«Томский государственный университет систем управления и радиоэлектроники» (ТУСУР)
Управление дополнительного образования института инноватики (УДО ИИ)

«Допустить к защите»_____

Заведующий кафедрой

.

Выпускная квалификационная работа

По программе профессиональной переподготовки «Специалист в области проектирования, строительства и эксплуатации инфокоммуникационных сетей»

Судаков Виктор Анатольевич

«Внедрение IPv6 в сети предприятия на основе серверов под ОС FreeBSD и маршрутизаторов Cisco»

Научный руководитель
Старший преподаватель кафедры автоматизированных систем управления (АСУ), начальник отдела информационно-технического сопровождения (ЦИТС) Э. Р. Абанеев

Томск – 2018

Оглавление

Часть I

Адресация IPv6 в сети предприятия

Задачей данной работы является систематизация практических знаний и умений по настройке протокола IPv6 в сети небольшого предприятия или домашней сети. Она рассчитана на системных администраторов, хорошо знакомых с протоколом IPv4 и его настройкой на маршрутизаторах фирмы Cisco Systems и серверах с Unix-подобными операционными системами семейства BSD.

Автор осознает, что в сетях предприятий и в домашних сетях в настоящее время широко используются технологии трансляции адресов (далее по тексту NAT). Назначая устройствам во внутренней сети адреса из приватного адресного пространства [1], администратор сети небольшого предприятия или домашней сети может думать, что еще долго не столкнется с проблемой нехватки IP адресов - главной проблемой, вызвавшей появление протокола IPv6 в конце 90-х годов. В конце концов, используя только адреса из RFC1918[1], можно адресовать свыше 17 миллионов хостов!

Однако, на взгляд автора, такие тенденции, как бурный рост рынка IoT устройств, простота автоматического назначения IP адресов в IPv6, поддержка Mobile IP (перемещение из одной сети в другую с сохранением постоянного IP адреса) [3], рост значимости приложений и прикладных протоколов, для которых необходима прямая связь по «реальному» IP-адресу между различными устройствами, а NAT является помехой (пиринговые сети, блокчейн, голос, видео) - все вышеперечисленное является стимулом к тому, чтобы системные администраторы начинали изучать IPv6 и внедрять в тестовых целях уже сейчас.

Пример стенда для изучения технологий IPv6 показан на Рис. 1 (стр. 11).

1. Общие принципы IPv6 адресации

Существуют различные типы адресов IPv6: одноадресные (Unicast), групповые (Anycast) и многоадресные (Multicast).

Адреса типа Unicast хорошо всем известны. Пакет, посланный на такой адрес, достигает в точности интерфейса, который этому адресу соответствует.

Адреса типа Anycast синтаксически неотличимы от адресов Unicast, но они адресуют группу интерфейсов. Пакет, направленный такому адресу, попадёт в ближайший (согласно метрике маршрутизатора) интерфейс. Адреса Anycast могут использоваться только маршрутизаторами.

Адреса типа Multicast идентифицируют группу интерфейсов. Пакет, посланный на такой адрес, достигнет всех интерфейсов, привязанных к группе многоадресного вещания.

Широковещательные адреса в IPv6 не используются, их задачи решаются средствами многоадресного вещания. Например, вместо широковещательного протокола ARP для поиска соседей по сети в IPv6 используется Neighbor Discovery Protocol (NDP), работающий по Multicast.

Unicast IPv6 адреса состоят из 64-битного префикса, используемого для маршрутизации, и 64-битного идентификатора, используемого для идентификации сетевого интерфейса хоста, подключенного к сети, например в адресе «2001:4860:4860::8888» сетевой частью является «2001:4860:4860:0», а интерфейсной «0:0:0:8888». Для простоты можно считать, что у оконечного сетевого устройства маска сети (хотя сам термин «маска сети» в IPv6 не используется, вместо него используется термин «префикс») всегда будет /64. На интерфейсах точка-точка более длинные префиксы использовать технически возможно, но не рекомендуется. Таким образом, IPv6 отменяет столь нелюбимые пользователями вычисления масок.

Также широко используются специальные адреса ::1 (в несокращенном виде 0:0:0:0:0:0:0:1) - эквивалент IPv4 адреса 127.0.0.1, loopback) и :: (в несокращенном виде 0:0:0:0:0:0:0:0) - так называемый IPv6 unspecified address, используемый устройством, которое ещё не имеет IPv6 адреса.

На OS FreeBSD при необходимости указания буквального IP адреса (например в адресной строке браузера) его следует указывать в квадратных скобках,

например

```
https://[2001:db8:85a3:8d3:1319:8a2e:370:7348]:443/
```

2. CIDR для IPv6

Для маршрутизации в Интернете используется блок глобальных Unicast адресов 2000::/3, что составляет 1/8 от всех возможных IPv6 адресов. Из этого блока выдаются крупные блоки RIR-ам (региональным интернет-регистратурам, например RIPE для Европы), RIR-ы раздают адреса LIR-ам (локальным интернет-регистратурам) блоками от /19 до /32 [6], а LIR-ы уже конечным пользователям, как показано на Рис. 2 (стр. 2).

Как и в IPv4, IPv6 сети можно агрегировать в более крупные блоки с целью уменьшения размеров таблиц маршрутизации, как показано на стр. 13

Хотя длина префикса может быть любой, на практике принято использовать префиксы, кратные 4 битам, по так называемой nibble boundary, так как один nibble соответствует одному символу в стандартной записи IPv6 адреса (Рис. 4).

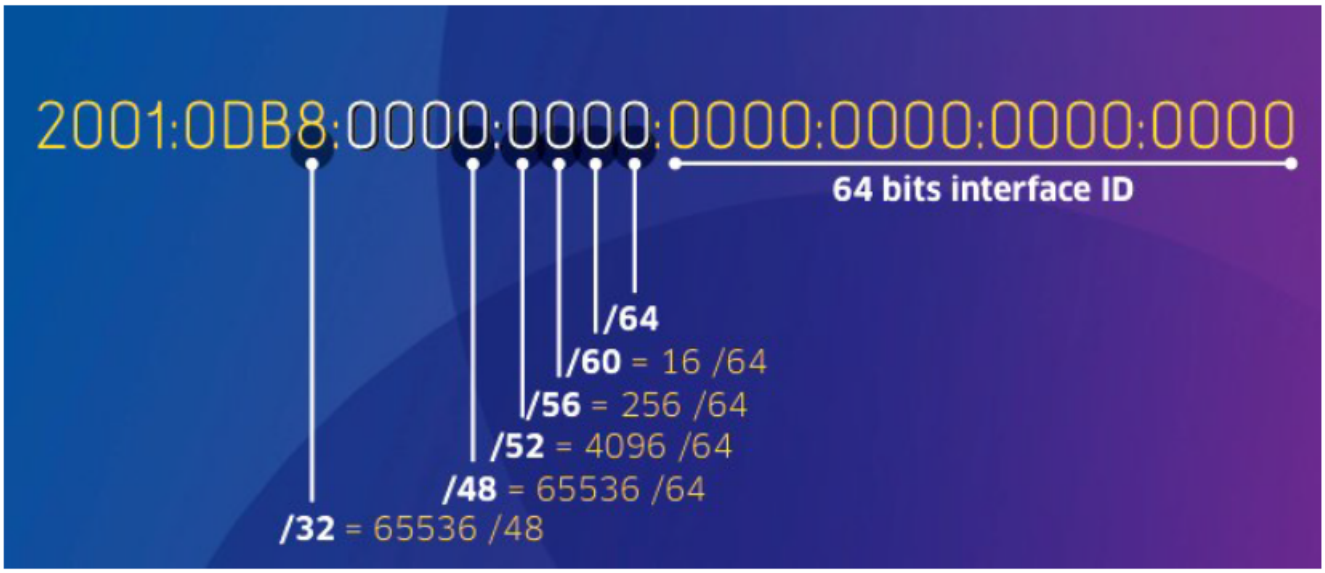


Рис. 4. Наиболее употребительные длины префиксов

Конечный пользователь может получить одну IPv6 сеть /64 (на один сег-

мент локальной сети, например домашний WiFi), а предприятиям обычно предоставляют блок /56 (256 отдельных сетей) или даже /48 (65536 отдельных сетей).

3. Рекомендации RIPE по планированию IPv6 адресации предприятия

В материалах вебинара RIPE[11], исходя из предположения, что предприятие получает блок адресов /48, предлагается четвертый секстет использовать для кодирования назначения сети. Для примера рассмотрим принцип планирования блока адресов 2001:470:ecba::/48. Назначение сетей можно закодировать следующим образом:

`2001:470:ecba:FLXX:/48`

где байт F обозначает функциональное назначение сети (0 = инфраструктура, 1 = сервера, 2 = офис, 3 = VPN, 4 = гостевая сеть и так далее до f включительно).

Байтом L можно закодировать 16 географических местоположений (0 = главный офис в Томске, 1 = филиал в Асино, 3 = филиал в Парабели и т.д.)

Байты XX задают номер сети определенного типа в определенном географическом местоположении. Можно использовать как номер этажа в здании.

Таким образом, сеть `2001:470:ecba:4102::/64` - это вторая (или на втором этаже) гостевая сеть в Асино. Такая самодокументирующаяся система позволяет быстро ориентироваться в большом количестве IPv6 сетей предприятия.

4. Dual-stack vs NAT64/DNS64

Точно неизвестно, когда мир перейдет на протокол IPv6. В ближайшем будущем протоколы IPv4 и IPv6 будут существовать совместно. Полный переход может занять многие годы. Специалисты IETF создали различные протоколы и инструменты, которые позволяют сетевым администраторам постепенно пе-

реводить свои сети на протокол IPv6. Методы перехода можно разделить на 3 категории[9]:

1. Двойной стек, позволяет протоколам IPv4 и IPv6 сосуществовать в одной сети. Устройства с двойным стеком одновременно работают с протокольными стеками IPv4 и IPv6.

2. Туннелирование — способ транспортировки IPv6-пакетов через IPv4-сеть. IPv6-пакет инкапсулируется внутри IPv4-пакета, как и другие типы данных. Подробнее технологии туннелирования рассмотрены в Параграфах 9 и 16.

3. Преобразование сетевых адресов 64 (NAT64) позволяет устройствам под управлением IPv6 обмениваться данными с устройствами под управлением IPv4 с помощью метода преобразования, похожего на метод преобразования из NAT для IPv4. IPv6-пакет преобразовывается в пакет IPv4-пакет и наоборот.

5. Варианты подключения к IPv6 Интернет. Выбор провайдера

По состоянию на октябрь 2018 года ни один томский оператор не предоставляет IPv6 физическим лицам.

По информации из Википедии[12], на октябрь 2018 года в России только два оператора предоставляют IPv6 своим абонентам: АО «Эр-Телеком» по технологии PPPoE и ПАО «МТС» мобильным абонентам.

Но этот факт не должен останавливать желающих экспериментировать с IPv6.

При отсутствии у провайдера поддержки IPv6 всегда имеется возможность настроить полноценное IPv6 подключение через туннельного брокера. Пакеты

IPv6 туннелируются через IPv4-Интернет до шлюза туннельного брокера, который уже маршрутизирует их в IPv6-Интернет. Настройка маршрутизации через туннельного брокера обычно не представляет сложности, так как на сайте туннельного брокера обычно можно найти пример конфигурации туннелей под различные операционные системы (Cisco IOS, Linux, FreeBSD, Windows и так далее). Также у туннельного брокера можно получить сеть /64 (предоставляется по умолчанию), а иногда бесплатный блок /56 или даже /48.

Примеры настроек через туннельного брокера приведены в Параграфах 9 и 16.

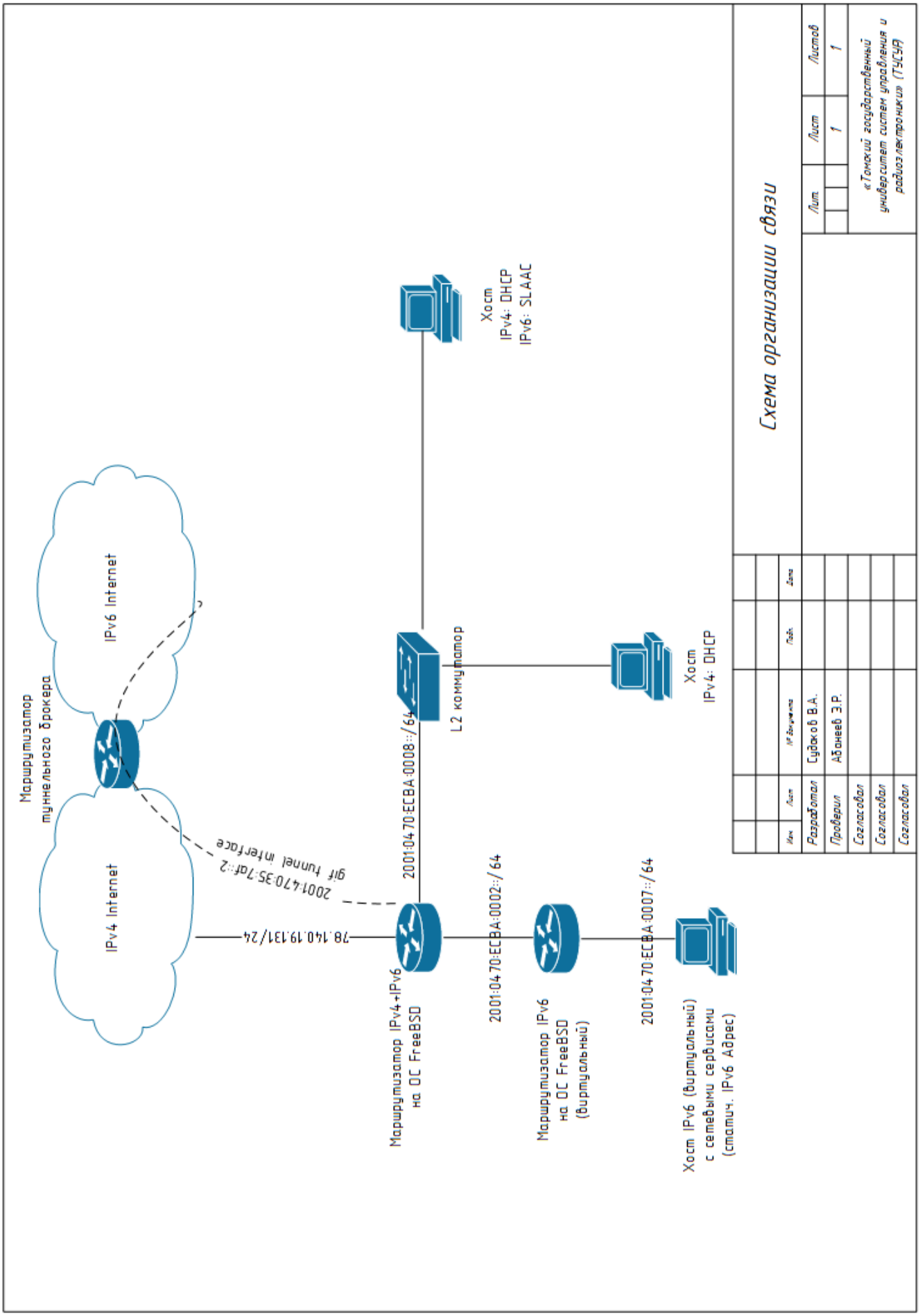


Рис. 1. Схема лабораторного стенда

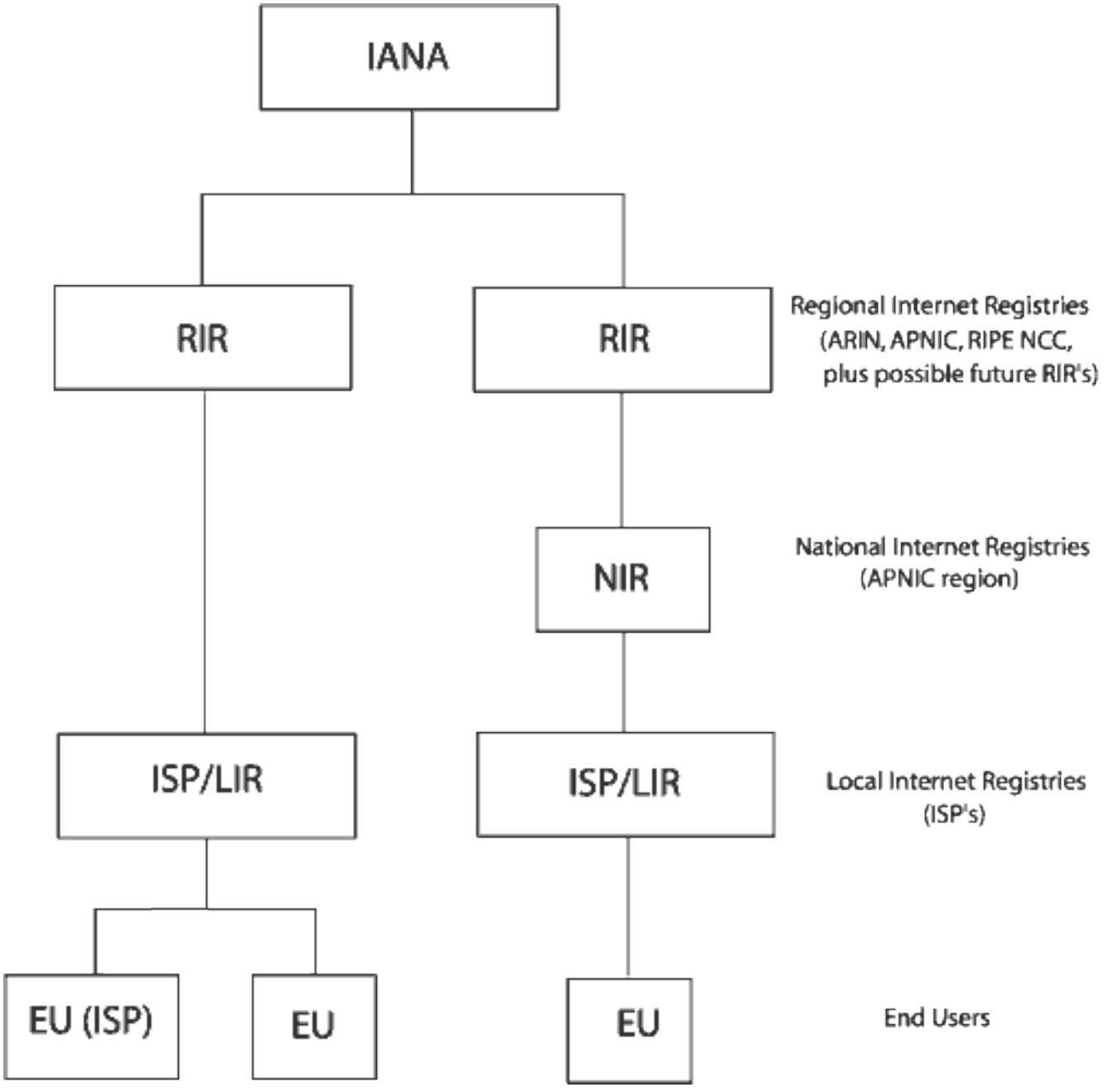
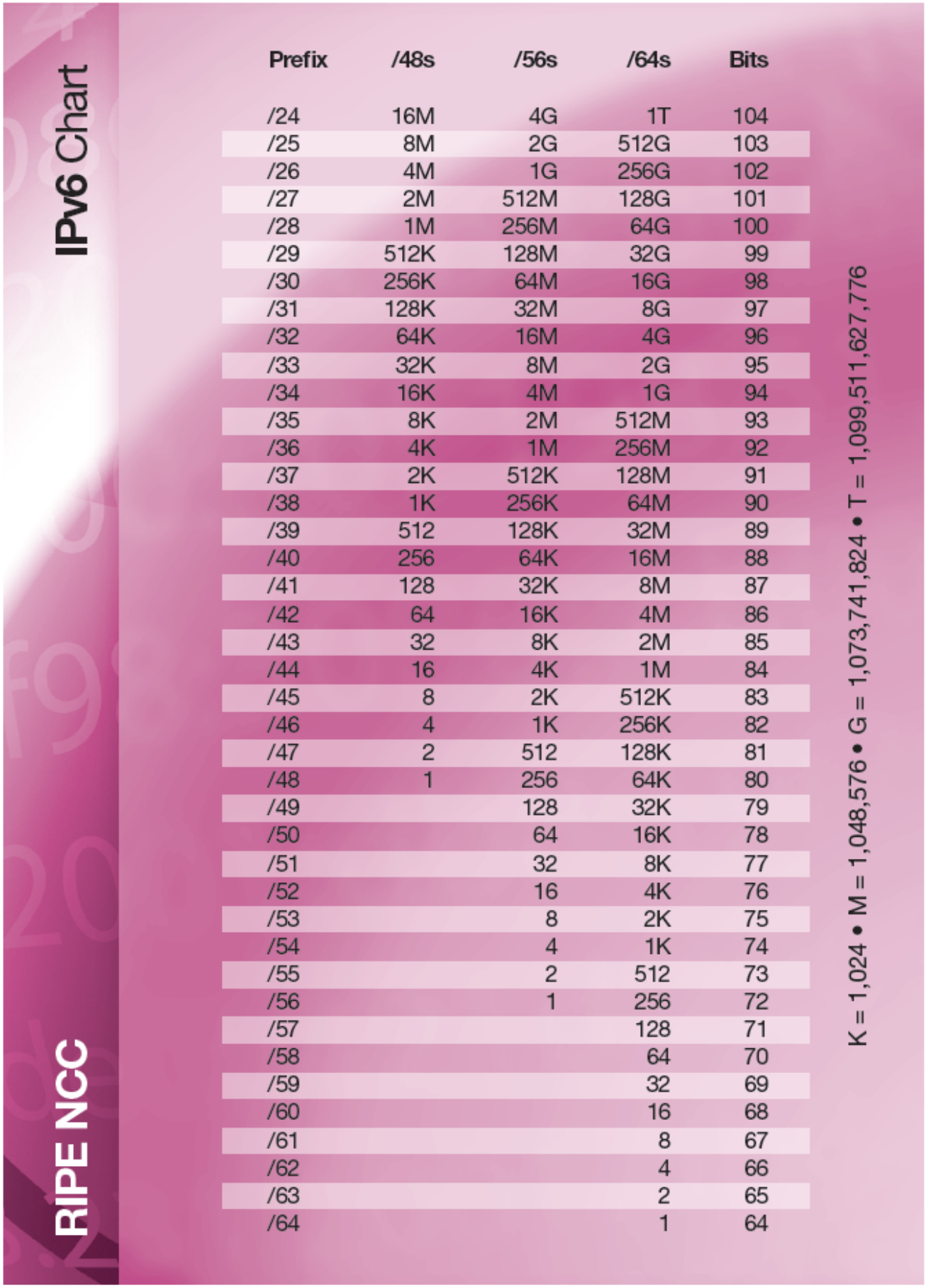


Рис. 2. Выдача IPv6 адресов



IPv6 Chart

RIPE NCC

Prefix	/48s	/56s	/64s	Bits
/24	16M	4G	1T	104
/25	8M	2G	512G	103
/26	4M	1G	256G	102
/27	2M	512M	128G	101
/28	1M	256M	64G	100
/29	512K	128M	32G	99
/30	256K	64M	16G	98
/31	128K	32M	8G	97
/32	64K	16M	4G	96
/33	32K	8M	2G	95
/34	16K	4M	1G	94
/35	8K	2M	512M	93
/36	4K	1M	256M	92
/37	2K	512K	128M	91
/38	1K	256K	64M	90
/39	512	128K	32M	89
/40	256	64K	16M	88
/41	128	32K	8M	87
/42	64	16K	4M	86
/43	32	8K	2M	85
/44	16	4K	1M	84
/45	8	2K	512K	83
/46	4	1K	256K	82
/47	2	512	128K	81
/48	1	256	64K	80
/49		128	32K	79
/50		64	16K	78
/51		32	8K	77
/52		16	4K	76
/53		8	2K	75
/54		4	1K	74
/55		2	512	73
/56		1	256	72
/57			128	71
/58			64	70
/59			32	69
/60			16	68
/61			8	67
/62			4	66
/63			2	65
/64			1	64

K = 1,024 • M = 1,048,576 • G = 1,073,741,824 • T = 1,099,511,627,776

Рис. 3. Префиксы в IPv6

Часть II

IPv6 на ОС FreeBSD

Будучи современной операционной системой, FreeBSD полностью поддерживает IPv6. Как и другие системы семейства *BSD, FreeBSD включает эталонную реализацию IPv6 от KAME. [2].

Так что система FreeBSD поставляется со всем, что нужно для работы и экспериментирования с IPv6.

6. Включение поддержки IPv6

Для работы FreeBSD в качестве IPv6-хоста необходимо добавить в /etc/rc.conf несколько переменных. Для активации IPv6 в режиме stateless address autoconfiguration (SLAAC) на интерфейсе fxp0 добавляем

```
ifconfig_fxp0_ipv6="inet6 accept_rtadv"
rtsold_enable="YES"
```

и даём команды `service netif restart` и `service rtsold start`. Система автоматически получит от маршрутизатора локальной сети IPv6 адрес, адрес шлюза по умолчанию и адреса DNS-серверов, если маршрутизатор настроен отдавать эту информацию.

Link-local адрес из диапазона fe80::/10 также будет назначен автоматически всем интерфейсам, на которых включен IPv6. Интерфейсу lo0 также будет автоматически назначен адрес ::1.

Для статического назначения глобального IP адреса и назначения маршрутизатора по умолчанию вручную, в /etc/rc.conf следует добавить, например

```
ifconfig_fxp0_ipv6="auto_linklocal 2001:470:ecba:3::50"
ipv6_defaultrouter="2001:470:ecba:3::1"
```

Где 2001:470:ecba:3::1 - IPv6-адрес маршрутизатора провайдера.

При использовании FreeBSD в режиме dual stack (одновременная поддержка IPv6 и IPv4, подробнее рассмотрена в Параграфе 4) можно настроить предпочитаемый протокол:

```
ip6addrctl_enable="YES"
ip6addrctl_policy="ipv6_prefer"
```

Web-браузеры на dual stack системах при обращении к сайту, у которого есть как IPv6, так и IPv4 адрес, обычно обращаются сразу по обоим протоколам, измеряют время отклика и выбирают тот протокол, через который время отклика меньше. Это может вызвать ситуацию, когда Web-браузер будет постоянно обращаться к сайту по IPv4 несмотря на наличие полной поддержки IPv6. Если такое поведение нежелательно, некоторые браузеры позволяют его отключить. Например, в Mozilla Firefox в about:config необходимо установить переменную `network.http.fast-fallback-to-IPv4=false`

7. Назначение IPv6 адресов хостам

Использование FreeBSD в качестве IPv6 маршрутизатора требует дополнительной конфигурации. Необходимо включить и сконфигурировать rtadvd (router advertisement daemon) на интерфейсах, с которых должны рассылаться router advertisements. В /etc/rc.conf включаем демон и перечисляем интерфейсы (интерфейс fxp0 не перечисляем, потому что в нашем примере он обращен к провайдеру).

```
rtadvd_enable="YES"
rtadvd_interfaces="fxp1 wlan0"
ipv6_gateway_enable="YES"
```

rtadvd имеет возможность [7] объявлять link MTU, IPv6-адреса DNS серверов, доменные суффиксы и другую информацию. Пример типового конфигурационного файла /etc/rtadvd.conf

```
fxp1:rdnss="2001:4860:4860::8888,2001:4860:4860::8844":\
:dnssl="sibptus.ru":
```

```
wlan0:rdnss="2001:4860:4860::8888,2001:4860:4860::8844":\
:dnssl="sibptus.ru":
```

В рассматриваемом примере хостам сети выдается доменный суффикс «sibptus.ru» и публичные DNS-сервера Google. После создания конфигурации демон следует активировать командой `service rtadvd start`.

8. DHCPv6

DHCP сервер в комплект базовой системы ОС FreeBSD не входит. Необходима установка пакета net/isc-dhcp43-server. По умолчанию пакет уже собран с поддержкой IPv6.

Назначение IPv6 адресов по DHCP имеет ряд преимуществ перед SLAAC[13]:

1. DHCP предоставляет возможность централизованного администрирования, то есть позволяет отследить, какие адреса, кем и когда использовались. Это может пригодится при аудите, тарификации и т.д.

2. Администраторы могут продолжать использовать привычные средства менеджмента адресного пространства, так как они как правило завязаны на DHCP.

3. DHCP умеет передавать гораздо большее количество параметров, чем Router Advertisements, например настройки для VoIP телефонов и т.д.

4. DHCP поддерживает динамическую регистрацию DHCP-клиентов в DNS.

5. DHCP поддерживает разные конфигурации для групп клиентов или даже отдельных клиентов, например для разных архитектур.

6. Контроль доступа, например отказ в выдаче адресов недоверенным хостам.

Тем не менее, анализ сетевых публикаций показал, что DHCPv6 на FreeBSD практически никем не используется, примеров успешных конфигураций нет. Автор также решил в рамках данной работы не рассматривать вопрос настройки DHCPv6 в ОС FreeBSD.

9. Маршрутизация через туннельного брокера

Ниже приводится пример конфигурации FreeBSD-маршрутизатора (переменные файла rc.conf) для работы через бесплатного туннельного брокера Hurricane Electric Free IPv6 Tunnel Broker [8]

```
gif_interfaces="gif0"
gifconfig_gif0="78.140.19.131 216.218.221.42"
ifconfig_gif0_ipv6="inet6 2001:470:35:7af::2 2001:470:35:7af::1\
 prefixlen 128 mtu 1480"
ipv6_defaultrouter="2001:470:35:7af::1"
ipv6_gateway_enable="YES"
```

Где 78.140.19.131 - внешний IPv4-адрес маршрутизатора, 216.218.221.42 - IPv4-адрес туннельного брокера. Линковочные IPv6-адреса туннельного интерфейса 2001:470:35:7af::2 и 2001:470:35:7af::1 предоставлены туннельным брокером. IPv6-маршрут по умолчанию установлен через туннель.

10. Динамическая маршрутизация на примере RIPng

В комплект ОС FreeBSD входит route6d (RIP6 Routing Daemon). Включается установкой переменной `route6d_enable="YES"` в /etc/rc.conf и в типовых конфигурациях не требует никакой настройки. Для объявления граничным маршрутизатором маршрута по умолчанию для остальных маршрутизаторов сети может быть использована опция -s в сочетании с наличием статического default route в таблице маршрутизации граничного маршрутизатора.

На Рис.5 показана работа протокола RIPng на FreeBSD.

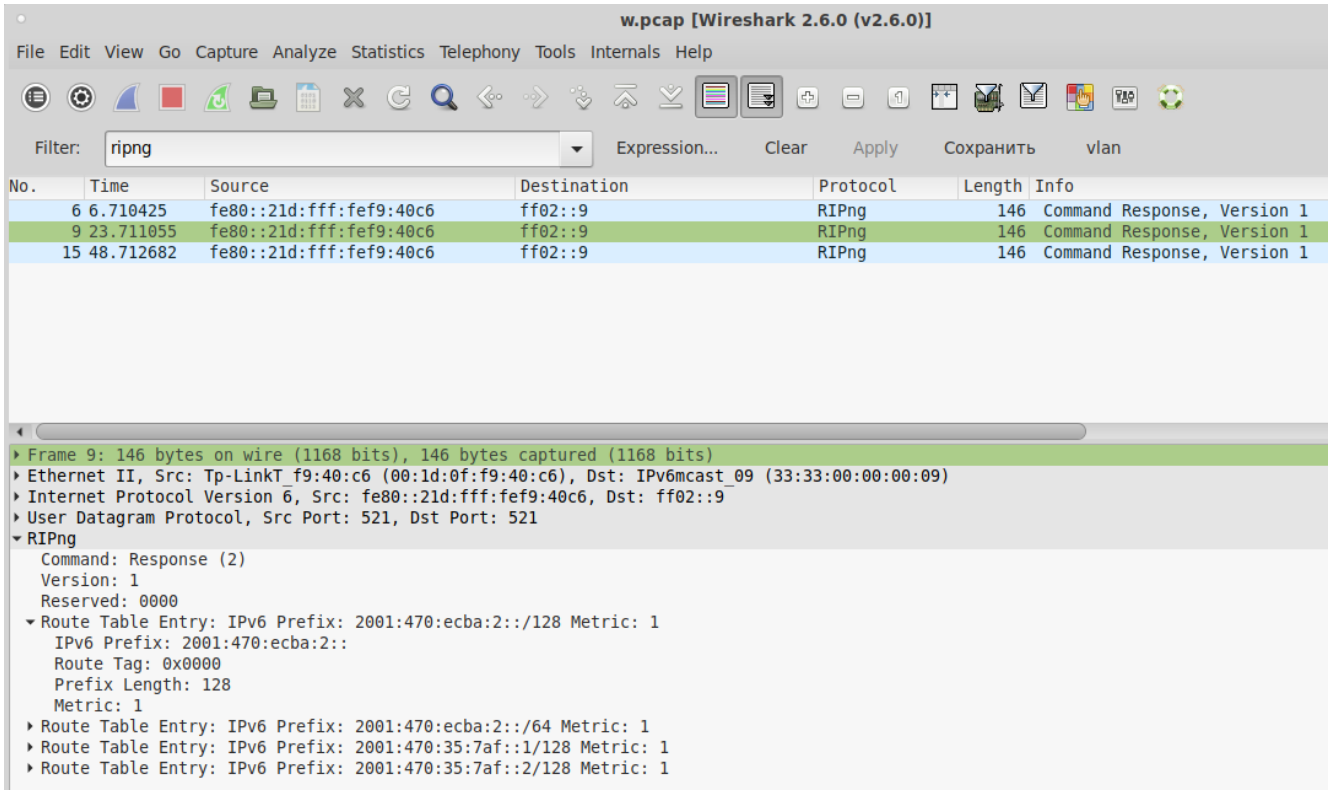


Рис. 5. RIP6

11. Сервисы NAT64 и DNS64

Что делать, если IPv6-only клиенту нужно обмениваться данными с IPv4-only сервером? Протоколы IPv6 и IPv4 несовместимы друг с другом, и такой обмен невозможен без специального посредника.

В роли такого посредника выступает технология NAT64+DNS64. NAT64 позволяет сделать отображение всего диапазона IPv4 в /96-диапазон адресов IPv6, а DNS64 может отдавать такие адреса клиентам (по сути производить подмену IPv4 адресов на IPv6 адреса из выбранного /96-диапазона).

Данная технология носит переходный характер и имеет серьезные ограничения, например она работает только в одну сторону (трафик должен инициировать IPv6-клиент); через нее не работают протоколы, передающие IP-адреса внутри полезной нагрузки (SIP, SDP, FTP и т.д.). Основная рекомендация по-

прежнему остаётся администраторам серверов поддерживать dual stack конфигурацию, если они не хотят отсечь перспективных IPv6-only клиентов.

Тем не менее, реализация данной технологии на ОС FreeBSD подробно рассмотрена в[14] на базе пакета net/tayga (Userland stateless NAT64 daemon) и DNS-сервера BIND (пакет dns/bind913). Также начиная с с версии FreeBSD 12 IPFW NAT64 module включена в ядро системы.

12. Использование ОС FreeBSD в качестве сервера в IPv6 сетях

Программное обеспечение из базовой системы FreeBSD (OpenSSH сервер и клиент, inetd, ftpd, unbound и др.) поставляется полностью готовым к работе в IPv6 сетях. Иногда посредством ключей командной строки, например -4 и -6, можно отключить или включить поддержку протоколов IPv4 и IPv6 соответственно, например команда `ssh -6 server` принудительно отключит IPv4 при обращении к серверу.

Программное обеспечение из коллекции портов (пакетов) может поддерживать либо не поддерживать IPv6, нужно принимать это во внимание при выборе ПО и внимательно читать документацию. Однако современное и распространенное ПО, такое как SMTP-сервер exim, IMAP-сервер dovecot, веб-сервера nginx и apache, DNS-сервера BIND и nsd, как правило, поддерживает IPv6 «из коробки», и при сборке пакета под FreeBSD поддержку IPv6 по умолчанию включают.

Некоторые программные продукты требуют небольшой дополлнительной конфигурации для работы в IPv6, например популярному Web-серверу nginx в конфигурационном файле нужно указать директиву

```
listen [::]:80;
```

только в этом случае будет создан tcp6-сокет. Убедиться в том, что сетевой сервис настроен на поддержку IPv6, можно командой

```
sockstat -l
```

В примере ниже показан nginx на dual stack сервере, работающий по обоим протоколам одновременно:

```
$ sockstat -l | egrep '^USER|nginx'
USER     COMMAND    PID   FD PROTO  LOCAL ADDRESS         FOREIGN ADDRES
www      nginx      64728 6  tcp4   *:80                  *:*
www      nginx      64728 7  tcp6   *:80                  *:*
root     nginx      64726 6  tcp4   *:80                  *:*
root     nginx      64726 7  tcp6   *:80                  *:*
$
```

Часть III

Маршрутизация IPv6 на устройствах Cisco

13. Включение поддержки IPv6

Использование Cisco в качестве маршрутизатора IPv6 ещё проще, чем ОС FreeBSD. Предположим, что провайдер предоставил нам линковочный адрес 2001:470:35:7af::2, адрес провайдерского шлюза по умолчанию (default gateway) 2001:470:35:7af::1, и для внутренних нужд нашего малого предприятия или домашней сети блок 2001:0470:ecba:1300::/56 из 256 сетей.

Интерфейс Gi0/0 маршрутизатора подключен к провайдеру. Интерфейс Gi0/1 подключен к локальной сети. Из блока 2001:0470:ecba:1300::/56 будем пока использовать только одну сеть 2001:0470:ecba:130f::/64 (а возможный диапазон сетей составит от 2001:0470:ecba:1300::/64 по 2001:0470:ecba:13ff::/64 включительно).

Конфигурация маршрутизатора:

```
ipv6 unicast-routing
!
interface GigabitEthernet0/0
description ISP
ipv6 address 2001:470:35:7af::2/64
!
interface GigabitEthernet0/1
description LAN
ipv6 address 2001:0470:ecba:130f::/64 eui-64
ipv6 address fe80::1 link-local
!
ipv6 route ::/0 2001:abc:33:44::1
```

14. Назначение IPv6 адресов хостам

Конфигурации из Параграфа 13 достаточно, чтобы маршрутизатор начал рассылать ICMPv6 router advertisements в сторону локальной сети. В качестве адреса шлюза по умолчанию маршрутизатор выдаст клиентам link-local адрес на соответствующем интерфейсе. Link-local адрес можно назначить вручную для красоты, либо предоставить маршрутизатору сформировать его автоматически.

15. DHCPv6

Сообщение ICMPv6 router advertisement содержит префикс, длину префикса и другие сведения для IPv6-хоста. Кроме того, такое сообщение указывает IPv6-хосту, как ему получить информацию по адресации. Сообщение ICMPv6 router advertisement может быть в одном из следующих 3 вариантов:

1. Только SLAAC. Устройство должно использовать префикс, длину префикса и шлюз по умолчанию, которые содержатся в сообщении ICMPv6 router advertisement. Другая информация недоступна с DHCPv6-сервера.

2. SLAAC и DHCPv6. Устройство должно использовать префикс, длину префикса и шлюз по умолчанию, которые содержатся в ICMPv6 router advertisement. На DHCPv6-сервере доступна и другая информация, например адрес DNS-сервера. Устройство получит эту дополнительную информацию в процессе поисков и запросов к DHCPv6-серверу. Этот процесс называется «DHCPv6 без запоминания состояний» (stateless DHCP), поскольку DHCPv6-серверы не выделяют и не отслеживают какие-либо назначения IPv6-адресов, а предоставляют дополнительную информацию, например об адресе DNS-сервера.

3. Только DHCPv6. Устройство не должно использовать информацию из

ICMPv6 router advertisement для пополнения своей информации об адресации. Вместо этого устройство будет использовать обычные процессы поисков и запросов к DHCPv6-серверам для получения всей своей информации об адресации. Такая информация включает в себя индивидуальный адрес IPv6, длину префикса, адрес шлюза по умолчанию и адреса DNS-серверов. В этом случае DHCPv6-сервер работает как DHCP-серверу для IPv4. DHCPv6-сервер выделяет и отслеживает IPv6-адреса, чтобы не назначать один и тот же IPv6-адрес на нескольких устройствах.

Маршрутизаторы отправляют сообщения ICMPv6 router advertisement, используя link-local адрес в качестве IPv6-адреса источника.

Рассмотрим пример конфигурации stateless DHCPv6 на маршрутизаторе Cisco. В данной конфигурации передается также IPv6 адрес NTP сервера.

```
ipv6 dhcp pool TEST
 dns-server 2001:470:ecba:2::1 2001:470:ecba:3::1
 sntp address 2001:470:ecba:2::1
 domain-name sibptus.ru
!
interface GigabitEthernet0/1
 ipv6 dhcp server TEST
 ipv6 nd other-config flag
```

16. Маршрутизация через туннельного брокера

Ниже приводится пример конфигурации маршрутизатора Cisco для работы через бесплатного туннельного брокера Hurricane Electric Free IPv6 Tunnel Broker [8]

```
!
```

```
interface Tunnel0
 description Hurricane Electric IPv6 Tunnel Broker
 no ip address
 ipv6 enable
 ipv6 address 2001:470:35:7af::2/64
 tunnel source 78.140.19.131
 tunnel destination 216.218.221.42
 tunnel mode ipv6ip
!
ipv6 route ::/0 Tunnel0
!
```

Где 78.140.19.131 - внешний IPv4-адрес маршрутизатора, 216.218.221.42 - IPv4-адрес туннельного брокера. IPv6-адрес туннельного интерфейса Tunnel0 2001:470:35:7af::2 предоставлен туннельным брокером. IPv6-маршрут по умолчанию установлен через туннель.

17. Динамическая маршрутизация в сети предприятия на примере OSPFv3

Тому, кто знаком с работой протокола OSPFv2 в IPv4 сетях, настройка OSPFv3 в IPv6 сетях не представляет никакой сложности. Наоборот, настройка значительно упрощена, поскольку не используется команда `network` и инверсные маски, принадлежность интерфейса к области OSPF задается непосредственно на интерфейсе.

Приведем пример конфигурации для двух интерфейсов маршрутизатора:

```
ipv6 router ospf 1
 router-id 1.1.1.1
!
```

```
interface GigabitEthernet0/0
  ipv6 ospf 1 area 0
!
interface GigabitEthernet0/1
  ipv6 ospf 1 area 0
!
```

Часть IV

Вопросы безопасности

18. Privacy Extensions for IPv6 SLAAC

Существует мнение, что NAT повышает безопасность сети, скрывая внутренние адреса. Это мнение настолько же расхожее, насколько спорное. Как минимум NAT не задумывался изначально как мера безопасности.

Данному вопросу посвящен RFC5902[10], в котором сказано в частности следующее:

It is commonly perceived that a NAT box provides one level of protection because external hosts cannot directly initiate communication with hosts behind a NAT. However, one should not confuse NAT boxes with firewalls. As discussed in RFC4864, Section 2.2, the act of translation does not provide security in itself. The stateful filtering function can provide the same level of protection without requiring a translation function.

В IPv6, где необходимость в NAT отсутствует, для аналогичной цели используется технология использования случайного идентификатора для формирования IPv6 адреса[4] (вместо формирования из MAC-адреса[5], что делает IPv6 адреса непредсказуемыми и затрудняет отслеживание.

В ОС FreeBSD рекомендуется включить переменную `ipv6_privacy=YES` для формирования временных RFC4941-адресов. Временный адрес в выводе команды ifconfig система помечает как `temporary`. Хорошо видно, что он не образован от MAC-адреса интерфейса:

```
root@ipv6:~ # ifconfig fxp1
fxp1: flags=8943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX,MULTICAST> metric
  options=80028<VLAN_MTU,JUMBO_MTU,LINKSTATE>
  ether 58:9c:fc:01:aa:b9
  hwaddr 58:9c:fc:01:aa:b9
  inet6 fe80::5a9c:fcff:fe01:aab9%fxp1 prefixlen 64 scopeid 0x1
  inet6 2001:470:ecba:2:5a9c:fcff:fe01:aab9 prefixlen 64 autoconf
  inet6 2001:470:ecba:2:39f2:dde7:1111:511e prefixlen 64 autoconf tempor
```

```
  nd6 options=23<PERFORMNUD,ACCEPT_RTADV,AUTO_LINKLOCAL>
  media: Ethernet 10Gbase-T <full-duplex>
  status: active
root@ipv6:~ #
```

Технология[4] также защищает хосты от сканирования, потому что просканировать $2^{64} = 18446744073709551616$ IPv6-адресов в сегменте за разумное время невозможно, в отличие от IPv4 с типичным количеством адресов в сегменте 256.

Побочную же функцию NAT-устройств, которая заключается в запрете входящих пакетов из Интернет на хосты внутренней сети, если соединение не было инициировано изнутри, необходимо решать средствами, специально предназначенными для этой цели - межсетевыми экранами. Ниже мы рассмотрим примеры простейших конфигураций межсетевых экранов для stateless и stateful фильтрации.

19. Межсетевой экран для IPv6 на базе маршрутизатора Cisco

На граничных маршрутизаторах Cisco обычно используют технологию stateful inspection. Рассмотрим пример конфигурации, разрешающей входящий трафик на порт 25, любой исходящий трафик и возвратный трафик (симметричный исходящему).

```
ipv6 inspect name TEST tcp
ipv6 inspect name TEST udp
ipv6 inspect name TEST icmp
!
interface GigabitEthernet0/0
description ISP
```

```
 ipv6 inspect TEST out
 ipv6 traffic-filter INBOUND in
!
ipv6 access-list INBOUND
 sequence 10 permit ipv6 any any eq 25
!
!
```

Необходимо помнить, что Cisco IPv6 ACL подразумевает в конце невидимый deny, а перед ним невидимый permit для ICMPv6 neighbor discovery. Но если ACL содержит в конце явный deny, то невидимый permit для ICMPv6 neighbor discovery отсутствует.

20. Межсетевой экран для IPv6 на базе ОС FreeBSD

В комплекте ОС FreeBSD содержится несколько готовых наборов правил межсетевого экранирования: open, client, simple, closed, workstation и др.

Если включить в /etc/rc.conf конфигурацию client,

```
firewall_enable="YES"
firewall_type="client"
firewall_client_net_ipv6="2001:470:ecba:2::/64"
firewall_client_net="192.168.150.0/24"
```

то операционная система сформирует следующий набор правил, оптимизированных для защиты только самого хоста:

```
00100 allow ip from any to any via lo0
00200 deny ip from any to 127.0.0.0/8
00300 deny ip from 127.0.0.0/8 to any
00400 deny ip from any to ::1
```

```
00500 deny ip from ::1 to any
00600 allow ipv6-icmp from :: to ff02::/16
00700 allow ipv6-icmp from fe80::/10 to fe80::/10
00800 allow ipv6-icmp from fe80::/10 to ff02::/16
00900 allow ipv6-icmp from any to any ip6 icmp6types 1
01000 allow ipv6-icmp from any to any ip6 icmp6types 2,135,136
01100 allow ip from 192.168.150.0/24 to 255.255.255.255
01200 allow ip from me to 192.168.150.0/24
01300 allow ip from 192.168.150.0/24 to me
01400 allow ip from me to 2001:470:ecba:2::/64
01500 allow ip from 2001:470:ecba:2::/64 to me
01600 allow ip from fe80::/10 to ff02::/16
01700 allow ip from 2001:470:ecba:2::/64 to ff02::/16
01800 allow udp from fe80::/10 to me dst-port 546
01900 allow tcp from any to any established
02000 allow ip from any to any frag
02100 allow tcp from any to me dst-port 25 setup
02200 allow tcp from me to any setup
02300 deny tcp from any to any setup
02400 allow udp from me to any dst-port 53 keep-state
02500 allow udp from me to any dst-port 123 keep-state
65535 deny ip from any to any
```

Набор правил «simple» служит для защиты внутренней сети, когда FreeBSD выполняет роль маршрутизатора. В случае использования этого набора в конфигурационном файле /etc/rc.conf.local необходимо также перечислить внутренний и внешний интерфейсы и IP-адресацию на них.

Заключение

Из приведенных примеров конфигураций хорошо видно, что настройка IPv6 не представляет никакой сложности по сравнению с IPv4. Более того,

1. Сложность IPv6 адресов для запоминания и визуализации слишком преувеличена молвой. Многие служебные адреса выглядят лаконичнее, чем в IPv4, например ::1 вместо 127.0.0.1 или FF02::5 вместо 224.0.0.5. Для серверов также несложно выбирать короткие и легкозапоминаемые адреса, например 2001:4860:4860::8888 (адрес публичного DNS сервера Google) или 2001:470:20::2 (адрес публичного DNS сервера Hurricane Electric). А маршрутизатору по умолчанию в локальном сегменте вполне можно назначить легкозапоминаемый адрес типа 2001:0470:ECBA:8::1/64

2. Автоматическая конфигурация конечных устройств в IPv6 очень сильно упрощена по сравнению с IPv4.

3. Устройство, подключенное к одноранговой сети, например домашней или малого офиса, сразу же начинает работать на link-local адресах из диапазона FE80::/10[9].

4. Настройка динамических протоколов маршрутизации на оборудовании Cisco также сильно упрощена, поскольку не используется команда `network` и инверсные маски.

5. Практически отпала необходимость вычисления масок подсети, как и само понятие «маски». В пользовательском сегменте длина префикса всегда равна /64. Агрегация сетей в блоки по nibble boundary также очень наглядна и не требует вычислений.

6. Большое количество выделяемых провайдером конечному пользователю сетей (до $2^{(64-48)} = 65536$) даёт возможность организовать наглядную иерархическую структуру сети предприятия, подробнее этот вопрос рассмотрен в Разделе 3.

7. О дефиците «белых» адресов можно забыть навсегда. Можно дать «белый» адрес каждой кофеварке и IP-телефону и перестать беспокоиться, будет ли очередное устройство корректно работать через имеющуюся реализацию NAT. Конечно, эта радость несколько омрачается отсутствием повсеместной поддержки IPv6, но тем больше стимула у пользователей и системных администраторов оказывать давление на вендоров и поставщиков услуг с целью скорейшего внедрения IPv6 на всех устройствах и всех сервисах и сайтах.

Таким образом, автор призывает системных администраторов активнее осваивать и внедрять IPv6 в своих сетях, и как минимум экспериментировать с новым протоколом в своих домашних сетях и в лабораториях.

Список иллюстративного материала

Список литературы

1. Address Allocation for Private Internets. *https://www.rfc-editor.org/rfc/rfc1918.txt*

2. Руководство FreeBSD. Глава 27. Сложные вопросы работы в сети. *https://www.freebsd.org/doc/ru_RU.KOI8-R/books/handbook/network-ipv6.html*
3. Mobility Support in IPv6 *https://tools.ietf.org/html/rfc6275*
4. Privacy Extensions for Stateless Address Autoconfiguration in IPv6 *https://www.rfc-editor.org/rfc/rfc4941.txt*
5. Modified EUI-64 format *https://tools.ietf.org/html/rfc4291*
6. IPv6 Address Allocation and Assignment Policy *https://www.ripe.net/publications/docs/ripe-699*
7. rtadvd.conf – config file for router advertisement daemon *https://www.freebsd.org/cgi/man.cgi?query=rtadvd.conf*
8. Hurricane Electric IPv6 Tunnel Broker *https://tunnelbroker.net/*
9. Маршрутизация и коммутация CCNA: введение в сети *https://1277437.netacad.com/courses/638180*
10. IAB Thoughts on IPv6 Network Address Translation *https://www.rfc-editor.org/rfc/rfc5902.txt*
11. Preparing an IPv6 Addressing Plan *Материалы Online Вебинара RIPE, IPv6-addressing-plan-howto.pdf*
12. IPv6 Deployment *https://en.wikipedia.org/wiki/IPv6_deployment#Russian_Federa*

13. ISC DHCP and IPv6 – the DHCPv6 story *https://www.isc.org/blogs/isc-dhcp-and-ipv6-the-dhcpv6-story/*
14. NAT64 *https://taras.lviv.ua/freebsd-tayga-ipv6-only-to-ipv4-hosts/*