experiences with zebra/quagga configurations and command line interface
copyright notice

- All the pages/slides in this presentation, including but not limited to, images, photos, animations, videos, sounds, music, and text (hereby referred to as “material”) are protected by copyright.

- This material, with the exception of some multimedia elements licensed by other organizations, is property of the authors and/or organizations appearing in the first slide.

- This material, or its parts, can be reproduced and used for didactical purposes within universities and schools, provided that this happens for non-profit purposes.

- Information contained in this material cannot be used within network design projects or other products of any kind.

- Any other use is prohibited, unless explicitly authorized by the authors on the basis of an explicit agreement.

- The authors assume no responsibility about this material and provide this material “as is”, with no implicit or explicit warranty about the correctness and completeness of its contents, which may be subject to changes.

- This copyright notice must always be redistributed together with the material, or its portions.
hosts need routing

- each host with a network stack performs some elementary routing
- at the very least, the network stack may be used to access local services (e.g., Xorg)
- the host must decide when a packet needs to be sent to the network interface card (nic) and when it needs to be bounced to the loopback interface (lo)
the IP layer uses a routing table to decide which interface the packet needs to be forwarded to.

Routing Table:

<table>
<thead>
<tr>
<th>network</th>
<th>nmask</th>
<th>nexthop</th>
<th>int</th>
</tr>
</thead>
<tbody>
<tr>
<td>200.3.24.0</td>
<td>255.255.255.0</td>
<td>12.0.0.4</td>
<td>eth1</td>
</tr>
<tr>
<td>193.2.0.0</td>
<td>255.255.248.0</td>
<td>11.0.0.2</td>
<td>eth0</td>
</tr>
<tr>
<td>100.4.5.0</td>
<td>255.240.0.0</td>
<td>11.0.0.3</td>
<td>eth0</td>
</tr>
<tr>
<td>0.0.0.0</td>
<td>0.0.0.0</td>
<td>11.0.0.2</td>
<td>eth0</td>
</tr>
</tbody>
</table>
routers

- a router (also called gateway or intermediate-system)
  - has more than one network interface card
  - feeds incoming IP packets (that are not for the router itself) back in the routing process
    - this operation is called relaying or forwarding

is it for me?

yes no
routing protocols

- routing protocols are used to automatically update routing tables, relieving administrators from the need to do it manually.
- Routers (i.e., devices that run routing protocols) in netkit are virtual machines that run a specific piece of software that implements routing protocols.

zebra/quagga
about zebra/quagga

Quagga Routing Software Suite,
GPL licensed IPv4/IPv6 routing software

- a software that implements several routing protocols
  - rip (v1 and v2)
  - ospf (v2 and v3)
  - is-is
  - bgp
- quagga: “a fork of GNU Zebra [that] aims to build a more involved community around Quagga than the current centralised model of GNU Zebra”
- zebra development stopped at release 0.95a
- quagga superseded zebra
  - in most cases, in netkit you can equivalently refer to “quagga” or “zebra”
zebra: a routing daemon

file access
connections
routing updates

bgpd

ripd

ospfd

zebra: a routing daemon

kernel

netkit – [ lab: quagga ]
inspecting zebra configuration files

when zebra is started, each daemon checks these files to read the starting configuration
sample daemons file

```
# virtual machine

pc1:/etc/zebra# less daemons
# This file tells the zebra package
# which daemons to start.
# Entries are in the format: <daemon>=(yes|no|priority)
# where 'yes' is equivalent to infinitely low priority, and
# lower numbers mean higher priority. Read
# /usr/doc/zebra/README.Debian for details.
# Daemons are: bgpd zebra ospfd ospf6d ripd ripngd
zebra=yes
bgpd=no
ospfd=no
ospf6d=no
ripd=yes
ripngd=no
```

- the zebra main daemon will be started
- the rip daemon will be started too
sample zebra configuration file
(zebra.conf)

pc1:/etc/zebra# less zebra.conf
!
!*- zebra -*-
!
! zebra sample configuration file
!
! $Id: zebra.conf.sample,v 1.14 1999/02/19 17:26:38 developer
Exp $
!
hostname Router
password zebra
enable password zebra
!
! interface lo
zebra.conf

the prompt of the zebra interface
the password to connect to the daemon
administrative password
sample ripd configuration file

(ripd.conf)

```
pc1:/etc/zebra# cat ripd.conf
!
hostname ripd
password root
enable password root
!
router rip
redistribute connected
network 100.1.0.0/16
!
log file /var/log/zebra/ripd.log
pc1:/etc/zebra#
```

**talk rip on some interface**

**redistribute to rip neighbors information about all directly connected subnets**

**send rip multicast packets to interfaces falling into this prefix**
a simple topology

Legend

1. subnet
2. last byte of ip address
3. assigned prefix
4. network interface

A

100.0.0.0/24

B

r1

1
eth0

100.0.0.0/24

C

r2

2
eth0

150.0.0.0/30

D

r3

2
eth0

200.0.0.0/30

netkit – [ lab: quagga ]

last update: Nov 2011
launching the lab script

- the lab configuration is such that
  - three virtual hosts are created and connected to the right collision domains (virtual hubs)
  - for each virtual host
    - network interfaces are automatically configured
    - quagga configuration files are updated
  - the zebra routing daemon is automatically started

```
user@localhost:~$ cd netkit-lab_quagga
user@localhost:~/netkit-lab_quagga$ lstart
```
connecting to the main zebra daemon

```
r3:~# telnet localhost zebra
Trying 127.0.0.1...
Connected to r3.
Escape character is '^[].

Hello, this is Quagga (version 0.99.10).
Copyright 1996-2005 Kunihiro Ishiguro.

User Access Verification

Password: zebra
zebra>
```

we are unprivileged users
privileges on a router

- **privileged user**: performs persistent configurations affecting the router behavior
  - sees critical information
  - performs elementary configurations that do not affect router behavior (e.g., set time and date)
  - commits changes

- **unprivileged user**: sees non-critical information

- **configurator**
  - configure terminal
  - enable
  - disable
  - quit
available commands

- press ‘?’ at the command prompt...

zebra>
  echo      Echo a message back to the vty
  enable    Turn on privileged mode command
  exit      Exit current mode and down to previous mode
  help      Description of the interactive help system
  list      Print command list
  quit      Exit current mode and down to previous mode
  show      Show running system information
  terminal  Set terminal line parameters
  who       Display who is on vty
zebra>  

- ...Or...

© Computer Networks Research Group Roma Tre

netkit – [ lab: quagga ]
available commands

- type ‘list’ (an excerpt of the output follows)

```
zebra> list
  enable
  exit
  help
  list
  quit
  show interface [IFNAME]
  show ip forwarding
  show ip route
  show ipv6 forwarding
  show ipv6 route
  show memory
  show version
  terminal length <0-512>
  terminal no length
  who
zebra> 
```

inspecting interfaces

```
zebra> show interface eth0
Interface eth0 is up, line protocol detection is disabled
   index 3 metric 1 mtu 1500
   flags: <UP,BROADCAST,RUNNING,MULTICAST>
   HWaddr: ee:97:f2:ab:47:0c
   inet 200.0.0.2/30 broadcast 200.0.0.3
   inet6 fe80::ec97:f2ff:feab:470c/64
     15 input packets (0 multicast), 948 bytes, 0 dropped
     0 input errors, 0 length, 0 overrun, 0 CRC, 0 frame
     0 fifo, 0 missed
     9 output packets, 642 bytes, 0 dropped
     0 output errors, 0 aborted, 0 carrier, 0 fifo, 0 heartbeat
     0 window, 0 collisions
zebra>  
```

- this roughly corresponds to using `ifconfig` at the shell prompt
inspecting the zebra routing table

```
zebra> show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP, O - OSPF,
I - ISIS, B - BGP, > - selected route, * - FIB route

R>* 100.0.0.0/24 [120/3] via 200.0.0.1, eth0, 00:06:28
C>* 127.0.0.0/8 is directly connected, lo
R>* 150.0.0.0/30 [120/2] via 200.0.0.1, eth0, 00:06:28
C>* 200.0.0.0/30 is directly connected, eth0
C>* 220.0.0.0/24 is directly connected, eth1
```

- FIB entries from this table (marked with a ‘>’) are injected into the kernel routing table
altering zebra configuration

zebra> enable
Password: zebra
zebra# configure terminal
zebra(config)# hostname zebra-r3
zebra-r3(config)# password foo
zebra-r3(config)# enable password foo
zebra-r3(config)# quit
zebra-r3# write file
Configuration saved to /etc/zebra/zebra.conf
zebra-r3# disable
zebra-r3> exit
Connection closed by foreign host.
r3:~#
inspecting the rip routing table

```
$ r3:~# telnet localhost ripd
......
Password: zebra
ripd> show ip rip
Codes: R - RIP, C - connected, S - Static, O - OSPF, B - BGP
Sub-codes:
    (n) - normal, (s) - static, (d) - default, (r) - redistribute, (i) - interface

<table>
<thead>
<tr>
<th>Network</th>
<th>Next Hop</th>
<th>Metric</th>
<th>From</th>
<th>Tag</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>R(n) 100.0.0.0/24</td>
<td>200.0.0.1</td>
<td>3</td>
<td>200.0.0.1</td>
<td>0</td>
<td>02:43</td>
</tr>
<tr>
<td>R(n) 150.0.0.0/30</td>
<td>200.0.0.1</td>
<td>2</td>
<td>200.0.0.1</td>
<td>0</td>
<td>02:43</td>
</tr>
<tr>
<td>C(i) 200.0.0.0/30</td>
<td>0.0.0.0</td>
<td>1</td>
<td>self</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>C(i) 220.0.0.0/24</td>
<td>0.0.0.0</td>
<td>1</td>
<td>self</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
```
a one-fits-all shell

- instead of having to connect to each single daemon, users can interact with quagga by using a built-in shell, called **vtysh**

  ```
  r1:~# vtysh
  Hello, this is Quagga (version 0.99.10).
  r1# 
  ```

- the user is not prompted for a password
- all the commands from the single routing daemons (including quagga itself) are available in this shell