Juniper Routing

1

- Routing konfigurálás áttekintése
  - Statikus routing beállítása
  - OSPF konfigurálás
  - BGP konfigurálás
  - MPLS konfigurálás
- Access listák használata
- Specifikus alapkonfiguráció áttekintése

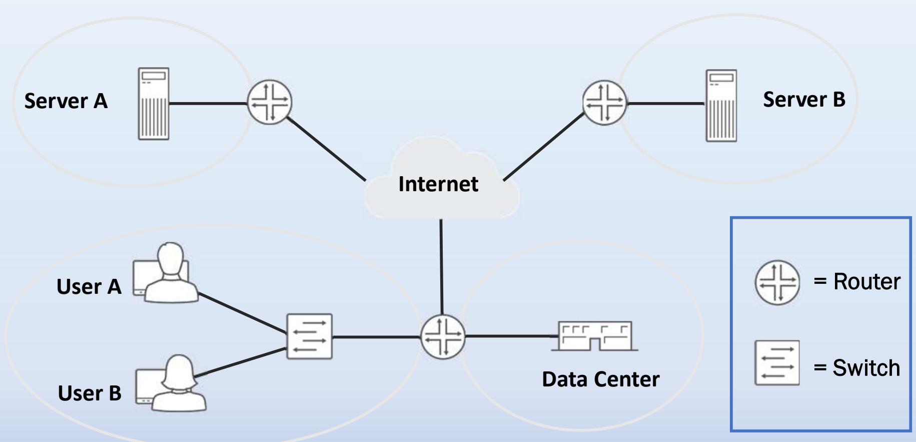
2

Routing alapok

3

What Is Routing?

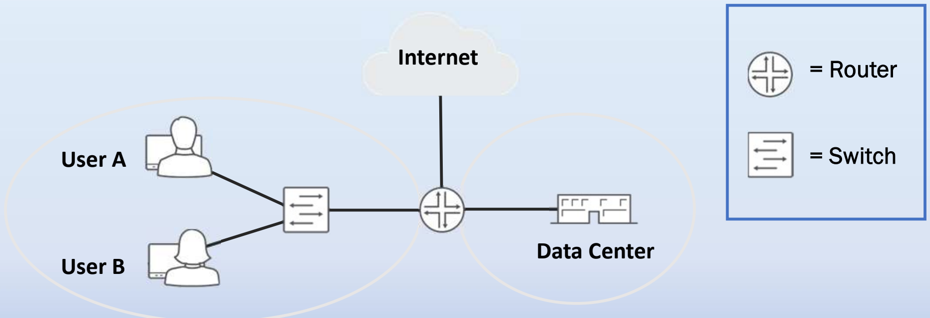
- The process of moving data between Layer 3 networks



4

Components of Routing

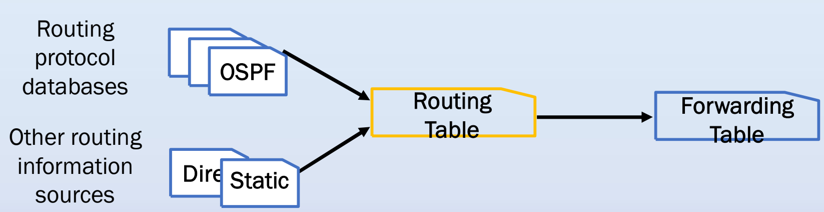
- For a device to communicate with another device in a remote network, the following requirements exist:
  - End-to-end communications path
  - Routing information on participating Layer 3 devices



5

The Routing Table

- Compiles information learned from routing protocols and other routing information sources
- Selects an active route to each destination
- Populates the forwarding table
- Main unicast routing tables are `inet.0`, for IPv4 routing, and `inet6.0`, for IPv6 routing



6

Route Preference

- Ranks routes received from different sources
- Primary criterion for selecting the active route
  - Used as a tiebreaker when the same destination prefix is available through multiple sources

Route Preference Values

Routing Information Source	Default Preference
Direct	0
Local	0
Static	5
OSPF internal	10
RIP	100
OSPF AS external	150
BGP (both EBGP and IBGP)	170

More Preferred

Less Preferred

7

Viewing the Routing Table

- Use **`show route`** to display route table contents:

```
user@router> show route

inet.0: 6 destinations, 7 routes (6 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.1.1.0/24        *[Static/5] 00:10:24
                    > to 172.29.30.253 via ge-0/0/10.0
                    [OSPF/10] 00:03:38, metric 2
                    > to 172.18.25.2 via ge-0/0/13.0
172.18.25.0/30     *[Direct/0] 00:11:05
                    > via ge-0/0/13.0
172.18.25.1/32     *[Local/0] 00:11:05
                      Local via ge-0/0/13.0
172.29.30.0/24     *[Direct/0] 00:11:05
                    > via ge-0/0/10.0
172.29.30.1/32     *[Local/0] 00:11:05
                      Local via ge-0/0/10.0
…
```

Route source and preference

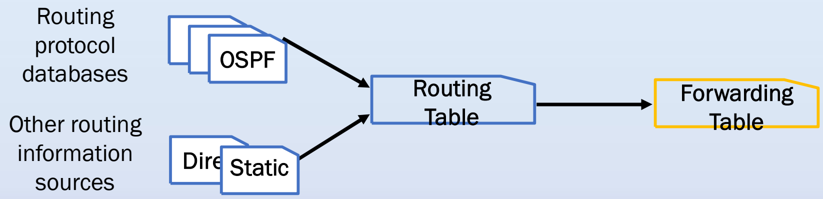
Asterisk (*) indicates that the route is selected as active

Route table name

8

The Forwarding Table

- Stores required information for packet forwarding operation; contents include the destination prefixes and the associated outgoing interfaces
  - Use **`show route forwarding-table`** to view contents



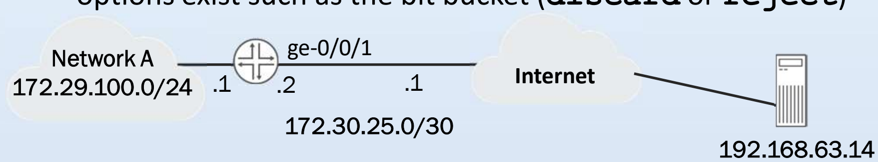
9

- **Routing konfigurálás áttekintése**
  - **Statikus routing beállítása**
  - **OSPF konfigurálás**
  - **BGP konfigurálás**
  - **MPLS konfigurálás**
- **Access listák használata**
- **Specifikus alapkonfiguráció áttekintése**

10

Static Routes

- Manually configured routes added to the route table
  - Defined under `[edit routing-options]` hierarchy
- Require a valid next hop
  - Typically the IP address of a directly connected device; other options exist such as the bit bucket (**`discard`** or **`reject`**)




```
user@router> show route 192.168.63.14

inet.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[Static/5] 01:09:34
                    > to 172.30.25.1 via ge-0/0/1.0
```

Default static route

11

Configuring Static Routing

- Static route configuration example:

```
[edit routing-options]
user@router# show
rib inet6.0 {
    static {
        route 0::/0 next-hop 3001::1;
    }
}
static {
    route 0.0.0.0/0 next-hop 172.30.25.1;
    route 172.28.102.0/24 {
        next-hop 10.210.11.190;
        no-readvertise;
    }
}
```

IPv6 default static route

IPv4 default static route

Restricts route from being advertised into a routing protocol through routing policy; Strongly recommended for static routes used for management traffic

12

Configure Static Routing

- To configure the default route:
  # set routing-options static route 0.0.0.0/0 next-hop 10.0.1.1
- To configure a static route:
  # set routing-options static route 192.168.100.0/24 next-hop 10.0.2.2
- To prevent static route from being readvertised:
  # set routing-options static route 192.168.100.0/24 no-readvertise
- To remove inactive routes from the forwarding table:
  # set routing-options static route 192.168.100.0/24 active

13

Monitoring Static Routing

- Monitoring:
  - Use `show route protocol static` to display static routes:

```
user@router> show route protocol static

inet.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

0.0.0.0/0          *[Static/5] 00:41:59
                    > to 172.30.25.1 via ge-0/0/1.0
```

Route source and preference

Default static route

Next-hop interface and IP address

…
  - Use the ping utility to verify end-to

```
user@router> ping 192.168.63.14 rapid count 25
PING 192.168.63.14 (192.168.63.14): 56 data bytes
!!!!!!!!!!!!!!!!!!!!!!!!!
--- 192.168.63.14 ping statistics ---
25 packets transmitted, 25 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.027/0.057/0.145/0.032 ms
```

Test confirms reachability

14

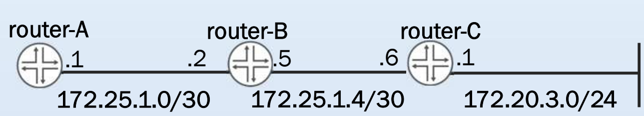
Monitoring Static Routing – additional commands

- **> show route** – display the contents of the primary route table(s) (inet.0; inet6.0)
- **> show route all** – display all route tables
- **> show route forwarding-table** – display entries in all forwarding tables

15

Next-Hop Resolution

- Resolving indirect next hops:
  - By default, the software can resolve only directly connected next hops
  - Use the **`resolve`** option to allow resolution of indirectly connected next hops:




```
[edit routing-options]
user@router-A# show
static {
    route 172.20.3.0/24 {
        next-hop 172.25.1.6;
        resolve;
    }
}
```

Indirect next hop

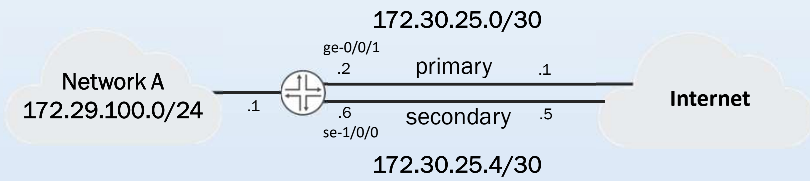
`resolve` option required

16

CLICO

Qualified Next Hops

- Use **`qualified-next-hop`** to allow independent preference for static routes to the same destination:




```
[edit routing-options]
user@router# show
static {
    route 0.0.0.0/0 {
        next-hop 172.30.25.1;
        qualified-next-hop 172.30.25.5 {
            preference 7;
        }
    }
}
```

Primary next hop due to default route preference (5)

Secondary next hop due to configured route preference (7)

17

CLICO

Configuration Example

- Sample IPv4 and IPv6 static route configuration:

```
[edit routing-options]
user@R1# show
rib inet6.0 {
    static {
        route 0::/0 {
            next-hop 3001::1;
            preference 250;
        }
    }
}
static {
    defaults {
        preference 250;
    }
    route 0.0.0.0/0 {
        next-hop 172.30.25.1;
        qualified-next-hop 172.30.25.5 {
            preference 251;
        }
    }
    route 172.28.102.0/24 {
        next-hop 10.210.11.190;
        no-readvertise;
    }
}
```

IPv6 default static route

Parameters defined under defaults section are applied to IPv4 static routes that do not include explicit definitions

IPv4 default static route with secondary qualified next hop

Restricts route from being advertised through routing policy; highly suggested for static routes used for management traffic

18

- **Routing konfigurálás áttekintése**
  - Statikus routing beállítása
  - **OSPF konfigurálás**
  - BGP konfigurálás
  - MPLS konfigurálás
- Access listák használata
- Specifikus alapkonfiguráció áttekintése

19

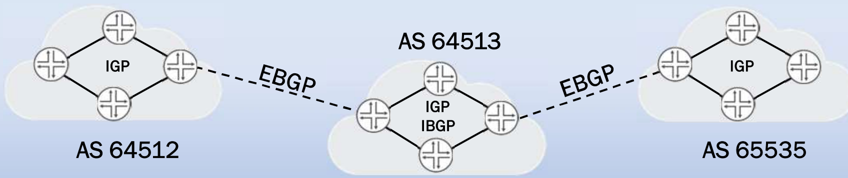
Dynamic Routing

- Method of dynamically learning routing information
- Dynamic routing has the following benefits:
  - Lower administrative overhead
  - Increased network availability
  - Greater network scalability

20

Dynamic Routing Protocols

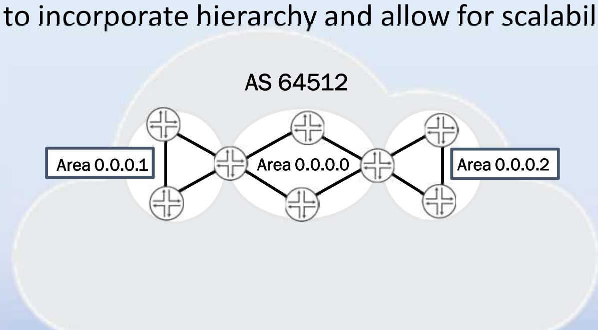
- A summary of dynamic routing protocols:
  - IGPs operate within a single autonomous system
    - Single network administration that provides for unique routing policy and flexible use of network resources
    - Examples include RIP, IS-IS, and OSPF
  - EGPs operate among different autonomous systems
    - Independent administrative entities that communicate between independent network infrastructures
    - Current EGP in use today is BGP



21

OSPF Protocol Overview

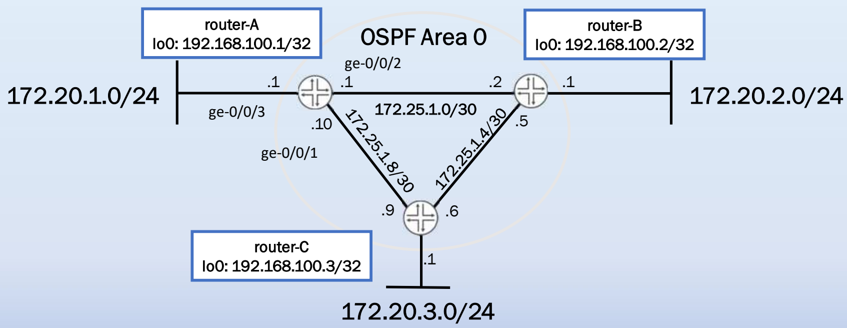
- A link-state IGP:
  - Reliably floods link-state information to neighbors
  - Creates a complete database of network
  - Calculates best path to each destination
  - Uses areas to incorporate hierarchy and allow for scalability



22

Case Study: Objective and Topology

- Use a single OSPF area to provide connectivity among all connected subnets and loopback addresses; ensure that no adjacencies are formed on interfaces connecting to the 172.20.x.0/24 subnets



23

Case Study: Configuring OSPF

- Sample OSPF configuration taken from Host-A:

```
[edit protocols ospf]
user@router-A# set area 0 interface ge-0/0/1.0

[edit protocols ospf]
user@router-A# set area 0 interface ge-0/0/2.0

[edit protocols ospf]
user@router-A# set area 0 interface ge-0/0/3.0 passive

[edit protocols ospf]
user@router-A# set area 0 interface lo0.0

[edit protocols ospf]
user@router-A# show
area 0.0.0.0 {
    interface ge-0/0/1.0;
    interface ge-0/0/2.0;
    interface ge-0/0/3.0 {
        passive;
    }
    interface lo0.0;
}
```

Specify the logical interface. If a unit is not referenced, the Junos OS assumes unit 0.

Use the **`passive`** option to prohibit adjacency formation.

The Junos OS converts `area 0` to its proper dotted decimal notation (0.0.0.0).

24

CLICO

The Router ID

- OSPF uses the RID to identify the router from which a packet originated
  - You can manually define the RID under the [**edit routing-options**] hierarchy
  - If you come online is used (
  - If lo0 : hardware interface is used

```
[edit routing-options]
user@R1# show
router-id 192.168.100.1;
```

The RID is a 32-bit number in dotted quad notation.

Note: we strongly recommend that you configure a RID to avoid unpredictable behaviour if the interface addresses are changed

25

CLICO

Case Study: Verifying OSPF Neighbor State

- Use **`show ospf neighbor`** to display adjacencies
  - Use **`detail`** or **`extensive`** options for added information

```
user@router-A> show ospf neighbor
Address          Interface              State     ID               Pri  Dead
172.25.1.9       ge-0/0/1.0             Full      192.168.100.3    128    38
172.25.1.2       ge-0/0/2.0             Full      192.168.100.2    128    35
```

The state of the adjacencies shows `Full`, which means neighbors can exchange routing information

26

Case Study: Viewing OSPF Routes

- Use **`show route protocol ospf`** to display OSPF routes

```
user@router-A> show route protocol ospf

inet.0: 15 destinations, 15 routes (15 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.20.2.0/24      *[OSPF/10] 00:03:55, metric 2
                    > to 172.25.1.2 via ge-0/0/2.0
172.20.3.0/24      *[OSPF/10] 00:00:04, metric 2
                    > to 172.25.1.9 via ge-0/0/1.0
172.25.1.4/30      *[OSPF/10] 00:03:46, metric 2
                    > to 172.25.1.9 via ge-0/0/1.0
                      to 172.25.1.2 via ge-0/0/2.0
192.168.100.2/32   *[OSPF/10] 00:03:55, metric 1
                    > to 172.25.1.2 via ge-0/0/2.0
192.168.100.3/32   *[OSPF/10] 00:03:46, metric 1
                    > to 172.25.1.9 via ge-0/0/1.0
224.0.0.5/32       *[OSPF/10] 00:16:13, metric 1
                      MultiRecv…
```

27

Other Key OSPF Monitoring Commands

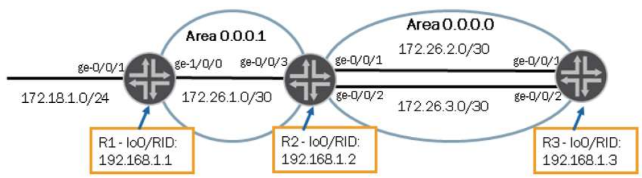
- Additional **show** commands exist to provide detailed information on the operation of OSPF:
  - show ospf interface
  - show ospf route
  - show ospf database
  - show ospf statistics

28

Displaying OSPF Interface Parameters

- Use the **show ospf interface** command to display OSPF interface parameters

```
user@R2> show ospf interface
Interface           State   Area            DR ID           BDR ID          Nbrs
ge-0/0/1.0          BDR     0.0.0.0         192.168.1.3     192.168.1.2        1
ge-0/0/2.0          DR      0.0.0.0         192.168.1.2     192.168.1.3        1
lo0.0               DR      0.0.0.0         192.168.1.2     0.0.0.0            0
ge-0/0/3.0          DR      0.0.0.1         192.168.1.2     192.168.1.1        1
```



29

Displaying OSPF Route Information

- Use the **show ospf route** command to display routes learned from, and advertised to, OSPF

```
user@R2> show ospf route
Topology default Route Table:

Prefix              Path   Route      NH    Metric  NextHop       Nexthop
                    Type   Type       Type          Interface     addr/label
192.168.1.1         Intra  AS BR      IP         1  ge-0/0/3.0    172.26.1.1
192.168.1.3         Intra  Area BR    IP         1  ge-0/0/1.0    172.26.2.2
172.18.1.0/24       Ext2   Network    IP         0  ge-0/0/3.0    172.26.1.1
172.26.1.0/30       Intra  Network    IP         1  ge-0/0/3.0
172.26.2.0/30       Intra  Network    IP         1  ge-0/0/1.0
172.26.3.0/30       Intra  Network    IP       100  ge-0/0/2.0
172.26.4.0/30       Inter  Network    IP         2  ge-0/0/1.0    172.26.2.2
192.168.1.1/32      Intra  Network    IP         1  ge-0/0/3.0    172.26.1.1
192.168.1.2/32      Intra  Network    IP         0  lo0.0
192.168.1.3/32      Intra  Network    IP         1  ge-0/0/1.0    172.26.2.2
192.168.1.4/32      Inter  Network    IP         2  ge-0/0/1.0    172.26.2.2
```

External prefix injected by R1

Metric for ge-0/0/2.0 interface was modified in earlier configuration example.

30

Displaying the OSPF Link-State Database

- Use the **show ospf database** commands to view the OSPF link-state database

ABRs maintain a separate database for each OSPF area to which they are attached.

```
user@R2> show ospf database

    OSPF database, Area 0.0.0.0
 Type       ID               Adv Rtr           Seq      Age  Opt  Cksum  Len
Router   *192.168.1.2      192.168.1.2      0x8000000c  1387  0x22 0x84ae  60
Router    192.168.1.3      192.168.1.3      0x80000023  1249  0x22 0x545e  60
Network   172.26.2.2       192.168.1.3      0x80000005  2049  0x22 0x43e3  32
Network   172.26.3.2       192.168.1.3      0x80000005  2449  0x22 0x38ed  32
Summary  *172.26.1.0       192.168.1.2      0x80000007  2541  0x22 0x4db7  28
Summary   172.26.4.0       192.168.1.3      0x80000025  2249  0x22 0xe9f8  28
Summary  *192.168.1.1      192.168.1.2      0x80000006  1618  0x22 0xa3bb  28
Summary   192.168.1.4      192.168.1.3      0x8000001a  1649  0x22 0x57ef  28
ASBRSum  *192.168.1.1      192.168.1.2      0x80000007  2310  0x22 0x93c9  28

    OSPF database, Area 0.0.0.1
 Type       ID               Adv Rtr           Seq      Age  Opt  Cksum  Len
Router    192.168.1.1      192.168.1.1      0x80000007    56  0x22 0x82c3  48
…
    OSPF AS SCOPE link state database
 Type       ID               Adv Rtr           Seq      Age  Opt  Cksum  Len
Extern    172.18.1.0       192.168.1.1      0x80000005    96  0x22 0x374c  36
```

31

Displaying OSPF Statistics

- Use the **show ospf statistics** command to view OSPF statistics

```
user@R2> show ospf statistics

Packet type             Total                  Last 5 seconds
                   Sent       Received        Sent       Received
   Hello             52             17           0              0
     DbD              9              7           0              0
   LSReq              2              2           0              0
LSUpdate             46             45           0              0
   LSAck             37             33           0              0

DBDs retransmitted     :                   0, last 5 seconds :          0
LSAs flooded           :                  40, last 5 seconds :          0
LSAs flooded high-prio :                  10, last 5 seconds :          0
LSAs retransmitted     :                   0, last 5 seconds :          0
LSAs transmitted to nbr:                   8, last 5 seconds :          0
LSAs requested         :                   2, last 5 seconds :          0
LSAs acknowledged      :                  39, last 5 seconds :          0
…
```

32

OSPF for IPv6

- OSPF for IPv6
  - Defined in RFC5340—July 2008
  - Fundamental mechanics of OSPF unchanged
    - Areas
    - LSA flooding
    - DR elections
    - Stub, NSSA
    - Options
  - There are some significant changes to account for the difference between IPv4 and IPv6 addressing
  - Configuration mainly requires substituting `ospf3` for `ospf`

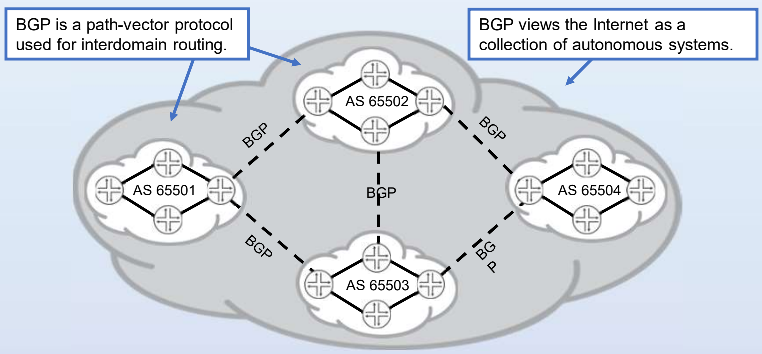
33

- **Routing konfigurálás áttekintése**
  - Statikus routing beállítása
  - OSPF konfigurálás
  - **BGP konfigurálás**
  - MPLS konfigurálás
- Access listák használata
- Specifikus alapkonfiguráció áttekintése

34

What Is BGP?

- BGP is the core routing protocol within the Internet

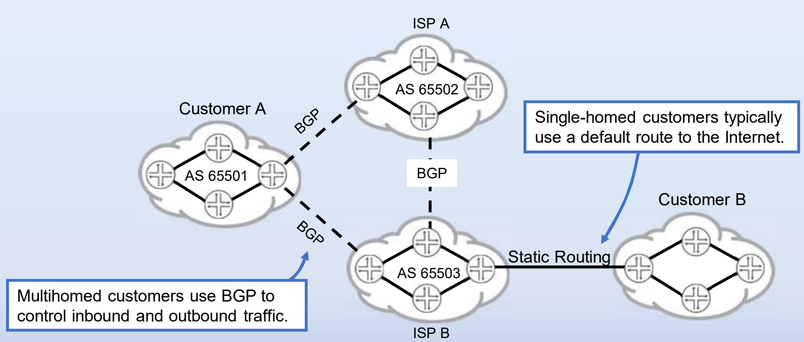


Note: BGP Is an IETF standard defined in RFC 4271 (supersedes RFC 1771).

35

When Should I Use BGP?

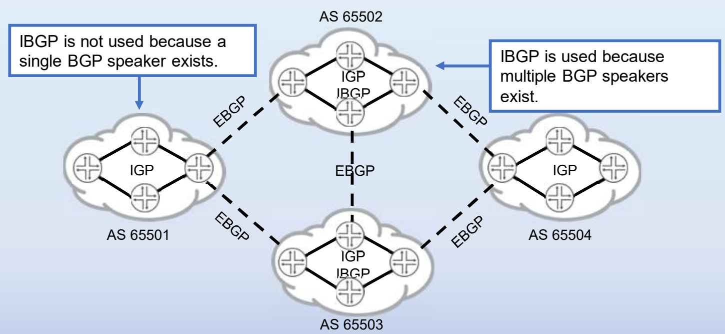
- BGP is typically used in large enterprise environments where multiple ISP connections exist, and in all service provider environments



36

BGP Peers

- BGP peers can reside in different ASs or the same AS
  - Peers in different ASs use the external session type (EBGP)
  - Peers in the same AS use the internal session type (IBGP)

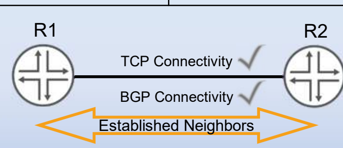


37

BGP Peers

- BGP peering sessions are manually defined and rely on TCP connections
  - No automatic neighbor discovery

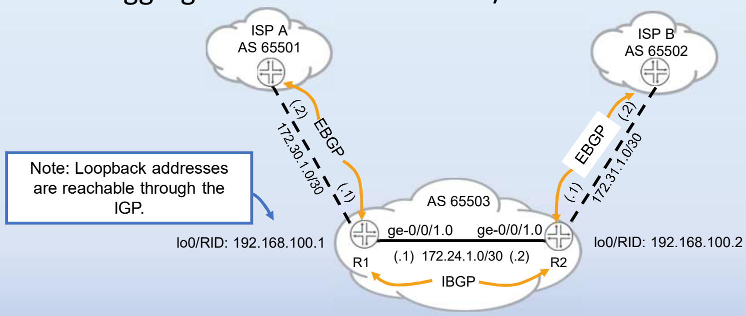
BGP Neighbor States	
TCP Connectivity	**BGP Connectivity**
Idle	OpenSent
Connect	OpenConfirm
Active	Established



38

Case Study: Objectives and Topology

- Configure IBGP and EBGP as outlined on the diagram
  - Use loopback-based IBGP peering sessions
  - Ensure next-hop reachability for both EBGP peers
  - Advertise an aggregate route of 172.24.0.0/22 to both ISPs



39

Case Study: Configuring BGP

```
[edit]
user@R1# show routing-options
router-id 192.168.100.1;
autonomous-system 65503;

[edit]
user@R1# show protocols bgp
group int-65503 {
    type internal;
    local-address 192.168.100.1;
    neighbor 192.168.100.2;
}
group ext-65501 {
    type external;
    peer-AS 65501;
    neighbor 172.30.1.2;
}
```

Device's assigned AS number

BGP group names are user-defined

BGP session type determines if the peering session is IBGP or EBGP

EBGP peer's assigned AS number

40

Case Study: Changing the Next Hop

```
[edit]
user@R1# show policy-options
policy-statement next-hop-self-policy {
    term alter-next-hop {
        then {
            next-hop self;
        }
    }
}

[edit]
user@R1# show protocols bgp
group int-65503 {
    type internal;
    local-address 192.168.100.1;
    export next-hop-self-policy;
    neighbor 192.168.100.2;
}
group ext-65501 {
    type external;
    peer-AS 65501;
    neighbor 172.30.1.2;
}
```

Using the `self` option alters the next-hop value to the address used for the respective peering session

Policy is applied as an `export` policy for the internal BGP group

41

Case Study: Monitoring BGP (1 of 3)

- Use the **`show bgp summary`** command to show an overview of the system's BGP information:

```
user@R1> show bgp summary
Groups: 2 Peers: 2 Down peers: 0
Table     Tot Paths  Act Paths Suppressed  History Damp State    Pending
inet.0           12          6          0        0          0          0
Peer               AS   InPkt OutPkt  OutQ Flaps Last Up/Dwn
State|#Act/Rec/Acc/Damped.
172.30.1.2      65501     914     915    0     0      6:51:16 6/6/6/0          0/0/0/0
192.168.100.2  65503     978     983    0     0      7:19:03 0/6/6/0          0/0/0/0
```

42

Case Study: Monitoring BGP (2 of 3)

- Use the show bgp neighbor command to show the BGP neighbor database:

```
user@R1> show bgp neighbor
Peer: 172.30.1.2+62790 AS 65501 Local: 172.30.1.1+179 AS 65503
  Type: External    State: Established    Flags: <Sync>
  Last State: OpenConfirm   Last Event: RecvKeepAlive
  Last Error: None
  Export: [ adv-aggregate ]
  Options: <Preference PeerAS Refresh>
  Holdtime: 90 Preference: 170
  Number of flaps: 0
  Peer ID: 172.18.1.1       Local ID: 192.168.100.1    Active Holdtime: 90
  Keepalive Interval: 30         Peer index: 0
  BFD: disabled, down
  Local Interface: ge-0/0/3.0
…
```

43

Case Study: Monitoring BGP (3 of 3)

- Use the show bgp group command to show the BGP group database:

```
user@R1> show bgp group
Group Type: Internal    AS: 65503                    Local AS: 65503
  Name: int-65503       Index: 0                     Flags: <Export Eval>
  Export: [ next-hop-self-policy ]
  Holdtime: 0
  Total peers: 1        Established: 1
  192.168.100.2+51067
  inet.0: 0/6/6/0

Group Type: External                                 Local AS: 65503
  Name: ext-65501       Index: 1                     Flags: <Export Eval>
  Export: [ adv-aggregate ]
  Holdtime: 0
  Total peers: 1        Established: 1
  172.30.1.2+62790
  inet.0: 6/6/6/0
…
```

44

Case Study: Displaying BGP Routes (1 of 3)

- Use show route protocol bgp to display BGP routes installed in the RIB-Local:

```
user@R1> show route protocol bgp

inet.0: 15 destinations, 21 routes (15 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/16        *[BGP/170] 1d 21:43:42, localpref 100
                      AS path: 64501 65500 65501 65502 65503 I
                    > to 172.30.1.2 via ge-0/0/3.0
                    [BGP/170] 1d 21:43:42, localpref 100, from 192.168.100.2
                      AS path: 64502 65400 65501 65502 65503 I
                    > to 172.24.1.2 via ge-0/0/1.0
…
```

Note: You can add options to filter the output by BGP attributes such as AS path, next hop, and community.

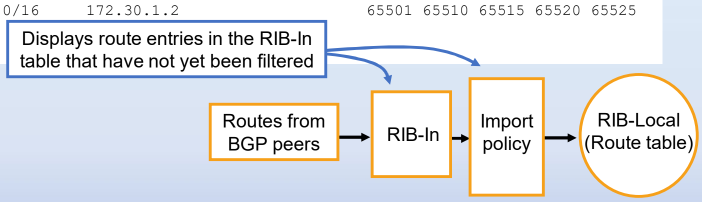
45

Case Study: Displaying BGP Routes (2 of 3)

- Use **`show route receive-protocol bgp`** ***neighbor*** to display received routes (RIB-In):

```
user@R1> show route receive-protocol bgp 172.30.1.2

inet.0: 14 destinations, 20 routes (14 active, 0 holddown, 0 hidden)
  Prefix          Nexthop      MED   Lclpref    AS path
* 10.0.0.0/16     172.30.1.2                    65501 65510 65515 65520 65525
I
* 10.1.0.0/16     172.30.1.2                    65501 65510 65515 65520 65525
I
* 10.2.0.0/16     172.30.1.2                    65501 65510 65515 65520 65525
I
…
```



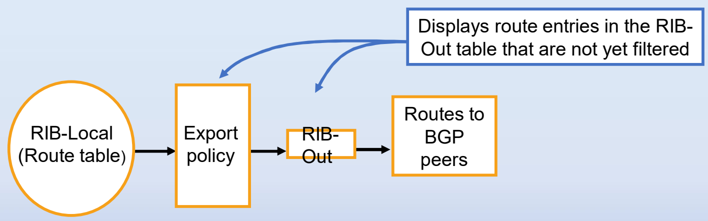
46

Case Study: Displaying BGP Routes (3 of 3)

- Use show route advertising-protocol bgp neighbor to display advertised routes (RIB-Out):

```
user@R1> show route advertising-protocol bgp 172.30.1.2

inet.0: 14 destinations, 20 routes (14 active, 0 holddown, 0 hidden)
  Prefix                  Nexthop              MED     Lclpref    AS path
* 172.24.0.0/22           Self                                    I
```



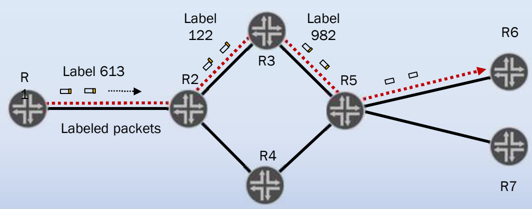
47

- **Routing konfigurálás áttekintése**
  - Statikus routing beállítása
  - OSPF konfigurálás
  - BGP konfigurálás
  - **MPLS konfigurálás**
- Access listák használata
- Specifikus alapkonfiguráció áttekintése

48

MPLS Foundations

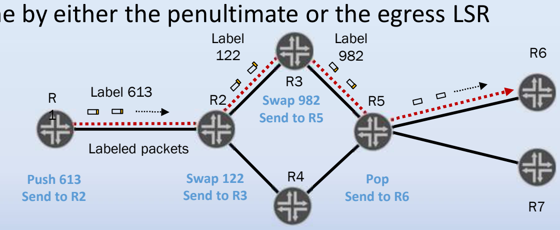
- A key fact: labels have only local significance
  - At every hop, the label of a packet sent through an MPLS label-switched path will typically change
  - Labels are not tied to the incoming interface
    - A packet with a given label will be treated the same way regardless from the interface it has been received on
    - There are only a few exceptions to this rule (e.g., carrier-of-carriers)



49

MPLS terminology

- MPLS label operations
  - Push: add an MPLS label to a packet
    - Typically done by the ingress LSR
  - Swap: replaces the incoming label value with another
    - Typically done by the transit LSR
  - Pop: removes a MPLS label from a packet
    - Typically done by either the penultimate or the egress LSR



50

Interface Configuration

- Configuring an interface to process MPLS frames
  - Enable `family mpls` at the logical unit level

```
[edit interfaces]
user@R2# show
ge-1/0/0 {
    unit 0 {
        family inet {
            address 172.20.100.21/30;
        }
        family mpls;
    }
}
```

51

Protocol Configuration

- Specify which interfaces will be running MPLS
  - Configured under protocols hierarchy

```
[edit protocols]
user@R2# show
mpls {
    interface ge-1/0/0.0;
}
```

  - You can also configure MPLS to include all interfaces

```
[edit protocols]
user@R2# show
mpls {
    interface all;
}
```

52

Configuring Static LSPs – ingress router

- Static LSP configuration on the ingress router

```
protocols {
    mpls {
        static-label-switched-path <lsp-name> {
            ingress {
                next-hop <address of next-hop router>;
                to <lsp endpoint>;
                push <label>;
            }
        }
    }
}
```

- The `push` and `next-hop` parameters will define the forwarding action
- The `to` parameter will be installed into the `inet.3` table to keep track of the LSP egress point

53

Configuring Static LSPs – transit router

- Static LSP configuration on a transit router

```
protocols {
    mpls {
        static-label-switched-path <lsp-name> {
            transit <incoming-label> {
                next-hop <address of next-hop router>;
                swap <outgoing label>;
            }
        }
    }
}
```

- On the penultimate router the action will be `pop` rather than `swap`
- There is no need to configure the egress router when doing penultimate hop popping

54

- Routing konfigurálás áttekintése
  - Statikus routing beállítása
  - OSPF konfigurálás
  - BGP konfigurálás
  - MPLS konfigurálás
- Access listák használata
- Specifikus alapkonfiguráció áttekintése

55

ACL on Juniper / Junos

Junos uses the concept of a firewall filter instead of an Access Control List (ACL), but they are essentially the same thing: A stateless packet filter.

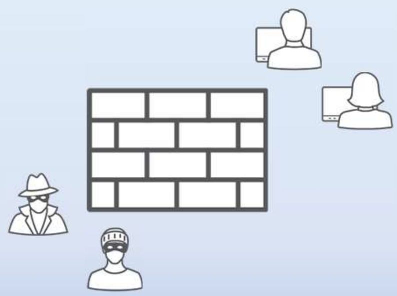
In the Cisco world you define a filter as an ACL typically apply it to an interface either inbound or outbound. Like this:

```
access-list 101 permit ip 10.1.1.0 0.0.0.255 any
access-list 101 deny ip any any
interface GigabitEthernet0
ip access-list 101 out
```

56

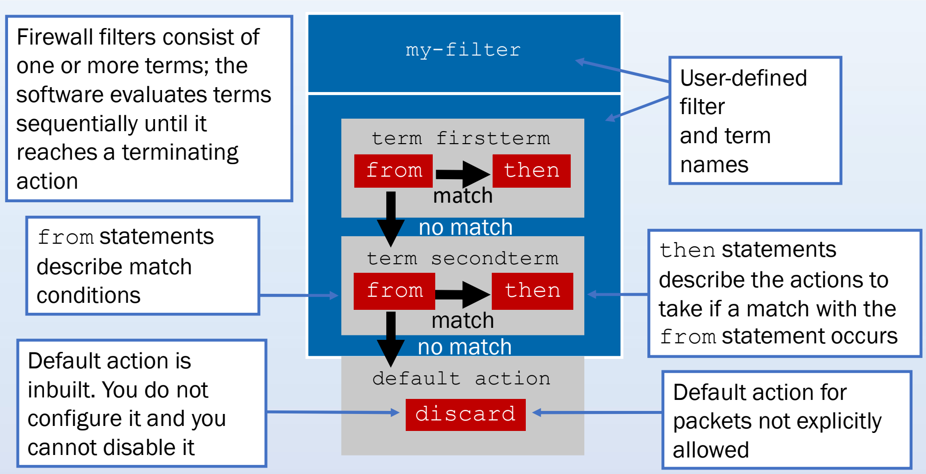
What Is a Firewall Filter?

- Firewall filters control the traffic entering and leaving a networking device in a stateless fashion:
  - Processes every packet independently
  - Used to filter and monitor network traffic



57

Building Blocks of Firewall Filters



Note: Ordering matters! If you must reorder terms within a filter, consider using the **`insert`** CLI command.

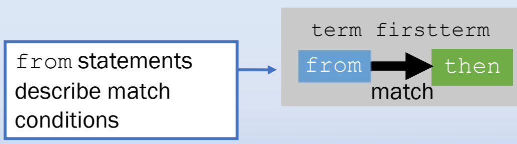
58

Common Match Criteria

- Can match based on most header fields:

```
▷ Internet Protocol, Src Addr: 10.0.1.100 (10.0.1.100), Dst Addr: 10.0.1.70 (10.0.1.70)
▷ Transmission Control Protocol, Src Port: 1307 (1307), Dst Port: telnet (23), Seq: 0, Ack: 0,
▷ Telnet
```

- Match conditions categories include:
  - Numeric range
  - Address
  - Bit field



59

Family Address Types for the fw filter

- For a filter that is applied to a **port** or **VLAN** to filter L2 and L3 packets:
  #set firewall family ethernet-switching
- For a filter that is applied to a **L3** (routed) **interface**:
  #set firewall family inet (to filter IPv4 packets)
  #set firewall family inet6 (to filter IPv6 packets)

60

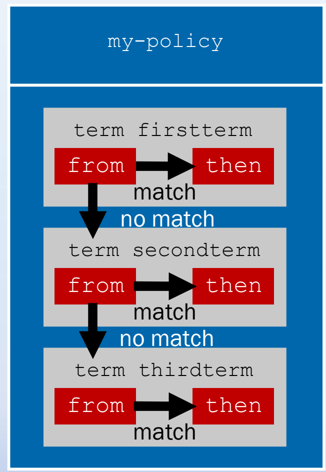
(1 of 2)

- Define firewall filters based on protocol family under the `[edit firewall]` hierarchy level:

```
[edit firewall family inet]
user@router# show
filter filter-in {
    term block-some-packets {
        from {
            source-address {
                10.10.10.0/24;
            }
        }
        then {
            count spoof-in;
            discard;
        }
    }
    term accept-others {
        then accept;
    }
}
…
```

The software applies `family inet` filters only to interfaces running IPv4

If `discard` is not present then packets are accepted



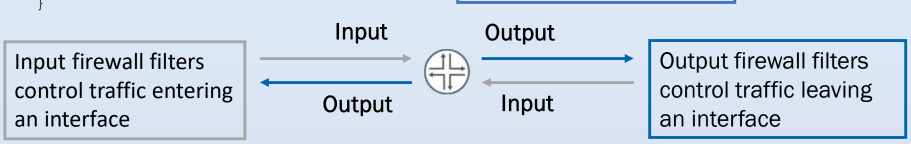
61

(2 of 2)

- Apply firewall filters as input or output on an interface
  - Protocol family on interface and filter must match:

```
[edit interfaces ge-0/0/1]
user@router# show
unit 0 {
    family inet {
        filter {
            input filter-in;
            output filter-out;
        }
        address 172.30.25.2/30;
    }
}
```

The software applies firewall filters using `input` and `output` statements



Tip: To avoid late night drives back to the office, use **`commit confirmed`** when activating filters!

62

ACL on Junos – sample filter

In the Junos world you would define the filter under firewall section. Like this:

```
Firewall {
        family inet {
                filter 101 {
                        term allow-10.1.1.0/24 {
                                from {
                                        source-address {
                                                10.1.1.0/24;
                                        }
                                }
                                then accept;
                        }
                        term reject {
                                then {
                                        reject;
                                }
                        }
                }
        }
}
```

63

ACL on Junos – the filter applied on an interface

```
Ge-0/0/0 {
        unit 0 {
                family inet {
                        filter          {
                                output 101;
                        }
                }
        }
}
```

64

Firewall Filter - sample

set firewall family inet filter **pelda_filter** term **terminal_access** from source-address 192.168.x-1.0/24
set firewall family inet filter **pelda_filter** term **terminal_access** from source-address 10.1.1.0/24
set firewall family inet filter **pelda_filter** term **terminal_access** from protocol tcp
set firewall family inet filter **pelda_filter** term **terminal_access** from port ssh
set firewall family inet filter **pelda_filter** term **terminal_access** from port telnet
set firewall family inet filter **pelda_filter** term **terminal_access** then accept
set firewall family inet filter **pelda_filter** term **terminal_access_denied** from protocol tcp
set firewall family inet filter **pelda_filter** term **terminal_access_denied** from port ssh
set firewall family inet filter **pelda_filter** term **terminal_access_denied** from port telnet
set firewall family inet filter **pelda_filter** term **terminal_access_denied** then log
set firewall family inet filter **pelda_filter** term **terminal_access_denied** then reject
set firewall family inet filter **pelda_filter** term **default-term** then accept

set interfaces ge-0/0/0 unit 0 family inet filter input pelda_filter

65

- Routing konfigurálás áttekintése
  - Statikus routing beállítása
  - OSPF konfigurálás
  - BGP konfigurálás
  - MPLS konfigurálás
- Access listák használata
- **Specifikus alapkonfiguráció áttekintése**

66

IPv6

67

Agenda: IPv6

→Introduction to IPv6

Routing Protocol Configuration Examples

68

What Is IPv6?

Next Generation Protocol:
- Defined by the IETF
- Defined as RFC 2460
- Intended to replace IPv4

69

IPv4 Versus IPv6

IPv4	IPv6
32-bit (4-byte) address supports 4,294,967,296 addresses	128-bit (16-byte) address supports 2^{128} (about 3.4×10^{38}) addresses
NAT can be used to extend address space limitations	Does not support NAT by design
Administrators must use DHCP or static configuration to assign IP addresses to hosts	Hosts use stateless address autoconfiguration to assign an IP address to themselves
IPsec support is optional	IPsec support is necessary
Options are integrated into the base header	Improved support for options using extension headers and overall simplification of the header format

70

IPv6 Addressing

- Extending address space is a major reason for IPv6
  - IPv4 address exhaustion is predicted to occur in the near future
  - 128-bit (16-byte) address supports 2^{128} (about 3.4×10^{38}) addresses
  - 2^{95} addresses for each person on Earth
  - 2^{52} addresses for each observable star in the known universe

71

IPv6 Address Types

- Address types:
  - Unicast: Unique address that identifies an IPv6 node
  - Multicast: Group of IPv6 interfaces
  - Anycast: Assigned to multiple interfaces on multiple nodes

72

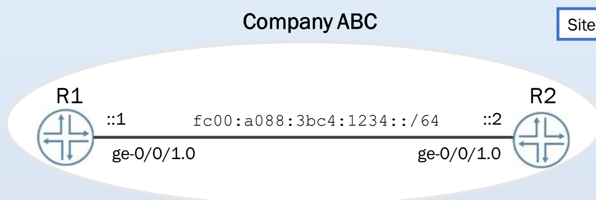
Interface Configuration Example

Use family inet6 for IPv6 operations

```
[edit interfaces]
user@R1# show
ge-0/0/1 {
    unit 0 {
        family inet6 {
            address fc00:a088:3bc4:1234::1/64;
        }
    }
}
```

```
[edit interfaces]
user@R2# show
ge-0/0/1 {
    unit 0 {
        family inet6 {
            address fc00:a088:3bc4:1234::2/64;
        }
    }
}
```

Site-local addresses



#set interfaces ge-0/0/1 unit 0 family inet6 address fc00:a088:3bc4:123::1/64

73

Interface Verification Example

- Use the **`show interface terse`** command to verify interface status and basic details

```
user@R1> show interfaces terse ge-0/0/1
Interface               Admin Link Proto    Local                   Remote
ge-0/0/1                up    up
ge-0/0/1.0              up    up   inet6    fe80::226:88ff:fe02:7481/64
                                            fc00:a088:3bc4:1234::1/64
```

Each interface has automatically determined its own link-local address

```
user@R2> show interfaces terse ge-0/0/1
Interface               Admin Link Proto    Local                   Remote
ge-0/0/1                up    up
ge-0/0/1.0              up    up   inet6    fe80::226:88ff:fe02:6b81/64
                                            fc00:a088:3bc4:1234::2/64
```

74

Displaying IPv6 Routing Information

- Use the **show route table inet6** command to view IPv6 routing information

```
user@R1> show route table inet6

inet6.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

fe80::/64          *[Direct/0] 00:59:12
                    > via ge-0/0/1.0
fe80::226:88ff:fe02:7481/128
                   *[Local/0] 00:59:12
                      Local via ge-0/0/1.0
fc00:a088:3bc4:1234::1/64 *[Direct/0] 00:59:12
                    > via ge-0/0/1.0
fc00:a088:3bc4:1234::1/128
                   *[Local/0] 00:59:12
                      Local via ge-0/0/1.0
```

75

Agenda: IPv6

- Introduction to IPv6

→Routing Protocol Configuration Examples

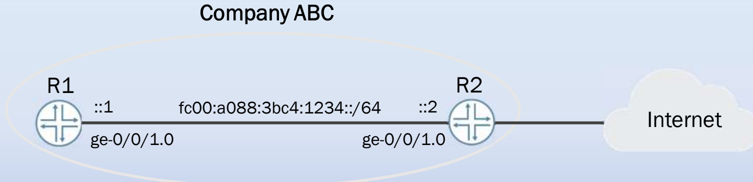
76

Static Route Configuration Example

- Sample IPv6 static route configuration:

```
[edit routing-options]
user@R1# show
rib inet6.0 {
    static {
        route 0::/0 {
            next-hop fc00:a088:3bc4:1234::2;
            preference 250;
        }
    }
}
```

IPv6 default static route



77

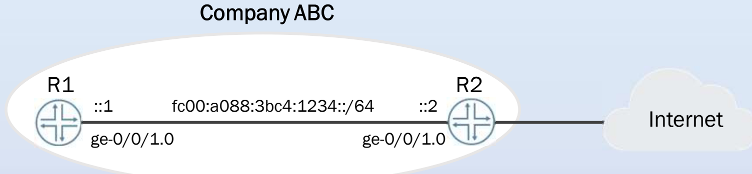
Displaying the Static Routes

- Use the **`show route table inet6.0 protocol static`** command to view static routes

```
user@R1> show route table inet6.0 protocol static

inet6.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

::/0               *[Static/5] 00:00:24
                    > to fc00:a088:3bc4:1234::2 via ge-0/0/1.0
```



78

OSPFv3 Configuration Example

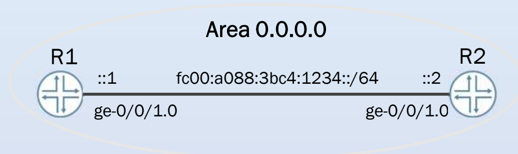
- Sample OSPFv3 single-area configuration
  - OSPFv3 continues to use a 32-bit RID

```
[edit]
user@R1# show routing-options router-id
router-id 192.168.100.1;

[edit]
user@R1# show protocols ospf3
area 0.0.0.0 {
    interface ge-0/0/1.0;
}
```

```
[edit]
user@R2# show routing-options router-id
router-id 192.168.100.2;

[edit]
user@R2# show protocols ospf3
area 0.0.0.0 {
    interface ge-0/0/1.0;
}
```



Note: The RID selection process is the same for OSPFv2 and OSPFv3. We recommend you manually set the RID, as shown in the example.

79

Monitoring OSPFv3 Operations

- Most operational **show** commands for OSPFv3 are almost identical to OSPFv2; the key difference is you must replace **ospf** with **ospf3**:

OSPFv2	OSPFv3
`show ospf neighbor`	`show ospf3 neighbor`
`show ospf interface`	`show ospf3 interface`
`show ospf database`	`show ospf3 database`
`show ospf route`	`show ospf3 route`

80

BGP Configuration Example

- BGP configuration is almost identical for IPv6 as it is for IPv4; the major difference is you specify an IPv6 address for the local and peer addresses:

```
[edit routing-options]
user@R1# show
router-id 192.168.100.1;
autonomous-system 64700;

[edit protocols bgp]
user@R1# show
group int-64700 {
    type internal;
    local-address fc00:0:0:1001::1;
    neighbor fc00:0:0:1002::1;
}
group ext-65100 {
    type external;
    peer-as 65100;
    neighbor fc00:0:0:2005::2;
}
```

81

Monitoring BGP Operations

- Use the same operational **show** commands for BGP in IPv4 and IPv6 environments:

```
user@R1> show bgp summary
Groups: 2 Peers: 2 Down peers: 0
Table          Tot Paths  Act Paths Suppressed    History Damp State    Pending
inet6.0                0          0          0          0          0          0
Peer                     AS      InPkt     OutPkt    OutQ   Flaps Last Up/Dwn
State|#Active/Received/Accepted/Damped...
fc00:0:0:1002::1       64700         11         12       0       0        4:00 Establ
  inet6.0: 0/0/0/0
fc00:0:0:2005::2       65100         11         12       0       0        4:05 Establ
  inet6.0: 0/0/0/0
```

82

Extras

83

SRX átkapcsolása router módba

```
> request system zeroize
# set system root-authentication plain-text-password

> show security flow status

# set security forwarding-options family mpls mode packet-based
# set security forwarding-options family iso mode packet-based
# set security forwarding-options family inet6 mode packet-based
# delete security
```

84

Pár hasznos parancs

- show | compare – ezzel megnézhetjük, hogy az utolsó commit óta mit állítottunk be
- rollback 0 – vissza tudunk térni az utolsó commit utáni állapothoz / kiresetelhetjük azokat a parancsokat, amiket az utolsó commit óta beírtunk
- show configuration | display set – konfiguráció “másolható/szerkeszthető” kilistázása
- show configuration | display set | no-more – konfig egybefüggő kilistázása

85

Pár hasznos parancs – infók a HW-ről

- show system memory – rendszer memória elosztási és használati információk megjelenítése
- show chassis hardware – hardver infók megjelenítése
- show chassis hardware detail - +RAM és disk infók is
- show chassis routing engine – Control plane CPU felhasználási infók
- show chassis forwarding – data plane CPU felhasználási infók
- request support information – mindenről is infó

86

87