Статични маршрути. Протокол RIP. Конфигуриране под Linux

проф. д-р инж. Венета Алексиева

Основни моменти

- Конфигуриране на статични пътища
- Конфигуриране на RIPv1
- Zebra/Quagga
- Конфигуриране на RIPv2
- RIPng

Статични маршрути

/etc/rc.d/rc.inet1

 # route [add | del] [-net | -host] <target> [<if>]

Или с пакета **iproute2**

 # ip route add [prohibit] <target> via <if> [from <IP>] [src <IP>]

- ***add/ del*** - да се добави/премахне маршрут
- целта ***<target>*** е мрежа (***-net***) или отделна машина (***-host***) . Ако те не се укажат, по подразбиране се счита маршрут до хост.
- Целта ***<target>*** задава IP адрес на получателя, използваната маска (ако отсъства се приема маската за пълен клас IP адрес) и следващия маршрутизатор, към който трябва да се предаде пакетът, за да се маршрутизира към получателя.
- ***[<if>] -*** задава интерфейса, по който трябва да се предаде пакетът към целта
- ***prohibit -*** когато маршрутизаторът трябва да докладва, че тази мрежа е недостижима

Път по подразбиране

 # route add default gw <IP_GW>

или с пакета **iproute2**

 # ip route add default via <IP_GW>

- За работните станции това е задължителна конфигурация

Статични маршрути

/etc/gateways

- Във файла може да се укажат статични пътища, които да се добавят в маршрутната таблица.

net | host *name1* gateway *name2* metric *value* passive | active | external

- Използват се за:
  - указване на начален път по подразбиране
  - информация за маршрутизатори, които не анонсират своите маршрути

Конфигуриране на RIPv1

 # routed

- Изпълнява се като демон във фонов режим

1. Стартира се програмата **routed**
2. Тя издава заявка за обновявания
3. Започва очакване на съответните отговори

Обменяни съобщения

- Маршрутизаторът периодично изпраща копия на своята маршрутна таблица по директно свързаните към него мрежи.
- При получаване на заявка, **routed** формира отговор, базиран на информацията, поддържана в неговите вътрешни таблици (списък от известните маршрути и брой хопове спрямо изпращача).
- Обновяване на маршрутната таблица :
  - Не съществуват записи за дестинацията и метриката индицира, че дестинацията е достижима;
  - Източникът на пакета е същият като маршрутизатора в съществуващ запис в маршрутната таблица;
  - Съществуващият запис в маршрутната таблица не е бил обновяван за определено време (по подразбиране 90 sec) и маршрутът е по-оптимален от текущия маршрут;
  - Новият маршрут до дестинацията е по-кратък отколкото някой, съхранен в маршрутната таблица. Сравнението се прави на базата на метриките на маршрутите.

Таблици

1. Прилагат се обновяванията
2. Съхраняват се промените във вътрешните таблици на **routed**
3. Обновява се таблицата на ядрото на операционната система
4. Промените рефлектират върху следващия пакет-отговор, който ще се изпрати.

Стартиране на routed

 # routed

- От стартовия скрипт */etc/rc.d/rc.inet2*

 # routed -q

- указва да не се изпращат анонсирания на маршрути. В този случай машината единствено ще очаква обновявания от други маршрутизатори

 # routed -i *dev*

- ще игнорира наличието на съответния интерфейс ***dev***, като няма нито да получава, нито да изпраща обновявания през него

Routed и */etc/gateways*

net | host *name1* gateway *name2* metric *value* passive | active | external

Описаните в */etc/gateways* маршрутизатори са:

- **активни (active)** -участват в обмена на RIP информацията. Ако за определения период от време не се получи информация от активен маршрутизатор, асоциираните с него маршрути ще се премахнат от маршрутната таблица.
- **пасивни (passive)** - не обменят маршрутна информация. Маршрутите, свързани с пасивните маршрутизатори, се записват в маршрутната таблица на ядрото само по време на началното стартиране на демона.
- **външни (external)** –също са пасивни, но не се включват в маршрутната таблица на ядрото нито се изпращат в обновяванията. Тази възможност единствено информира **routed**, че друг маршрутизиращ процес ще инсталира подобен маршрут и алтернативните маршрути към подобна дестинация не трябва да се инсталират от своя страна.

Обновявания от **routed**

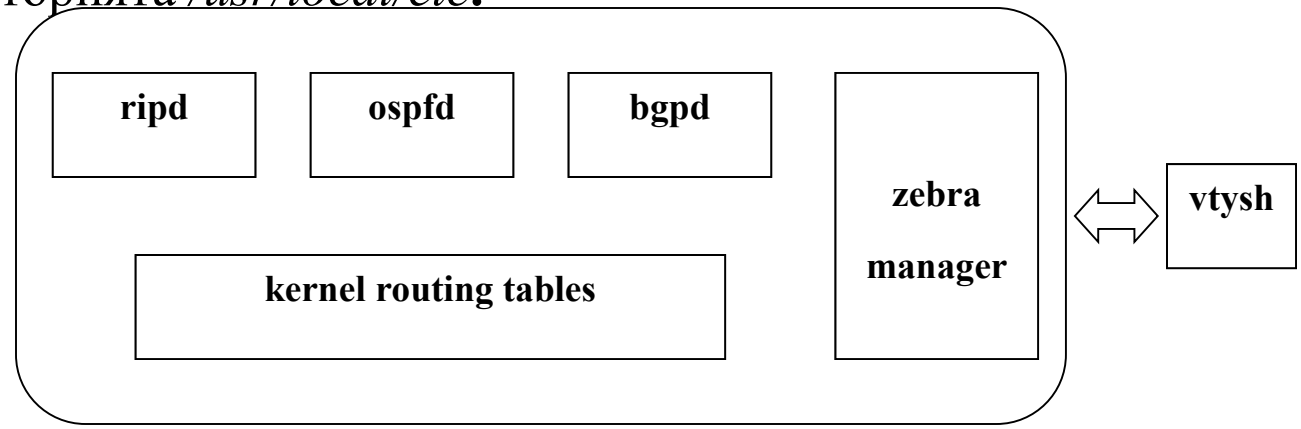
- по broadcast
- tcpdump

194.141.46.36 - PuTTY

```
15:25:04.698300 194.141.46.36.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.160}(1)[|rip] (DF)
15:25:06.875091 194.141.46.33.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.64}(1)[|rip] (DF)
15:25:20.514883 194.141.46.35.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.128}(1)[|rip] (DF)
15:25:24.174486 194.141.46.34.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.96}(1)[|rip] (DF)
15:25:34.551679 221.233.59.162.2855 > 194.141.46.187.1027:  udp 756 [tos 0x80]
15:25:34.698284 194.141.46.36.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.160}(1)[|rip] (DF)
15:25:36.874463 194.141.46.33.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.64}(1)[|rip] (DF)
15:25:50.513439 194.141.46.35.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.128}(1)[|rip] (DF)
15:25:54.173589 194.141.46.34.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.96}(1)[|rip] (DF)
15:26:04.698282 194.141.46.36.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.160}(1)[|rip] (DF)
15:26:06.873837 194.141.46.33.route > 194.141.46.63.route:  RIPv1-resp [items 2]
: {194.141.46.64}(1)[|rip] (DF)
```

Пакетът *zebra/ Quagga*

- Съвременната му дистрибуция е под наименованието *Quagga* (*http://www.nongnu.org/quagga*)
- За редистрибутирането на маршрутите между различните маршрутизиращи протоколи и за обновяването на маршрутната таблица на ядрото се използва отделен демон – *zebra* менажер.
- Всеки демон използва собствен конфигурационен файл, като в зависимост от протокола се задават различни конфигурационни команди.
- Конфигурационните файлове на пакета се намират по подразбиране в директорията */usr/local/etc*.



Свързване с всеки от демоните

- По **telnet** към съответен порт:
- Всеки от демоните има два основни режима на функциониране в интерактивен режим:
  - потребителски (*user*)
  - привилегирован (*enable*).

zebra	**2601/tcp**	**# zebra manager**
ripd	**2602/tcp**	**# RIP daemon**
ripngd	**2603/tcp**	**# RIP IPv6 daemon**
ospfd	**2604/tcp**	**# OSPF daemon**
bgp	**2605/tcp**	**# BGP-4 daemon**
ospf6d	**2606/tcp**	**# OSPF IPv6 daemon**

Стартиране на RIPv2

 # zebra -d

- -d е опция за работа във фонов режим

/usr/local/etc/ripd.conf

 # ripd –d

Конфигуриране на RIPv2

router rip

network <*IPNet*>/<*SM*>

network <*interface*>

- Разрешават се интерфейсите, чиято мрежова част на IP адресите съвпада с <***IPNet***>;
- Разрешава се интерфейса с име **<*interface*>**.

Конфигуриране на RIPv2

router rip

redistribute connected metric 1

default-metric 5
redistribute static
redistribute ospf

- Източниците на информация се задават като:
  - **static** – статично дефинирани маршрути;
  - **connected** – директно свързани мрежи;
  - **ospf** – протокол OSPF;
  - **bgp** – протокол BGP-4.
- Всеки източник има различна метрика, това налага привеждане на неговата метрика към тази на RIP протокола. За целта се указва стойност от 0-16, с която метрика полученият от друг източник маршрут ще се редистрибутира в RIP протокола.

Конфигуриране на RIPv2

router rip

neighbor 10.10.10.10

- За мрежи, не-поддържащи broadcast или multicast (като Frame Relay), е необходимо директно изпращане на съседен маршрутизатор, към когото да се изпращат RIP пакетите

router rip

passive-interface eth2

- маршрутизаторът няма да изпраща по broadcast или multicast пакети с изключение към съседните маршрутизатори, указани чрез командата **neighbor**

Автентикация при RIPv2

- RIP версия 2 поддържа два типа автентикация:
- **чист текст** - по подразбиране, но не се препоръчва да се използва когато има изисквания по отношение на сигурността
- **Message Digest 5 (MD5)** – когато е наложително криптиране на използваните пароли с MD5 алгоритъм.

Конфигуриране на автентикацията

key chain test

- Дефиниране на именована последователност от ключове (верига), които ще се използват за даден интерфейс;

key 1

- Дефиниране на самите ключове за веригата;

key-string mypassword

- Указване на паролата, която ще се изпраща в пакетите за автентициране;

interface eth0

ip rip authentication key-chain test

- Разрешаване на автентикацията за конкретен интерфейс и указване на ключовата верига;

ip rip authentication mode md5

- Указване вида на автентикацията – чист текст или MD5;

Въпроси ?

Благодаря за вниманието !