Linux Routers and Community Networks

Llorenç Cerdà-Alabern

http://personals.ac.upc.edu/llorenc
llorenc@ac.upc.edu

Universitat Politènica de Catalunya,
Barcelona, Spain
Lab 2: RIP and OSPF

Description
IOS fundamentals
Quagga set up
Basic commands
RIP review
RIP configuration
RIP Lab setup
OSPF review
OSPF configuration
OSPF Lab setup

Parts

I Introduction

II Lab 1: Basic Network Configuration

III Lab 2: RIP and OSPF

IV Lab 3: Firewall configuration

V Lab 4: Community Networks

VI Lab 5: Network Management
Lab 2: RIP and OSPF

Outline

- Description
- IOS fundamentals
- Quagga set up
- Basic commands
- RIP review

- RIP configuration
- RIP Lab setup
- OSPF review
- OSPF configuration
- OSPF Lab setup
Lab 2: RIP and OSPF

Description

Objectives

- **Quagga** is an open source routing software package that provides routing protocols support such as RIP, OSPF, IS-IS and BGP.
- Quagga is a branch of the original project called zebra.
- Quagga provides a **Cisco IOS-like** interface.
- In this lab we will review RIP and OSPF using Quagga.
Part III

Lab 2: RIP and OSPF

Outline

- Description
- **IOS fundamentals**
  - Quagga set up
  - Basic commands
  - RIP review
- RIP configuration
- RIP Lab setup
- OSPF review
- OSPF configuration
- OSPF Lab setup
Lab 2: RIP and OSPF

IOS fundamentals

Configuration modes

- **Two modes:**
  - *exec*: allows inspecting the router, e.g. show commands.
  - *configuration*: allows editing the router configuration.

- In configuration modes you edit the **running-config**.
- To **delete** commands from running config: preceded by `no`.
- The **prompt** indicates the mode, e.g. `>`, `#`, `#(config-if)`, etc.
- **Case insensitive.**
- `?` for help.
- **TAB** for command completion.
- Allows **abbreviated commands** as long there is no ambiguity. E.g. `sh` for show, or `conf term` for configure terminal.
- Quagga specific: accept **address/mask** notation, e.g. `10.0.0.1/24`. 
Part III

Lab 2: RIP and OSPF

Outline

- Description
- IOS fundamentals
- Quagga set up
- Basic commands
- RIP review
- RIP configuration
- RIP Lab setup
- OSPF review
- OSPF configuration
- OSPF Lab setup
Lab 2: RIP and OSPF

Quagga set up

Quagga daemons

- **zebra**: general configuration.
- **ripd**: RIP daemon.
- **ospfd**: OSPF daemon

- Use telnet to connect to the daemons:

```bash
root@OpenWrt:~# /etc/init.d/quagga start
quagga.init: Starting zebra ... done.
quagga.init: Starting ripd ... done.
quagga.init: Starting ospfd ... done.
root@OpenWrt:~# telnet localhost zebra
Entering character mode
Escape character is '^]'.
Hello, this is Quagga (version 0.99.22.3).
User Access Verification
Password: zebra
OpenWrt>
OpenWrt> enable
OpenWrt# ?
    clear  Reset functions
    configure  Configuration from vty interface
...
```
Lab 2: RIP and OSPF

Part III

Lab 2: RIP and OSPF

Outline

- Description
- IOS fundamentals
- Quagga set up
- **Basic commands**
- RIP review
- RIP configuration
- RIP Lab setup
- OSPF review
- OSPF configuration
- OSPF Lab setup
Lab 2: RIP and OSPF

Basic commands

Interfaces

- **Show Interfaces:**

  ```
  OpenWrt# show interface
  Interface br-lan is up, line protocol detection is disabled
   index 5 metric 1 mtu 1500
   flags: <UP,BROADCAST,RUNNING,MULTICAST>
   inet 192.168.5.1/24 broadcast 192.168.5.255
   inet6 fd20:1d78:f920::1/60
   inet6 fe80::12fe:edff:feaf:635e/64
  Interface dummy0 is down
   index 11 metric 1 mtu 1500
   flags: <BROADCAST,NOARP>
   HWaddr: 06:31:12:18:c8:5c
  ```

- **Assign IP address:**

  ```
  OpenWrt# conf term
  OpenWrt(config)# int dummy0
  OpenWrt(config-if)# ip add 10.0.0.1/24
  ```

- **Remove IP address:**

  ```
  OpenWrt(config-if)# no ip add 10.0.0.1/24
  ```
Lab 2: RIP and OSPF

Basic commands

Routing table

- **Show routing table:**

  ```
  OpenWrt# show ip route
  Codes: K - kernel route, C - connected, S - static, R - RIP,
  0 - OSPF, I - IS-IS, B - BGP, H - HSLS, o - OLSR,
  b - BATMAN, A - Babel,
  > - selected route, * - FIB route
  K>* 0.0.0.0/0 via 192.168.1.1, eth0.2
  C>* 10.0.0.0/24 is directly connected, dummy0
  C>* 127.0.0.0/8 is directly connected, lo
  C>* 192.168.1.0/24 is directly connected, eth0.2
  C>* 192.168.5.0/24 is directly connected, br-lan
  ```

- **Add route to network 10.0.0.1/24 via gateway 192.168.1.1:**

  ```
  OpenWrt# conf term
  OpenWrt(config-if)# ip route 10.0.0.1/24 192.168.1.1
  ```
Show current configuration

```
OpenWrt# show running-config
Current configuration:
!
password zebra
!
interface br-lan
  ipv6 nd suppress-ra
!
interface dummy0
  ipv6 nd suppress-ra
!
interface eth0
  ipv6 nd suppress-ra
!
interface eth0.1
  ipv6 nd suppress-ra
!
interface eth0.2
  ipv6 nd suppress-ra
!
interface lo
!
interface wlan0
  ipv6 nd suppress-ra
!
access-list vty permit 127.0.0.0/8
access-list vty deny any
```
Lab 2: RIP and OSPF

Basic commands

Miscelanea

- Avoid expiration of telnet session:
  ```
  OpenWrt# conf term
  OpenWrt(config)# line vty
  OpenWrt(config-line)# exec-timeout 0
  ```

- Save current configuration:
  ```
  OpenWrt# write
  Configuration saved to /etc/quagga/zebra.conf
  ```

- Change hostname (and prompt):
  ```
  OpenWrt# conf term
  OpenWrt(config)# hostname R1
  R1(config)#
  ```
Part III

Lab 2: RIP and OSPF

Outline

- Description
- IOS fundamentals
- Quagga set up
- Basic commands
- RIP review
- RIP configuration
- RIP Lab setup
- OSPF review
- OSPF configuration
- OSPF Lab setup
Lab 2: RIP and OSPF

RIP review

Routing Information Protocol (RIP)

One of the oldest and more simple routing protocols. In summary, it works as follows:

- The **metric** is the number of jumps until the destination: 1 if the destination is a network directly connected, 2 if it has to go through a router, etc.
- The routers send **periodically** (each 30 seconds) a broadcast RIP message in each interface with the known destinations and metrics. Sent with **UDP**, source and destination port: **520**.
- If we stop receiving RIP messages from a neighbour (180 seconds), we assume that it is down.
- The metric’s value of **infinity** is **16**.
- **RIP version 2**: The netmask is added to the destinations sent in the messages. The messages are sent to the multicast address: **224.0.0.9** (all RIPv2 routers).
RIP Convergence Problems

- Depending on the route update message order, convergence problems may arise (Count to Infinity):

- Evolution of $D=\text{N4}$ entry when R3 fails:

```
R1: R2 3 → R2 3
R2: R3 2 → R3 16 → R1 4
```

Llorenç Cerdà-Alabern
Linux Routers and Community Networks
Lab 2: RIP and OSPF

RIP review

Solutions to RIP Convergence Problems

- **Split horizon**: When the router sends the update, removes the entries having a gateway in the interface where the update is sent.

- **Triggered updates**: Consists of sending the update before the 30 seconds timer expires, when a metric changes in the routing table.
Lab 2: RIP and OSPF

**Outline**

- Description
- IOS fundamentals
- Quagga set up
- Basic commands
- RIP review
- RIP configuration
- RIP Lab setup
- OSPF review
- OSPF configuration
- OSPF Lab setup
Lab 2: RIP and OSPF

RIP configuration

Network Command

- Set the interfaces that have to send or process RIP update messages.
- Set which directly connected networks to advertise.
- Quagga implements RIPv2 by default and masks must be provided to network command.
Lab 2: RIP and OSPF

RIP configuration

Route Summarization

- By default, CISCO routers do route summarization. The summarization is done at the class boundary. For example, if in the routing tables we have the subnets 10.0.1.0/24 and 10.0.2.0/24, when sending a RIP message to the net 192.168.0.0/24 it will be sent 10.0.0.0/8.

- In order for the router to be advertise static routes (including the default route): command redistribute static.

- The router uses two metrics: the administrative metric and the routing algorithm metric. If several routes to a same destination exist, the route with the lower administrative metric is chosen. For example, RIP has administrative metric 120 and OSPF 110.

```
R1# sh ip ro
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
       I - ISIS, B - BGP, > - selected route, * - FIB route
R>* 17.16.4.0/24 [120/2] via 172.16.1.2, e0, 00:00:07
...
```

- Verification command: `show ip rip status`. 
RIP configuration example

```bash
root@OpenWrt:~# telnet localhost ripd
Entering character mode
Escape character is '^[].'
Hello, this is Quagga (version 0.99.22.3).
User Access Verification
Password: zebra
OpenWrt> enable
OpenWrt# configure terminal
OpenWrt> enable
OpenWrt(config)# hostname ripd
ripd(config)# router rip
ripd(config-router)# redistribute static
ripd(config-router)# network 172.16.0.0/24
...
ripd(config-router)# ^Z
ripdd# write
Configuration saved to /etc/quagga/ripd.conf
ripd# show ip rip ?
  status  IP routing protocol process parameters and statistics
```
# Lab 2: RIP and OSPF

## Outline

- **Description**
- **IOS fundamentals**
- **Quagga set up**
- **Basic commands**
- **RIP review**
- **RIP configuration**
- **RIP Lab setup**
- **OSPF review**
- **OSPF configuration**
- **OSPF Lab setup**
Objectives:

- We want to set up this network.
- Configured using quagga and RIP.
- Following the guidelines in the next slides.
Lab 2: RIP and OSPF

RIP Lab setup

**Preparation**

1. Install the packages `kmod-dummy`, `quagga`, `quagga-ospfd`, `quagga-ripd` and `quagga-zebra`.

2. Rename the file `/etc/quagga/ospfd.conf` to avoid starting the ospfd daemon:
   ```sh
   ~# mv /etc/quagga/ospfd.conf /etc/quagga/ospfd.conf.dst
   ```

3. Start quagga daemons (check that ospfd does not start):
   ```sh
   root@OpenWrt:/etc/quagga# /etc/init.d/quagga start
   quagga.init: Starting zebra ... done.
   quagga.init: Starting ripd ... done.
   root@OpenWrt:/etc/quagga#
   ```

4. Have a look to the configuration files: `/etc/quagga`.

5. Configure the network of figure using quagga.
Lab 2: RIP and OSPF

RIP Lab setup

RIP Testing

1. Check the routing tables. Does RIP quagga daemon do route-summaziation?

2. Use **traceroute** to figure out the path to different destinations.

3. Check the RIP messages sent by the router using **tcpdump**:
   
   ```
   ~# tcpdump -vni eth0.1 port 520
   ```

4. Disable Split Horizon in one interface, and observe the routes that are advertised by in the update messages.
   
   ```
   ripd# conf term
   ripd(config)# int eth0
   ripd(config-if)# no ip rip split-horizon
   ```

5. Disconnect one network and observe the **trigger updates** and **metric 16**.
Part III

Lab 2: RIP and OSPF

Outline

- Description
- IOS fundamentals
- Quagga set up
- Basic commands
- RIP review

- RIP configuration
- RIP Lab setup
- OSPF review
- OSPF configuration
- OSPF Lab setup
## Open Shortest Path First (OSPF)

- Standardized inside the **IETF**, aim of having a high performance protocol.
- **Link state** protocol: send information on neighbors networks and routers.
- Link State Advertisements, **LSA**: Send information to all other routers using **flooding**.
- Each router maintains **network topology database**.
- Algorithm **Shortest Path First (SPF)** to calculate optimal routes.
- The **metric** is dimensionless (does not represent the number of hops). The infinite metric is 0xFFFF.
- A **hello protocol** to discover neighbors.
- OSPF does not carry data via UDP or TCP. Instead, OSPF encapsulates messages into IP datagrams directly using protocol number 89.
Lab 2: RIP and OSPF

OSPF review

**OSPF areas**

- Designated Router (DR) and a Backup Designated Router (BDR). The DR is the only router in the broadcast domain that sends LSA.
- Router ID (RID): IP address of the router greatest value or dummy interface.
- **Priority** for the election of the DR and BDR: highest RID.
- **Area**: increases scalability. All networks inside an area can be aggregated in a single prefix.
- There must be a **backbone area 0**, to which all other areas are connected. Area 0 cannot be discontiguous.
- Routers can be Internal Routers (IR), if they have all the interfaces in the same area or Area Border Router (ABR) if they have interfaces in more than one area.
Lab 2: RIP and OSPF

Part III

Lab 2: RIP and OSPF

Outline

- Description
- IOS fundamentals
- Quagga set up
- Basic commands
- RIP review
- RIP configuration
- RIP Lab setup
- OSPF review
- OSPF configuration
- OSPF Lab setup
### Commands

- First you should configure an IP **dummy interface** in order to fix the RID.

- **network** command works similarly to RIP, but specifying the area.

- Area **route aggregation** is achieved using the **range** command in ABR routers.

- **Default route** is distributed using the command `default-information originate`.

- **Verification** commands: `show ip ospf ?`.
Lab 2: RIP and OSPF

OSPF configuration example

```
~# telnet localhost ospfd
Entering character mode
Escape character is '^[].
Hello, this is Quagga (version 0.99.22.3).
User Access Verification
Password: zebra
OpenWrt> enable
OpenWrt# configure terminal
OpenWrt(config)# hostname ospfd
ospfd(config)# router ospf
ospfd(config-router)# network 10.0.1.0/24 area 0
ospfd(config-router)# network ...
ospfd(config-router)# area 1 range 172.16.0.0/16
ospfd(config-router)# default-information originate
ospfd(config-router)# ^Z
ospfdd# write
Configuration saved to /etc/quagga/ospfd.conf
ospfd# show ip ospf ?
    border-routers for this area
database Database summary
interface Interface information
neighbor Neighbor list
route OSPF routing table
```
Part III

Lab 2: RIP and OSPF

Outline

- Description
- IOS fundamentals
- Quagga set up
- Basic commands
- RIP review
- RIP configuration
- RIP Lab setup
- OSPF review
- OSPF configuration
- OSPF Lab setup
Objectives:

- We want to set up this network.
- Configured using quagga and OSPF.
- Following the guidelines in the next slides.
Network Configuration

1. Reboot the routers to clean the configuration of RIP Lab.

2. Rename the desired daemons to start:

   ~# mv /etc/quagga/ripd.conf /etc/quagga/ripd.conf.dst
   ~# mv /etc/quagga/ospfd.conf.dst /etc/quagga/ospfd.conf

3. Start quagga daemons (check that ripd does not start):

   root@OpenWrt:/etc/quagga# /etc/init.d/quagga start
   quagga.init: Starting zebra ... done.
   quagga.init: Starting ospfd ... done.
   root@OpenWrt:/etc/quagga#

4. Have a look to the configuration files: /etc/quagga.

5. Assign IP addresses to interfaces using zebra daemon.

6. Configure OSPF using ospfd daemon.
Lab 2: RIP and OSPF

OSPF Lab setup

OSPF Testing

1. Check the **routing tables**.
2. Check the **routing metrics**.
3. Use **traceroute** to figure out the path to different destinations.
4. Activate area **range aggregation** and check routing table entries.
5. Capture OSPF messages sent by the router using **tcpdump**:
   ```
   ~# tcpdump -vni eth0.1 proto 89
   ```
6. **Disconnect one network** and observe the LSA messages captured with tcpdump, and the changes in the routing tables.