.168.33.254 raul
a 192.168.33.3 pkunistan
a 192.168.33.6 vova—ipsec
ns raul
update
enable
  !
 service dhcp
listen ethernet 2
  broadcast—address 192.168.33.255
default—lease—time 345600
domain—name cuba.int
domain—search cuba.int
domain—search bubblegum.int
gateway 192.168.33.254
max—lease—time 400000
min—lease—time 86400
subnet—mask 255.255.255.0
name—server 192.168.33.254
```

```
  host fidel
ip 192.168.33.1
mac 00:22:b0:51:16:37
host libcode
ip 192.168.33.2
mac 00:10:f3:04:18:63
host peter—fix
ip 192.168.33.4
mac f4:6d:04:72:0d:01
host pkunistan
ip 192.168.33.3
mac c8:60:00:61:41:77
host vova—ipsec
ip 192.168.33.6
mac 08:00:27:3f:fd:70
host white
ip 192.168.33.7
mac 00:lb:21:0d:cl:a6
host xos
ip 192.168.33.5
mac e0:cb:4e:62:29:16
subnet 192.168.33.0/24
range 192.168.33.10 192.168.33.220
enable
  !
 service proxy
  admin—email admin@bubblegum.ru
  listen 192.168.33.254 3128 intercept
  acl badl dstdomain sex.ru
  acl lan dst 192.168.33.0/24
  acl net33 src 192.168.33.0/24
  acl nolog srcdom—regex host.*
  acl nolog srcdom—regex pkunistan.*
  acl nolog srcdom—regex rdtsc.*
  acl ul uri .ul
  log access acls ! nolog
  log cache high
  cache limit disk max 500
  cache limit disk min 0
  cache limit memory max 16
  cache replacement—policy disk gdsf
  cache replacement—policy memory gdsf
  cache type aufs 16 4096
  refresh .\\.swf$ 10000 90% 20000
  refresh .\\.bmp$ 10000 90% 20000
  refresh .\\.png$ 10000 90% 20000
  refresh .\\.gif$ 10000 90% 20000
  refresh .\\.mpg$ 10000 90% 20000
```

```
 refresh .\\.avi$ 10000 90% 20000
 caching deny lan
 caching permit all http—
 access deny badl http—
 access permit net33
 http—access deny all
 enable
 !
service ntp
 listen       192.168.33.254
server      0.ru.pool.ntp.org
server      1.ru.pool.ntp.org
server      2.ru.pool.ntp.org
server 3.ru.pool.ntp.org
 !
service ssh
 listen 192.168.33.254 22
permit—adm—login
enable
 !
 ip forwarding
```

58.1.3 Пример использования VRF-Lite совместно с OSPF

Ниже представлены основные настройки конфигурации узлов для Схемы 1 главы VRF данного руководства.

Пример настройки NET2_1:

```
 interface ethernet 2
enable
 !
 interface ethernet 2.1
description "NET2.1"
ip address 100.0.0.99/24
enable
 !
 router ospf
 network 100.0.0.0/16 area 0
```

Пример настройки NET2_2:

```
 interface ethernet 2
enable
 !
 interface ethernet 2.2
description "NET2.2"
ip address 100.0.1.99/24
enable
```

```
!
router ospf
 network 100.0.0.0/16 area 0
!
```

Пример конфигурации NX1:

```
 interface ethernet 0
description "Link to NX2"
ip address 10.0.12.1/24
enable
 !
 interface ethernet 1
enable
 !
 interface ethernet 1.1
description 'To NET_1.1"
ip address 192.168.0.1/24
enable
 !
 interface ethernet 1.2
description 'To NX2_vrfl"
ip address 192.168.12.1/24
enable
 !
 interface ethernet 2
enable
 !
 interface ethernet 2.1
description 'To NET_2.1"
ip address 100.0.0.1/24
enable
 !
 interface ethernet 2.2
description 'To NX2_vrf2"
ip address 100.0.12.1/24
enable
 !
 interface vrf 1
slave ethernet 1.1
slave ethernet 1.2
enable
 !
 interface vrf 2
slave ethernet 2.1
slave ethernet 2.2
enable
 !
 router ospf vrf 1
```

```
network 192.168.0.0/16 area 0
!
router ospf vrf 2
network 100.0.0.0/16 area 0
```

Пример конфигурации NX2:

```
interface ethernet 0
description "Link to NX1"
ip address 10.0.12.2/24
enable
!
interface ethernet 1
enable
!
interface ethernet 1.1
description 'To NX3_vrfl"
ip address 192.168.23.2/24
enable
!
interface ethernet 1.2
description 'To NXl_vrfl"
ip address 192.168.12.2/24
enable
!
interface ethernet 2
enable
!
interface ethernet 2.1
description 'To NX3_vrf2"
ip address 100.0.23.2/24
enable
!
interface ethernet 2.2
description 'To NX_vrf2"
ip address 100.0.12.2/24
enable
!
interface ethernet 3
description "Link to NX3"
ip address 10.0.23.2/24
enable
!
interface vrf 1
slave ethernet 1.1
slave ethernet 1.2
enable
!
interface vrf 2
```

```
 slave   ethernet  2.1
slave ethernet 2.2
enable
 !
 router ospf vrf 1
network 192.168.0.0/16 area 0
 !
 router ospf vrf 2
network 100.0.0.0/16 area 0
```

Пример конфигурации NX3:

```
 interface ethernet 0
description "Link to NX2"
ip address 10.0.23.3/24
enable
 !
 interface ethernet 1
enable
 !
 interface ethernet 1.1
description 'To NX2_vrfl"
ip address 192.168.23.3/24
enable
 !
 interface ethernet 1.2
description 'To NET_1.2"
ip address 192.168.1.1/24
enable
 !
 interface ethernet 2
enable
 !
 interface ethernet 2.1
description 'To NX2_vrf2"
ip address 100.0.23.3/24
enable
 !
 interface ethernet 2.2
description 'To NET_2.2"
ip address 100.0.1.1/24
enable
 !
 interface ethernet 3
ip address dhcp
enable
 !
 interface vrf 1
```

```
 slave  ethernet  1.1
slave ethernet 1.2
enable
 !
 interface vrf 2
slave ethernet 2.1
slave ethernet 2.2
enable
 !
 router ospf vrf 1
network 192.168.0.0/16 area 0
 !
 router ospf vrf 2
network 100.0.0.0/16 area 0
```

58.1.4 Пример реализации MPLS-L3VPN

Ниже представлены основные настройки конфигурации узлов для Схемы 2 главы VRF данного руководства.

Основная конфигурация C3PO-1:

```
 interface ethernet 0
description "to rl"
ip address 192.168.1.1/24
enable
 !
 interface ethernet 3
 !
 interface loopback 0
ip address 99.0.0.1/32
enable
 !
 router bgp 5227
bgp router—id 99.0.0.1
neighbor 192.168.1.2 remote—as 5227
neighbor 192.168.1.2 update—source 192.168.1.1
!
 address—family ipv4 unicast
network 5.1.0.0/24 route—map rm—nh
network 99.0.0.1/32
exit
 !
 router access—list al—any permit any
 !
 router route—map rm—nh permit 10
```

```
 match ip address al—any
set ip next—hop 99.0.0.1
```

Основная конфигурация C3PO-2:

```
 interface ethernet 0
description "to r4"
ip address 192.168.2.2/24
enable
 !
 interface loopback 0
ip address 99.0.0.4/32
enable
 !
 router bgp 5228
bgp router—id 99.0.0.4
neighbor 192.168.2.1 remote—as 5228
neighbor 192.168.2.1 update—source 192.168.2.2
!
 address—family ipv4 unicast
network 5.4.2.0/24 route—map rm—nh
network 99.0.0.4/32
exit
 !
 router access—list al—any permit any
 !
 router route—map rm—nh permit 10
match ip address al—any
set ip next—hop 99.0.0.4
```

Основная конфигурация R2D2-1:

```
 interface ethernet 0
description "to rl"
ip address 192.168.3.2/24
enable
 !
 interface ethernet 3
 !
 interface loopback 0
ip address 99.0.0.2/32
enable
 !
 router bgp 5225
bgp router—id 99.0.0.2
neighbor 192.168.3.1 remote—as 5225
neighbor 192.168.3.1 update—source 192.168.3.2
```

```
address—family ipv4 unicast
network 5.1.0.0/24 route—map rm—nh
network 99.0.0.2/32
exit
!
router access—list al—any permit any
!
router route—map rm—nh permit 10
match ip address al—any
set ip next—hop 99.0.0.2
set metric 98
```

Основная конфигурация R2D2-2:

```
interface ethernet 0
description "to r4"
ip address 192.168.4.2/24
enable
!
interface loopback 0
ip address 99.0.0.3/32
enable
!
router bgp 5224
bgp router—id 99.0.0.3
neighbor 192.168.4.1 remote—as 5224
neighbor 192.168.4.1 update—source 192.168.4.2
!
address—family ipv4 unicast
network 5.1.3.0/24 route—map rm—nh
network 99.0.0.3/32
exit
!
router access—list al—any permit any
!
router route—map rm—nh permit 10
match ip address al—any
set ip next—hop 99.0.0.3
```

Основная конфигурация rl:

```
mpls ip
!
interface ethernet 0
description "to C3PO—1"
ip address 192.168.1.2/24
mpls ip
enable
```

```
 interface ethernet 1
description "to r2"
ip address 10.0.1.1/24
mpls ip
enable
  !
 interface ethernet 2
description "to R2D2—1"
ip address 192.168.3.1/24
mpls ip
enable
  !
 interface loopback 0
ip address 1.1.1.1/32
mpls ip
enable
  !
 interface vrf 1
slave ethernet 0
mpls ip
enable
  !
 interface vrf 2
slave ethernet 2
mpls ip
enable
  !
 router bgp 5226
bgp router—id 1.1.1.1
bgp cluster—id 1.1.1.1
neighbor 2.2.2.2 remote—as 5226
neighbor 2.2.2.2 update—source 1.1.1.1
!
  address—family ipv4 unicast
no neighbor 2.2.2.2 activate
exit
!
  address—family ipv4 vpn
neighbor 2.2.2.2 activate
exit
  !
 router bgp 5227 vrf 1
bgp router—id 192.168.1.1
neighbor 192.168.1.1 remote—as 5227
neighbor 192.168.1.1 update—source 192.168.1.2
!
  address—family ipv4 unicast
```

```
  neighbor 192.168.1.1 next—hop—self
label vpn export 527
rd vpn export 5227:1
rt vpn both 52:27
export vpn
import vpn
exit
!
router bgp 5225 vrf 2
bgp router—id 192.168.3.1
neighbor 192.168.3.2 remote—as 5225
neighbor 192.168.3.2 update—source 192.168.3.1
!
address—family ipv4 unicast
neighbor 192.168.3.2 next—hop—self
label vpn export 525
rd vpn export 5225:1
rt vpn both 52:25
export vpn
import vpn
exit
!
router ospf
ospf router—id 1.1.1.1
redistribute static
network 0.0.0.0/4 area 0
!
router ldp
router—id 1.1.1.1
!
address—family ipv4
  discovery transport—address 1.1.1.1
  !
  iface ethernet 1
  exit
  !
exit
!
```

Основная конфигурация r2:

```
mpls ip
!
interface ethernet 0
description "to rl"
ip address 10.0.1.2/24
mpls ip
enable
```

```
 interface ethernet 1
description "to r4"
ip address 10.0.2.2/24
mpls ip
enable

 interface ethernet 2

 interface ethernet 3

 interface loopback 0
ip address 2.2.2.2/32
mpls ip
enable
  !
 router bgp 5226
bgp router—id 2.2.2.2
bgp cluster—id 2.2.2.2
neighbor 1.1.1.1 remote—as 5226
neighbor 1.1.1.1 update—source 2.2.2.2
neighbor 4.4.4.4 remote—as 5226
neighbor 4.4.4.4 update—source 2.2.2.2
!
  address—family ipv4 unicast
no neighbor 1.1.1.1 activate
no neighbor 4.4.4.4 activate
exit
!
  address—family ipv4 vpn
neighbor 1.1.1.1 activate
neighbor 1.1.1.1 route—reflector—client
neighbor 4.4.4.4 activate
neighbor 4.4.4.4 route—reflector—client
exit
  !
 router ospf
  ospf router—id 2.2.2.2
network 0.0.0.0/0 area 0
  !
 router ldp
!
  address—family ipv4
    discovery transport—address 2.2.2.2
    !
    iface ethernet 0
    exit
    i
```

```
  iface ethernet 1
  exit
  !
 exit
 !
 !
```

Основная конфигурация r4:

```
mpls ip
 !
interface ethernet 0
 description "to r2"
 ip address 10.0.2.4/24
 mpls ip
 enable
 !
interface ethernet 2
 description "to C3PO—2"
 ip address 192.168.2.1/24
 mpls ip
 enable
 !
interface ethernet 3
 description "to R2D2—2"
 ip address 192.168.4.1/24
 mpls ip
 enable
 !
interface ethernet 4
 !
interface loopback 0
 ip address 4.4.4.4/32
 mpls ip
 enable
 !
interface vrf 1
 slave ethernet 3
 mpls ip
 enable
 !
interface vrf 2
 slave ethernet 2
 mpls ip
 enable
 !
router bgp 5226
 bgp route—map delay—timer 1
 bgp router—id 4.4.4.4
```

```
 bgp cluster—id 4.4.4.4
timers bgp 2 6
 neighbor 2.2.2.2 remote—as 5226
neighbor 2.2.2.2 update—source 4.4.4.4
!
 address—family ipv4 unicast
no neighbor 2.2.2.2 activate
exit
!
 address—family ipv4 vpn
neighbor 2.2.2.2 activate
exit
 !
router bgp 5228 vrf 2
bgp router—id 192.168.2.1
timers bgp 2 6
 neighbor 192.168.2.2 remote—as 5228
neighbor 192.168.2.2 update—source 192.168.2.1
!
 address—family ipv4 unicast
neighbor 192.168.2.2 next—hop—self
label vpn export 528
rd vpn export 5228:1
rt vpn both 52:27
export vpn
import vpn
exit
 !
router bgp 5224 vrf 1
bgp router—id 192.168.4.1
neighbor 192.168.4.2 remote—as 5224
neighbor 192.168.4.2 update—source 192.168.4.1
!
 address—family ipv4 unicast
neighbor 192.168.4.2 next—hop—self
label vpn export 524
rd vpn export 5224:1
rt vpn both 52:25
export vpn
import vpn
exit
 !
router ospf
 ospf router—id 4.4.4.4
redistribute static
network 0.0.0.0/4 area 0
 !
router Idp
```

```
router—id 4.4.4.4
!
address—family ipv4
 discovery transport—address 4.4.4.4
 !
 iface ethernet 0
 exit
 !
exit
!
```

Лист регистрации изменений

Изм.	Номера листов (страниц)				Всего листов (страниц) в докум.	№ документа	Входящий № сопроводительного докум. и дата	Подп.	Дата
	измененных	замененных	новых	аннулированных					