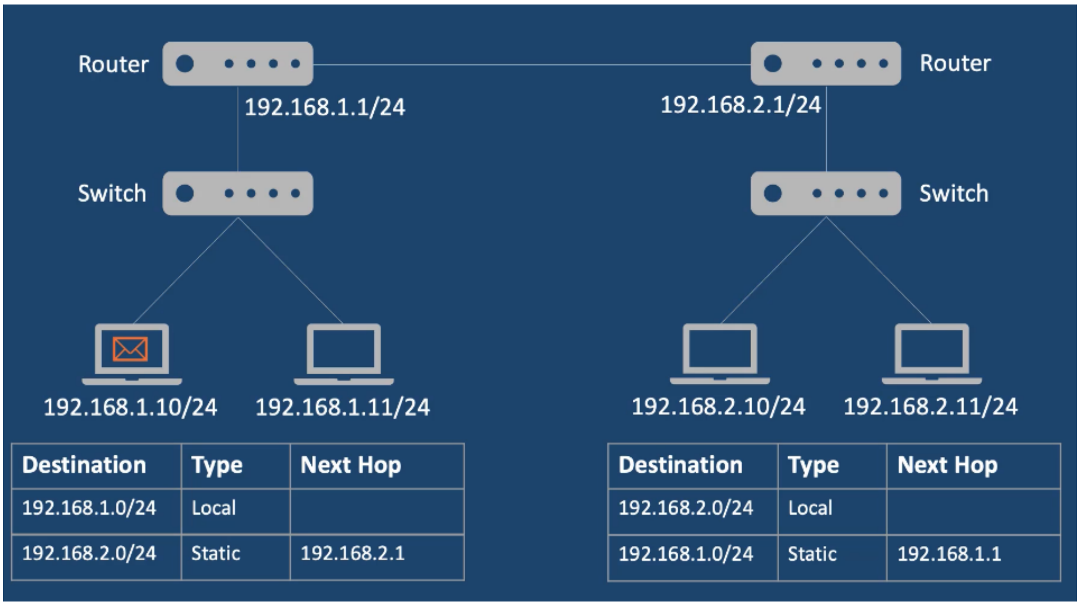


Preferred address - when destined to same subnet

Primary adddress - when destined to different subnet

Static vs Dynamic Routing

Static Routing

- Routing table have to be created and updated manually
- For full connectivity, a route must be configured for every network
- Isn’t fault tolerant - any changes on the network, like a link going down is not detected and routes need to be updated manually
- Works well for small networks, but becomes an administrative overhead for large networks

Dynami Routing

- Routing tables are created and updated by a routing protocol
- Routing information is automatically shared between participating devices
- Is fault tolerant - changes on the network are detected and routes are automatically adjusted
- Easy to configure and works well for large networks
- Examples - RIP, OSPF

Junos Routing

- Junos operating system maintains two databases for routing information:
  - Routing table – contains all the routing information learned by all routing protocols
  - Forwarding table – contains the routes actually used to forward packets through the router

Routing Overview

- Routing is the transmission of packets from a source to a destination address
- A routing protocol determines the path by which the packets are forwarded, and shares this information with immediate neighbor devices and other devices in the network
- It also adjusts to changing network conditions

Junos Routing Table

- Used by the routing protocol processes to maintain a database of routing information
- In this table, the routing protocol process stores statically configured routes, directly connected interface routes, and all routing information learned from all routing protocols
- The routing protocol process uses this collected routing information to select the **active route** to each destination – this route is used to forward packets

Junos 3 Types of Routing Table

- By default, Junos maintains three routing tables:
  - unicast routing table – stores routing information for all unicast routing protocols – BGP, IS-IS, OSPF, RIP
  - multicast routing table – stores routing information for all multicast routing protocols – DVMRP, PIM
  - MPLS routing table – stores MPLS path and label information

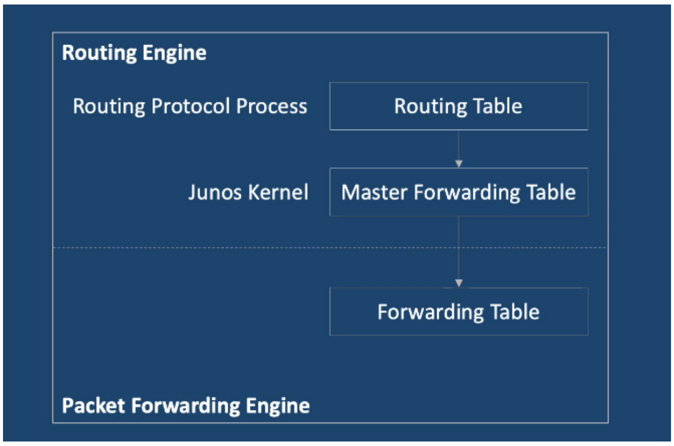
Junos Default Routing Table

- `inet.0` – default IPv4 unicast routing table
- `inet.1` – IPv4 multicast forwarding cache
- `inet.2` – unicast routes for multicast reverse path forwarding (RPF) lookup
- `inet.3` – IPv4 MPLS routing table for path information
- `inet6.0` – default IPv6 unicast routing table
- `inet6.1` – IPv6 multicast forwarding cache
- `mpls.0` – for MPLS label switching operations
- `iso.0` – used for IS-IS routing

Forwarding Table

- All active routes from the routing table are installed into the forwarding table
- Active routes are used to forward packets to their destinations
- The Junos kernel maintains a master copy of the forwarding table
- It copies the forwarding table to the packet forwarding engine, which is responsible for forwarding packets

- If the routing table is a list of all possible paths a packet can take, the forwarding table is a list of only the best routes to a destination
- The best route is determined based on the routing protocol being used, but generally the number of hops between the source and destination determine the best possible route



Terminal Junos Route

```
root@SRX> show route

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

0.0.0.0/0          *[Static/5] 00:33:50
                    >  to 10.0.30.1 via ge-0/0/0.0
10.0.10.0/24       *[Direct/0] 00:34:33
                    >  via fxp0.0
10.0.10.254/32     *[Local/0] 00:34:33
                       Local via fxp0.0
10.0.20.0/24       *[Direct/0] 00:33:50
                    >  via ge-0/0/1.0
10.0.20.254/32     *[Local/0] 00:33:50
                       Local via ge-0/0/1.0
10.0.30.0/24       *[Direct/0] 00:33:50
                    >  via ge-0/0/0.0
10.0.30.254/32     *[Local/0] 00:33:50
                       Local via ge-0/0/0.0

inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

ff02::2/128        *[INET6/0] 00:34:33
                       MultiRecv
```

active - routes used by the system to forward traffi
holddown - routes that are in pending state before the system declares them as inactive. A holddown route was one the active route and is no longer the active route
hidden - routes that are not used because of a routingp policy

[local/0] - this is a preference value of 0 after /, both local and direct are with preference value of 0

Terminal Junos Forwarding

```
root@SRX> show route forwarding-table
Routing table: default.inet
Internet:
Enabled protocols: Bridging,
Destination        Type RtRef Next hop           Type Index    NhRef Netif
default            user     1 2:7f:a5:fb:64:bc   ucst      563     4 ge-0/0/0.0
default            perm     0                    rjct       36     1
0.0.0.0/32         perm     0                    dscd       34     1
10.0.10.0/24       intf     0                    rslv      323     1 fxp0.0
10.0.10.0/32       dest     0 10.0.10.0          recv      321     1 fxp0.0
10.0.10.1/32       dest     0 2:54:7d:45:b5:70   ucst      324     1 fxp0.0
10.0.10.100/32     dest     1 2:e5:f7:52:35:e    ucst      325     2 fxp0.0
10.0.10.254/32     intf     0 10.0.10.254        locl      322     2
10.0.10.254/32     dest     0 10.0.10.254        locl      322     2
10.0.10.255/32     dest     0 10.0.10.255        bcst      320     1 fxp0.0
10.0.20.0/24       intf     0                    rslv      558     1 ge-0/0/1.0
10.0.20.0/32       dest     0 10.0.20.0          recv      556     1 ge-0/0/1.0
10.0.20.1/32       dest     0 2:50:e3:71:f6:c8   ucst      564     1 ge-0/0/1.0
10.0.20.254/32     intf     0 10.0.20.254        locl      557     2
10.0.20.254/32     dest     0 10.0.20.254        locl      557     2
10.0.20.255/32     dest     0 10.0.20.255        bcst      555     1 ge-0/0/1.0
10.0.30.0/24       intf     0                    rslv      554     1 ge-0/0/0.0
10.0.30.0/32       dest     0 10.0.30.0          recv      552     1 ge-0/0/0.0
10.0.30.1/32       dest     0 2:7f:a5:fb:64:bc   ucst      563     4 ge-0/0/0.0
10.0.30.254/32     intf     0 10.0.30.254        locl      553     2
10.0.30.254/32     dest     0 10.0.30.254        locl      553     2
10.0.30.255/32     dest     0 10.0.30.255        bcst      551     1 ge-0/0/0.0
224.0.0.0/4        perm     0                    mdsc       35     1
224.0.0.1/32       perm     0 224.0.0.1          mcst       31     1
255.255.255.255/32 perm     0                    bcst       32     1
```

Usage or forwarding entries permanent in nature

- Junos kernel adds some forwarding entries and considers them **permanent in nature**.
- One such example is the default forwarding entry, which matches all packets when no other matching entry exists.
- <u>When a packet matches the default forwarding entry, the router discards the packet and sends an ICMP destination unreachable message back to the sender.</u>
- If a user-defined default route is configured, the router uses it instead of the permanent default forwarding entry.

Forwarding Route Types

- **perm**: routes installed by the kernel when the routing table initializes
- **dest**: remote addresses directly reachable through an interface
- **intf**: installed as a result of configuring an interface
- **user**: routes installed by the routing protocol process or as a result of the configuration
- **iddn**: destination route for which the interface is unreachable
- **ignr**: ignore this route

Next Hope Types

- ucst: unicast
- bcst: broadcast
- hold: next hop is waiting to be resolved into a unicast or multicast type
- locl: local address of an interface
- indr: indirect next hop
- dscd: discard silently without sending an ICMP unreachable msg
- reject: discard and send an ICMP unreachable message

show pfe route ip

- ***show forwarding table from PFE***

Route Preference

- Junos assigns a default preference value to each route that the routing table receives
- This is same as administrative distance on equipment from other vendors
- The default value depends on the source of the route
- It ranges from 0 through 4,294,967,295 (2^32-1)
- A lower value indicates a more preferred route

Routing Information Source	Default Preference
Directly connected route	0
Static route	5
OSPF internal route	10
RIP	100
BGP	170

- *Default preference value for most routing information sources can be modified to make them more or less desirable*
- The exception to this is **direct / local routes** - these are always preferred regardless of the modified route preference associated with other routing information sources
- If **multiple equal-cost paths exist for the same destination**, the routing protocol daemon (rpd) randomly selects one of the available paths. (provides load distribution)

Static Routes

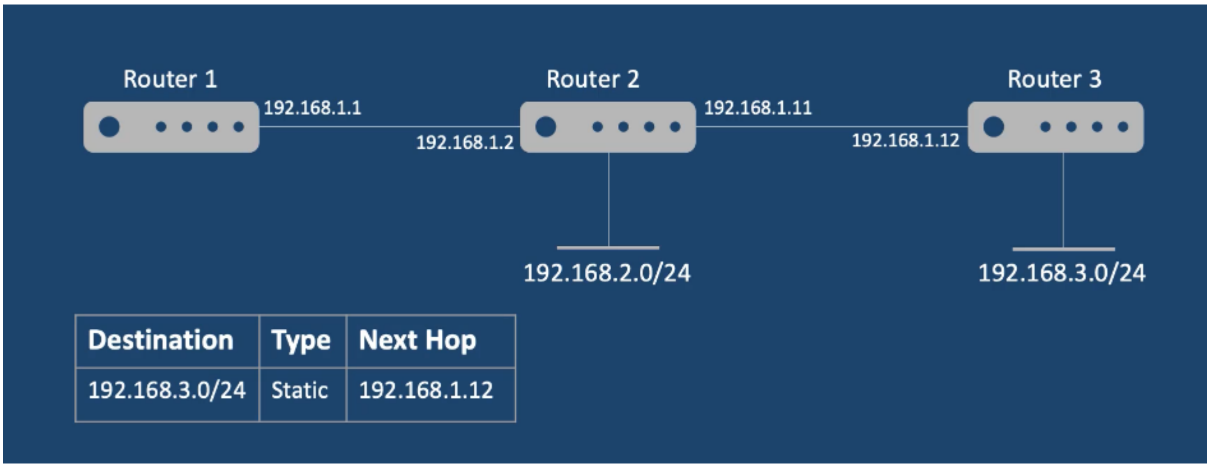
\# edit routing-options
\# set static route <des ip> next-hop <nxt hop ip>
\# commit
—-----------
we can configure static route **only if nxt hop ip is reachable**

- Routes that are permanent fixtures in the routing and forwarding tables are often configured as static routes
- These routes generally do not change, and are not manipulated by external protocols

- To create a static route in the routing table, you must, at minimum, define the route as static, define the network you want to route to, and associate a next-hop IP address with it
- The static route is inserted into the forwarding table when the next-hop address is reachable
- All traffic destined for the static route is transmitted to the next-hop IP address

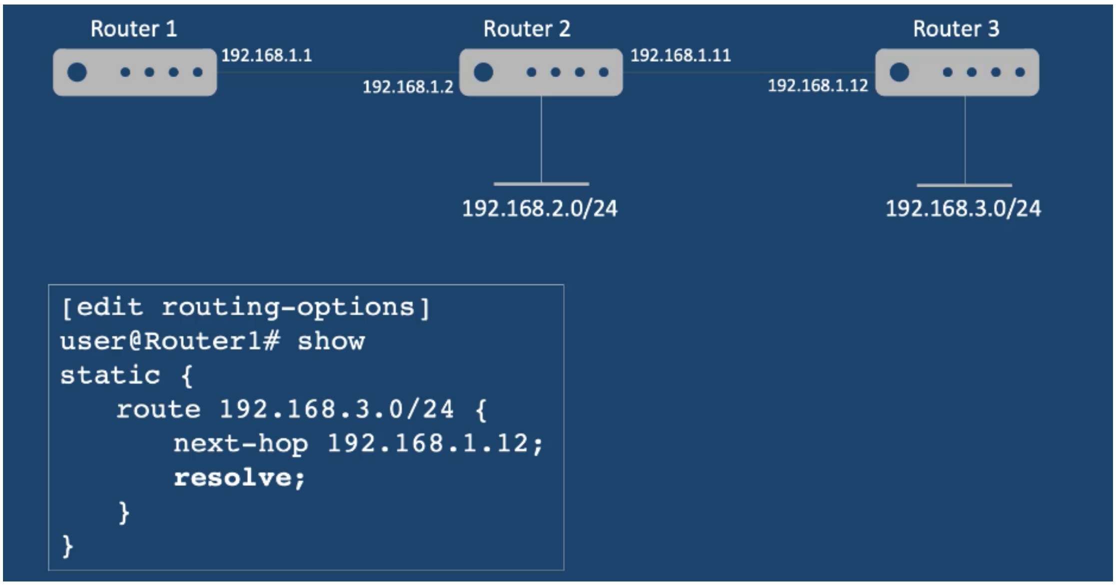
Resolve Indirect Hop

Note that Next Hop is **not directly reachable**



Resolve

- By default, Junos requires that next hop IP address of static routes be reachable using direct route
- Unlike software from other vendors, Junos does not perform recursive lookups of next hops by default
- This behavior can be changed by using the ***resolve*** keyword
- In addition to the solve option, **a route to the indirect next hop is also required**



No Re-advertise

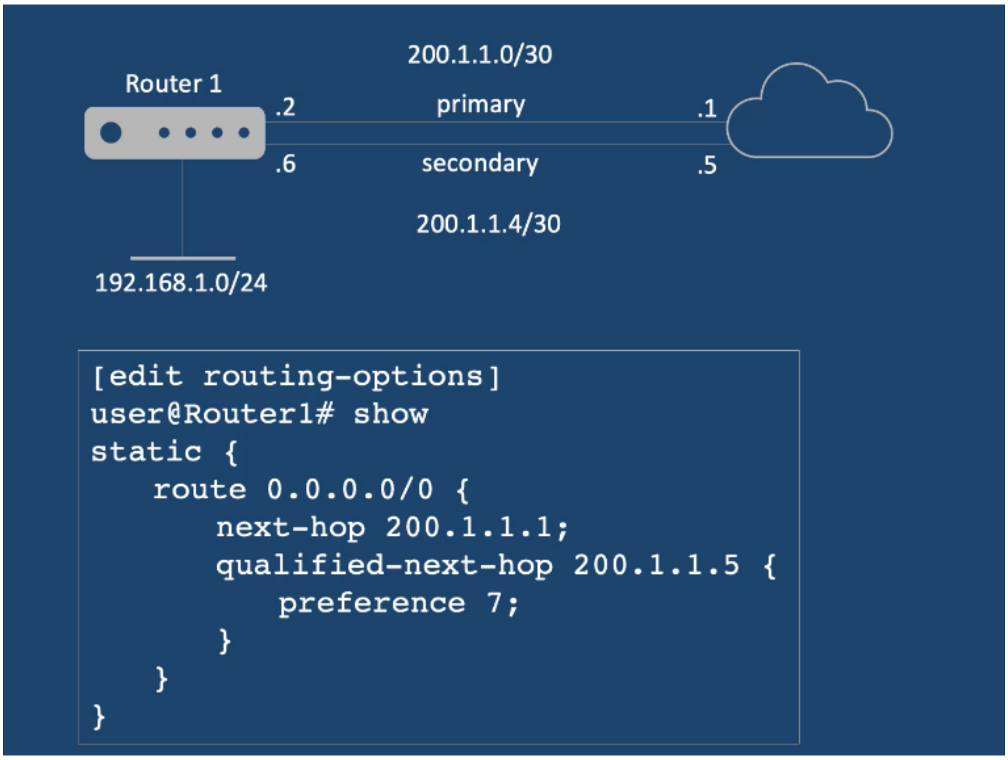
- Static routes are eligible to be advertised - that is, exported from the routing table into dynamic routing protocols, if a policy to do so is configured
- To mark an **IPv4 static route as being ineligible for re-advertisement**, include the ***no-readvertise*** statement

- It is recommended to use the no-readvertise option on static routes used for management traffic

 # set static route <ip> no-readvertise

Qualified Next Hop

- A static route can have multiple next hops associated with it
- In such cases, multiple routes are inserted into the routing table, and Junos must make a route selection
- **By default, Junos chooses in a random fashion one of the next-hop addresses** to install into the forwarding table
- Qualified next hop allow you to configure multiple next-hop addresses for a route and have them treated differently
- **It allows you to specify a different preference value for the qualified next hop**



 # set static route <ip> <topic>

Route Retention

- By default, static routes are not retained in the **forwarding table** when the routing process shuts down
- When the routing process starts up again, any routes configured as static routes must be added to the forwarding table again
- Causes latency
- To avoid this, routes can be configured as retain, causing them to kept in the forwarding table even after the routing process shuts down

- Retention ensures that the routes are always in the forwarding table, even immediately after a system reboot.

 # set static route <ip> <topic>

Passive Route

- When a static routes' next hop address is unreachable, the route is marked as passive, and it is removed from the routing or forwarding tables
- Marking a route as **passive** <u>forces the route to be included in the routing tables</u> regardless of next-hop reachability
- If a route is flagged as passive and its next-hop address is unreachable, the route is included in the routing table, and all traffic destined for the route is rejected.

 # set static route <ip> <topic>

Dynamic Routing

- Static routing works well for smaller networks or when you need tight control over routing
- On large networks, static routing is hard to manage
- For large networks or networks that change frequently, dynamic routing is a better choice
- *<u>With dynamic routing, you configure the network interfaces to participate in a routing protocol</u>*
- *<u>Devices running routing protocols can dynamically learn routing information from each other</u>*
- When there is a change on the network, routing information is automatically updated

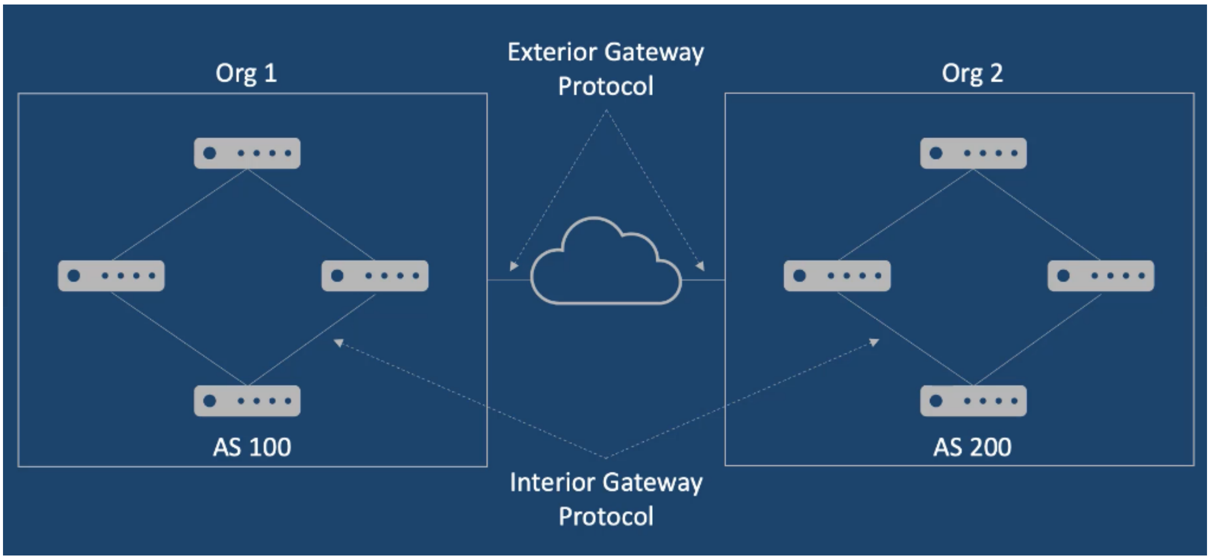
Advantages
- Easy to configure - devices learn routing information automatically, eliminating the need for manual routing entries
- Increased network availability - when routes change or fail, dynamic routing reroutes traffic automatically
- Better network scalability - when the network grows, new routes can be easily learn

Autonomous System - a collection of routers under a common administrative domain
Dynamic routing protocols can be classified as **Exterior Gateway** and **Interior Gateway Protocols**

Types of Dynamic Routings

- Routing protocols used for routing between autonomous systems (inter-autonomous system routing) are called Exterior Gateway Protocols (EGPs)
- Example – Border Gateway Protocol (BGP)

- Routing protocols used for routing inside an autonomous system (intra-autonomous system routing) are called Interior Gateway Protocols (IGPs)
- Examples – RIP, OSPF, IGRP
- Interior Gateway Protocols are classified as Distance Vector and Link-State routing protocols



Distance Vector Routing Protocol

- Routers that share a link and configured to use the same routing protocol are called as Neighbors
- With distance vector routing protocols, routing updates are shared with neighbors
- Routing updates include a distance vector, typically expressed as the number of hops to the destination

- Routing updates are flooded out all protocol-enabled interfaces at regular intervals
- The router only knows about its own interfaces and remote network that can be reached through its neighbors
- The router is not aware of the network topology

Link-State Routing Protocol

- Routers have complete view of the network topology
- Also known as shortest path first protocols, link-state protocols compute the best path to each destination
- Routing updates are shared with all routers

- Routing updates are sent only when there is a change on the network, and only the changes are sent
- Convergence time is less compared to distance vector protocols

Longest Route Machine

- Algorithm used by routers to select an entry from a routing table

- If multiple routes exist to a destination:
  - the router uses the longest match to determine the next-hop address
  - the routing entry that has the **longest number of network bits** that match the destination is the best match

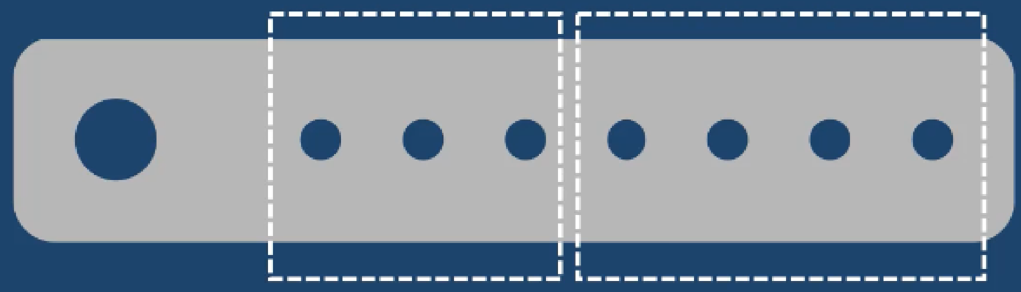
Destination	Type	Next Hop
192.168.1.64/26	Static	Router B
192.168.1.0/24	Static	Router C
192.168.0.0/16	Static	Router D

Address	Binary Representation
192.168.1.70	11000000.10101000.00000001.01000110
192.168.1.64/26	11000000.10101000.00000001.01000000
192.168.1.0/24	11000000.10101000.00000001.00000000
192.168.0.0/16	11000000.10101000.00000000.00000000

Routing Instances

- A routing instance is a collection of routing tables, interfaces, and routing protocol parameters
- The default routing instance, `master`, uses the main `inet.0` routing table
- There are can be multiple routing tables for a single routing instance – for example, route tables for unicast IPv4, unicast IPv6, and multicast IPv4 can exist in a single routing instance

- Each routing instance has a unique name and corresponding IP unicast table
- For example, if you configure a routing instance with the name `my-instance`, the corresponding IP unicast table is `my-instance.inet.0`
- By using routing instances, a single device can effectively imitate multiple devices



Routing Instance

- Each routing instance consists of:
  - routing tables
  - interfaces that belong to these routing tables
  - routing configuration

- Only one instance of a routing protocol can be configured in a single routing instance

edit routing-instances R1
set instance-type virtual-router

```
root@SRX# set instance-type ?
Possible completions:
  evpn                   EVPN routing instance
  evpn-vpws              EVPN VPWS routing instance
  forwarding             Forwarding instance
  l2backhaul-vpn         L2Backhaul/L2Wholesale routing instance
  l2vpn                  Layer 2 VPN routing instance
  layer2-control         Layer 2 control protocols
  mpls-forwarding        Routing instance provides a MPLS forwarding-context
  mpls-internet-multicast  Internet Multicast over MPLS routing instance
  no-forwarding          Nonforwarding instance
  virtual-router         Virtual routing instance
  vpls                   VPLS routing instance
  vrf                    Virtual routing forwarding instance
```

set interface ge-0/0/1.0
top
commit

```
root@SRX# run show route instance
Instance             Type
         Primary RIB                                     Active/holddown/hidden
master               forwarding
         inet.0                                          5/0/0
         inet6.0                                         1/0/0

R1                   virtual-router
         R1.inet.0                                       2/0/0
         R1.inet6.0                                      1/0/0

__juniper_private1__ forwarding
         __juniper_private1__.inet.0                     5/0/0

__juniper_private2__ forwarding
         __juniper_private2__.inet.0                     0/0/1

__juniper_private3__ forwarding

__juniper_private4__ forwarding

__master.anon__      forwarding

mgmt_junos           forwarding
```

```
root@SRX# run show route

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

0.0.0.0/0          *[Static/5] 00:53:28
                    >  to 10.0.30.1 via ge-0/0/0.0
10.0.10.0/24       *[Direct/0] 00:54:11
                    >  via fxp0.0
10.0.10.254/32     *[Local/0] 00:54:11
                       Local via fxp0.0
10.0.30.0/24       *[Direct/0] 00:53:28
                    >  via ge-0/0/0.0
10.0.30.254/32     *[Local/0] 00:53:28
                       Local via ge-0/0/0.0

R1.inet.0: 2 destinations, 2 routes (2 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.20.0/24       *[Direct/0] 00:00:17
                    >  via ge-0/0/1.0
10.0.20.254/32     *[Local/0] 00:00:17
                       Local via ge-0/0/1.0

inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

ff02::2/128        *[INET6/0] 00:54:11
                       MultiRecv

R1.inet6.0: 1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

ff02::2/128        *[INET6/0] 00:00:17
                       MultiRecv

[edit]
root@SRX#
```