IT Technology
Project Network Report

LILLEBAELT ACADEMY
UNIVERSITY OF APPLIED SCIENCE

This report is made by:

Dilshad Khalaf
dils0008@edu.eal.dk
Milan Kristof Vince
mila1025@edu.eal.dk
Nikolay Petrov
niko010h@edu.eal.dk

2017-06-05

Table of Content

Introduction

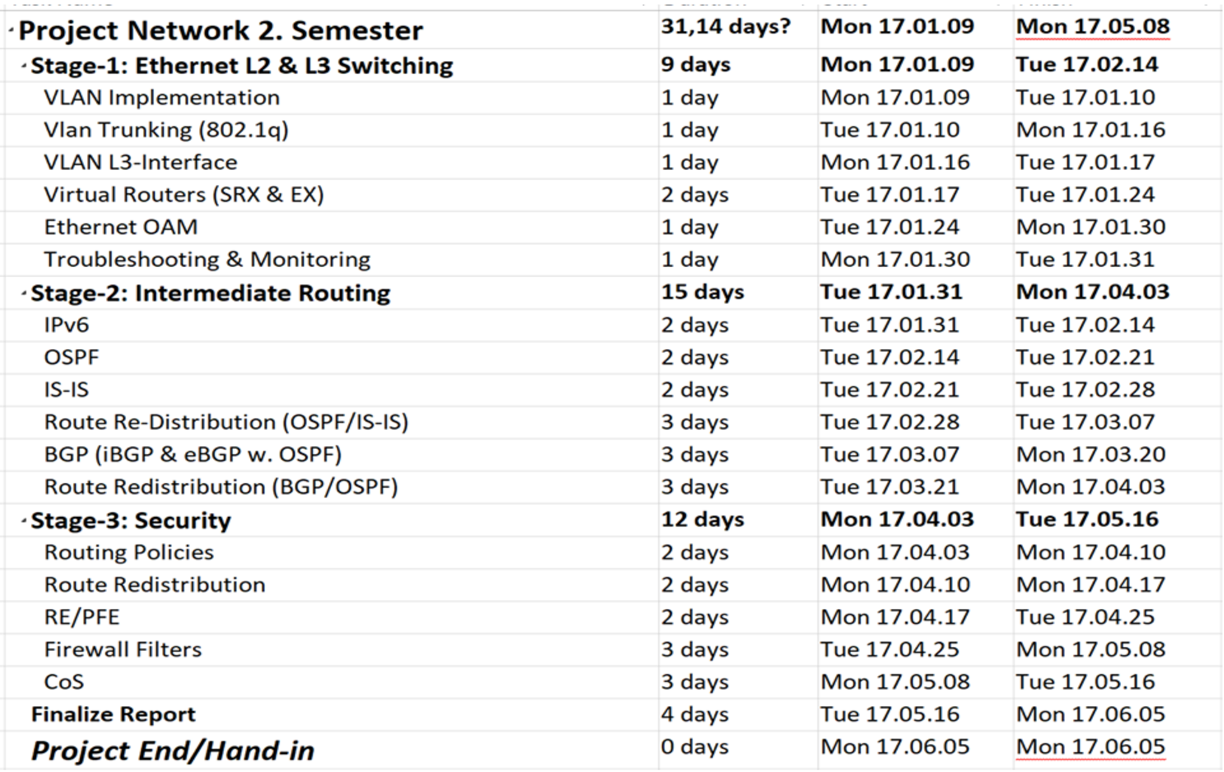
Nowadays many people are using network devices to access the internet for different purposes such as logging into their social media profiles, search information on the web and other things, without understanding what’s happening “behind the scenes”. Each year the number of users is growing because of all the devices that are used such as computers, smartphones, tablets and today even household appliances that can access the internet. The industry of technology is growing tremendously which has not been witnessed before. Various studies tell that, despite billions of people using the internet every day, a great percentage does not understand how a message can be sent from one computer to another. In this report, we have studied different components in computer networking to gain a better understanding of how everything is linked together. In our project plan that was given to our class, IT-Technology at Lillebaelt Academy, we have covered some of the most important topics and worked with different protocols such OSPF, BGP and IS-IS. All these protocols help understand how devices in a network can communicate together. It also contains how IPv4 and IPv6 works and some of the advantages IPv6 has over IPv4. But most importantly, this document covers how Ethernet works and how the TCP/IP protocol stack model is used, for computers to communicate with each other over the internet. Moreover, we have made configurations for different purposes on SRX routers, which is a product by one of the biggest companies, Juniper, that delivers networking devices such as routers, switches etc.

Project Management

IT-Security EAL - Project Plan of the second Semester

Throughout the 2nd semester our group have been working on the topics that are in the project plan as shown in the screenshot below. In relation to this report we in our group decided to split up different tasks that each group member could work on.

The project plan was given to all the students by Peter Liljehof Thomsen, and we, our group have complied with it throughout the second semester, and throughout this report. It has not been changed and we take the full responsibility to ensure that this is the original version.



Project Network 2. Semester	**31,14 days?**	**Mon 17.01.09**	**Mon 17.05.08**
Stage-1: Ethernet L2 & L3 Switching	**9 days**	**Mon 17.01.09**	**Tue 17.02.14**
VLAN Implementation	1 day	Mon 17.01.09	Tue 17.01.10
Vlan Trunking (802.1q)	1 day	Tue 17.01.10	Mon 17.01.16
VLAN L3-Interface	1 day	Mon 17.01.16	Tue 17.01.17
Virtual Routers (SRX & EX)	2 days	Tue 17.01.17	Tue 17.01.24
Ethernet OAM	1 day	Tue 17.01.24	Mon 17.01.30
Troubleshooting & Monitoring	1 day	Mon 17.01.30	Tue 17.01.31
Stage-2: Intermediate Routing	**15 days**	**Tue 17.01.31**	**Mon 17.04.03**
IPv6	2 days	Tue 17.01.31	Tue 17.02.14
OSPF	2 days	Tue 17.02.14	Tue 17.02.21
IS-IS	2 days	Tue 17.02.21	Tue 17.02.28
Route Re-Distribution (OSPF/IS-IS)	3 days	Tue 17.02.28	Tue 17.03.07
BGP (iBGP & eBGP w. OSPF)	3 days	Tue 17.03.07	Mon 17.03.20
Route Redistribution (BGP/OSPF)	3 days	Tue 17.03.21	Mon 17.04.03
Stage-3: Security	**12 days**	**Mon 17.04.03**	**Tue 17.05.16**
Routing Policies	2 days	Mon 17.04.03	Mon 17.04.10
Route Redistribution	2 days	Mon 17.04.10	Mon 17.04.17
RE/PFE	2 days	Mon 17.04.17	Tue 17.04.25
Firewall Filters	3 days	Tue 17.04.25	Mon 17.05.08
CoS	3 days	Mon 17.05.08	Tue 17.05.16
Finalize Report	4 days	Tue 17.05.16	Mon 17.06.05
Project End/Hand-in	0 days	Mon 17.06.05	Mon 17.06.05

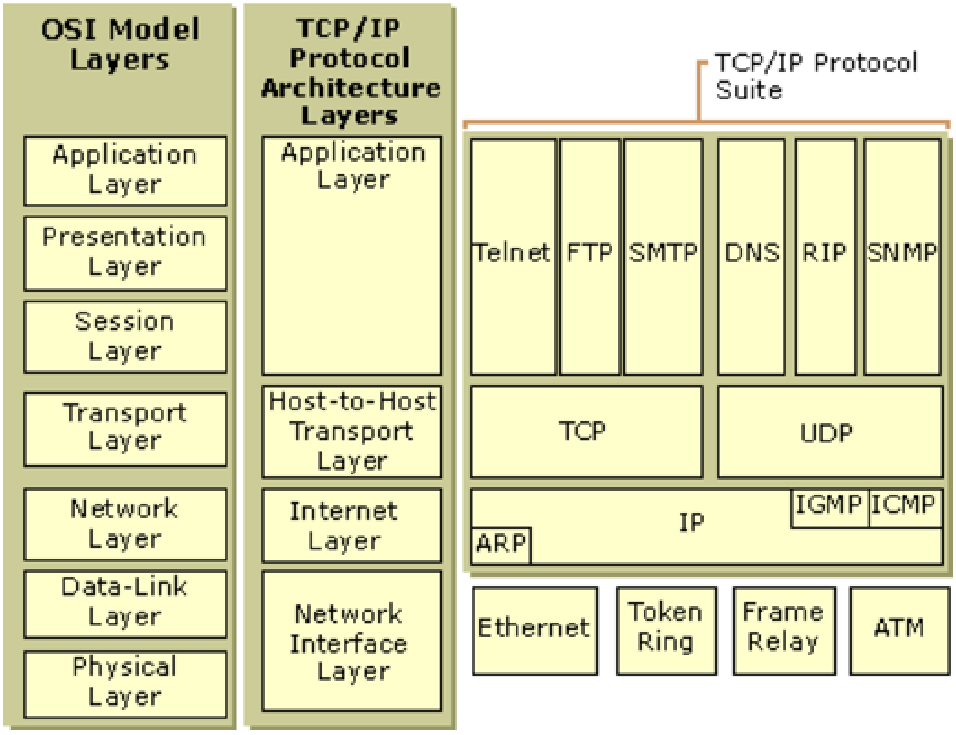
Responsibilities of each group member

TCP/IP Model	Dilshad Khalaf
Ethernet OAM	Dilshad Khalaf
VLANs	Milan Vince
RE and PFE	Milan Vince
IPv4 and IPv6	Nikolay Petrov
IPv6 Header and Configuration	Nikolay Petrov
OSPF	Dilshad Khalaf
OSPFv3	Nikolay Petrov
IS-IS	Dilshad Khalaf
BGP	Dilshad Khalaf
Class of Service	Milan Vince

TCP / IP Model

Summary

The TCP/IP Model is a four-layer communication language or protocol of the Internet. It was developed by the by the U.S. government agency and is also known as the DARPA model. This protocol consisting of different layers is used to describe how different software and hardware components are used for establishing communication from one device to another via the internet. Each layer in the TCP/IP model corresponds to one or more layers of the OSI (Open System Interconnection) model.



Application layer: The Application layer runs applications with the ability to access services used, to exchange data through the internet. There are great numbers of protocols used within the applications and new protocols are always being developed. Some of the well-known protocols include, *Telnet, SSH, FTP, HTTP, SMTP, DNS, and SNMP*.

- **Telnet:** A less secure network protocol that provide a command line interface for communication with a device.
- **Hypertext Transfer Protocol (HTTP):** Used by the World Wide Web and defines how messages are formatted and transmitted including what actions webservers and browsers should take in response to various commands.
- **File Transfer Protocol (FTP):** A data transfer protocol used for exchanging files over the internet.
- **Domain Name System (DNS):** Translates domain names into IP addresses.
- **Simple Network Management Protocol (SNMP**): Popular protocol for network management and monitoring.

Transport layer: The Transport layer resolves all host-to-host communications. The main protocols included in this layer are the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

- **TCP:** A reliable protocol responsible for maintaining a connection until the application programs are shut down. When data is received from the application layer, the TCP breaks it down into packets, numbering them and forwards it to the IP layer for delivery.

- **UDP:** Just like the TCP, the UDP sends short packets of data called datagrams, although it is a “best effort” protocol, which means there are no error recovery attempted. It is most commonly used for broadcasting and other low-latency connections between applications on the Internet.

Internet layer: The Internet layer is responsible for addressing and packaging. The data packed, is known as IP datagrams packets, and contains a source and destination address. These addresses are used to forward the datagrams between hosts across the internet. The core protocols of this layer are the IP, ARP, and ICMP.

- **Internet Protocol (IP):** Responsible for addressing, packaging, and forwarding of datagram packets.
- **Address Resolution Protocol (ARP):** Responsible for mapping IP addresses into MAC addresses.
- **Internet Control Message Protocol (ICMP):** Diagnostics and reporting errors in case of unsuccessful delivery of datagram packets.

Network Interface layer: The Network Interface layer is responsible for converting IP datagram packets into bits, and send to the network medium through either physical cable, fiber or wire. This layer operates on the physical hardware such as switches, hubs and bridges within LAN technologies including Ethernet.

- **Ethernet:** Most widely installed LAN technology that identifies the MAC addresses of the sender and receiver, and is responsible for the control of how data is transmitted over a LAN through a cable.

Ethernet

Summary

Ethernet is a technology developed in 1973 by Xerox, and is the name of the most widely installed local area network (LAN). It is also referred to as IEEE 802.3.
A LAN often covers small area like an office, building or a campus, and supplies networking capability to a group of computers over a wire.
Ethernet has been improved and evolved over time and can now deliver speeds of different gigabits per second. Most desktop and laptop computers come today with an integrated Ethernet card inside so that it is ready to be connected to a LAN. When Ethernet was developed, it was initially designed to run over coaxial cables but can now use special twisted pair cables or fiber optical cabling.[1]

[1] https://www.lifewire.com/what-is-ethernet-3426740

There are different Ethernet systems that can be installed in a network, depending on how big it is. Fast Ethernet was introduced in the mid-1990s as 100-BaseT2 or 100-BaseTX and could provide speeds of up to 100 Mbps. This high amount of speed was critically important for businesses and universities as the need for greater performance was high.[2]
In contrast with wired LANs, Ethernet standards of 802.11 technologies, also known as Wi-Fi standards, made connection to a LAN without the use of wired cables a possibility. Instead radio waves and 802.11 protocols are used to establish connection and communication. The speed rate of 802.11 standards are always under development and today they can provide speeds higher than 150 Mbps. [3]

Data Link and Physical Layer

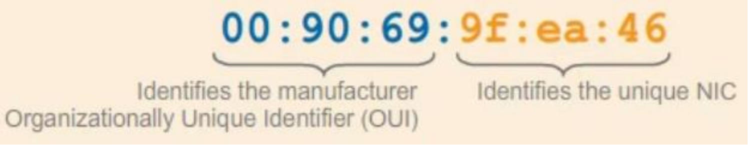
Ethernet operates at the first two layers of the OSI model - the Physical and Data link layer. The Data link is divided into two sub-layers known as the Logical Link Control (LLC) layer and the Medium Access Control (MAC) layer and is responsible for converting packets into frames. The Physical layer converts frames into signals or bit streams which are then transmitted across the media (a physical link such as wire).[4]

Data transmission and MAC addresses

In an Ethernet network, when a computer needs to send data, the NIC (Network Interface Controller), encapsulates it into an Ethernet frame. All the field in the ethernet frame are used to transmit the data to the correct receiver. A computer can either send data to one single device using unicast, multiple devices using multicast or broadcast if the data is intended for all the devices in the network.

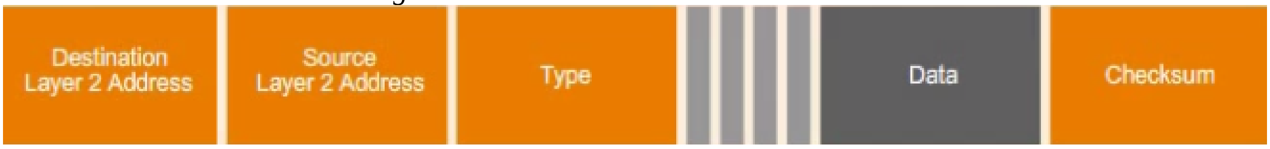
In Ethernet, layer 2 addresses also known as MAC (Media Access Control) addresses are used to identify the sender and receiver of data, that is transmitted into a network. This address is unique and burned into a PROM chip on the NIC by its manufacture. MAC addresses are made of hexadecimals with a length of 48-bits or 6-bytes.[5]
The first three octets or 24-bits in a MAC address uniquely identifies the manufacture of the NIC, and the last three octets or 24-bits identifies the NIC. NIC are purchased and assigned by the IEEE.[6]

Figure E-0: 48-bits MAC address


00:90:69:9f:ea:46
Identifies the manufacturer
Organizationally Unique Identifier (OUI)
Identifies the unique NIC

[2] https://www.lifewire.com/how-fast-is-ethernet-817549
[3] https://www.lifewire.com/wireless-standards-802-11a-802-11b-g-n-and-802-11ac-816553
[4] http://www.ccnablog.com/data-link-layer-ethernet-and-physical-layer/
[5] https://learningportal.juniper.net/juniper/user_activity_info.aspx?id=769
[6] http://aruljohn.com/mac.pl

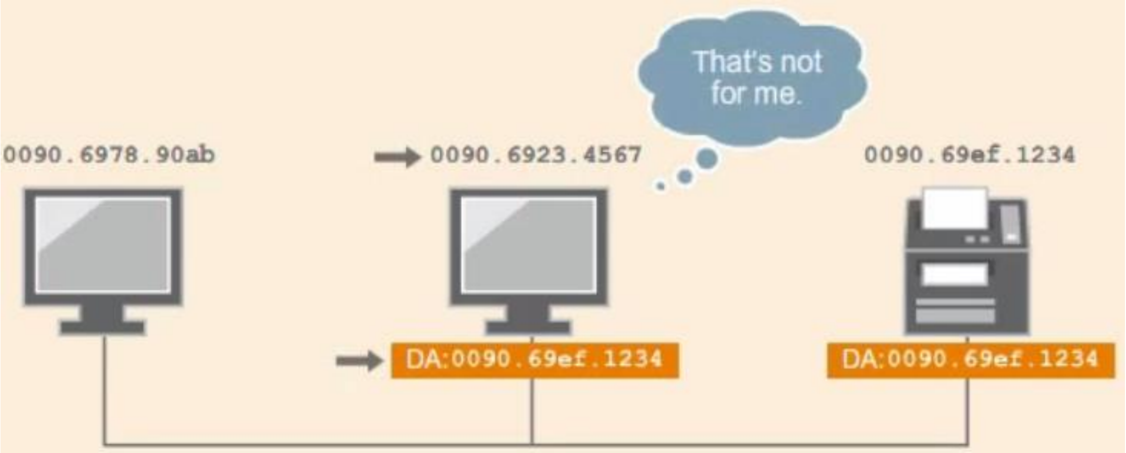
Figure E-1: Frame Format - Ethernet II Frame



- **Destination Layer 2 Address:** MAC address of the device receiving the data
- **Source Layer 2 Address:** MAC address of the device sending the data
- **Type:** Identifies the Layer 3 protocol such as IP or TCP or other protocols that is being transported in the Ethernet frame
- **Data:** Contains the original data that is being sent or received
- **Checksum:** This field includes a cyclick redundancy check (CRC), which is a method used to check for errors in the data that has been transmitted

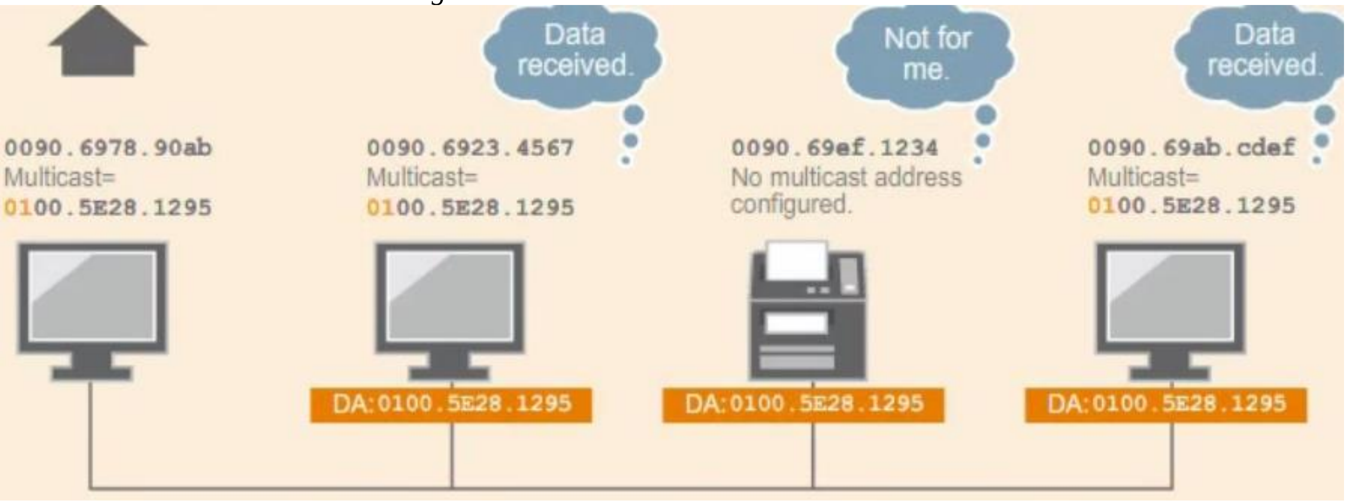
Unicast: Ethernet is a shared medium or bus which means that whenever a device sends data to another, all devices in the network will “receive” it. When the data is received, each device will examine the destination address to find out if the data is intended for it. If the destination address of is different, the data it will be discarded. Only the device with the same MAC address will accept the data.

Figure E-2: Unicast data transmission



Multicast: Multicast MAC addresses are used to send data to a group of devices on the same network. For the devices to receive multicast traffic, they must be configured to have the same first bit of the first byte.

Figure E-3: Multicast data transmission



Broadcast: If a device needs to send data to all other devices on the same network, it used a broadcast destination MAC address: ffff.ffff.ffff

Ethernet devices: Bridges, Switches, and Routers

Bridges: A layer 2 device that operates on the Data link layer of the OSI model and is used to "break" a large network into separate collision domains. This process is called Network Segmentation and can decrease collisions, and the number of devices competing for network access. Bridges only run half-duplex which means that devices on a LAN must allow a minimum idle period between transmission of Ethernet packets. MAC address tables are used to help control and limit traffic on a LAN. A bridge saves information on source and destination address to identify devices. If a device is not saved in the table, the bridge will flood the frame to all its ports. A bridge has only 2-4 ports.

Switches: Just like a bridge, a switch is a layer 2 device and can create a separate collision domain on each of its ports. In contrast with bridges, the switch is a full-duplex device and comes with few to hundreds of ports. With switches collisions are effectively eliminated and devices can freely send and receive data at the same time. Another benefit is that devices no longer compete for network access and switches allows to create vLANs.

Routers: Routers are small devices that operate on layer 3 of OSI model or layer 2 of the TCP/IP model. They are used to join multiple computer networks together via either wired or wireless connections. As mentioned before, IEEE 802.11 is a name for a series of protocols that are used for wireless networking. These protocols are often referred to as Wireless LAN, WLAN, or Wi-Fi and allows network devices on a LAN to communicate without the need of a wire or cable. [7]

[7] https://learningportal.juniper.net/juniper/user_activity_info.aspx?id=769

Virtual Local Area Network (VLAN)

Summary

Virtual LANs (VLANs) are logical groups of workstations, network devices and servers. VLANs are created on the software of a switch and allows network administrators to form separate broadcast domains for reasons such as scalability, security and the ease of network administration. When creating a VLAN, a logical bridge is placed between two LANs and this allows both the LANs to communicate over the bridge. There are multiple reasons to use this type of technology or method as VLANs in general improves the performance of a network.

Reasons to consider VLANs

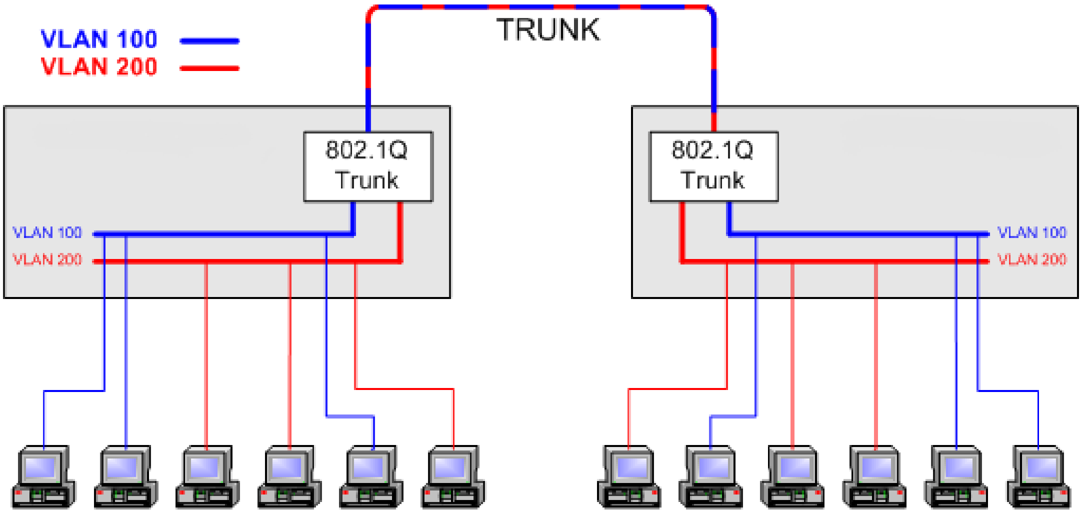
Some reasons and key benefits of implementing VLANs in a network are:

- When a LAN has more than 200 devices
- When a LAN has large amount of broadcast traffic
- Better performance because of decreased latency and traffic on the network
- Provides additional security for network communication
- Provides flexibility by creating multiple broadcast domains that enables administrators easier access and control.[8]

VLAN Connection: Trunking

When configuring a switch we can configure a port to run VLAN Trunking Protocol (VTP), to create a connection to another switch. The trunk uses a protocol called, 802.1Q that places tags on packets when they are transmitted across the connection between the two switches. These tags are used to identify which packet belongs to which vLAN.

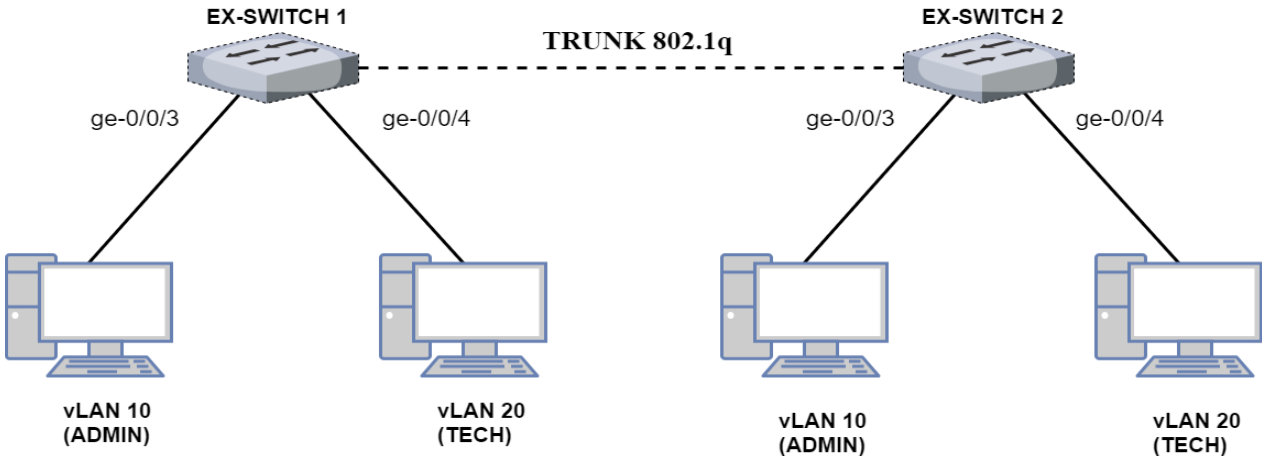
Figure V-1: This figure shows how the VLAN Trunk 802.1Q functions



[8] https://www.techopedia.com/definition/4804/virtual-local-area-network-vlan

In the diagram below we can see how both the switches are connected via a Trunk. By using the Trunk as the connectivity, both switches can now allow access from the same VLAN to the other. For example in this diagram VLAN 10 on EX-SWITCH 1, can communicate with VLAN 10 on EX-SWITCH 2. Same goes for VLAN 20.[9]

Figure V-2: Communication & 802.1q Trunk

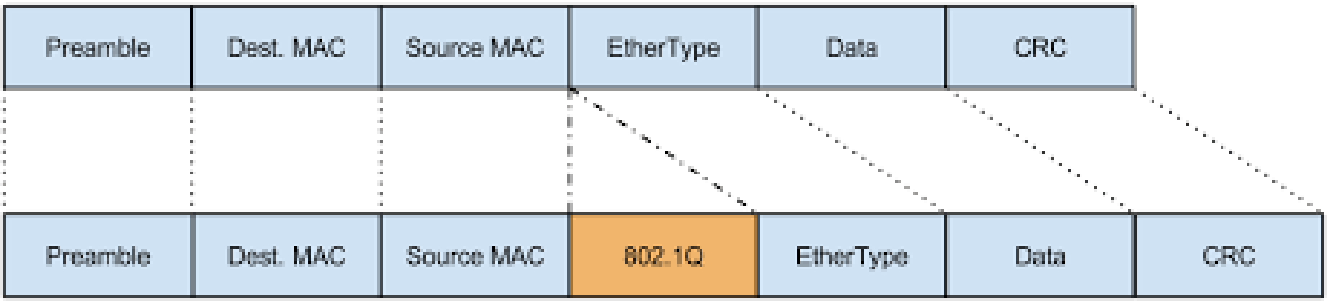


VLAN Tagging

VLAN Tagging allows the user to insert a VLAN-ID into a packet header to identify which VLAN that packet belongs to. In an Ethernet packet, when a computer within a VLAN generates a packet the 802.1Q, TPID (Tag Protocol Identifier) field is given a value of 0x8100. This value indicates that the packet is VLAN-tagged.

The packet also has a VLAN-ID field that uniquely has an 802.1q ID that identifies which VLAN the packet belongs to. For simple networks that only has a single VLAN, all packets include a default 802.1q tag. But this does not mark the packet as tagged.

Figure V-3: Inside the 802.1Q is the TPID and VLAN-ID

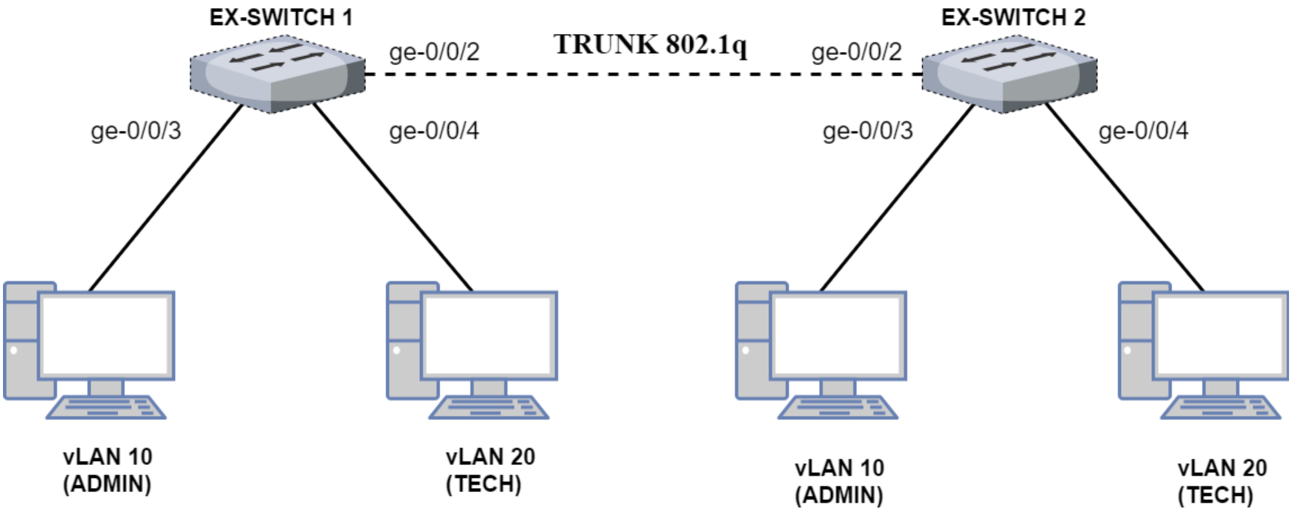


Preamble	Dest. MAC	Source MAC	EtherType	Data	CRC	
Preamble	Dest. MAC	Source MAC	802.1Q	EtherType	Data	CRC

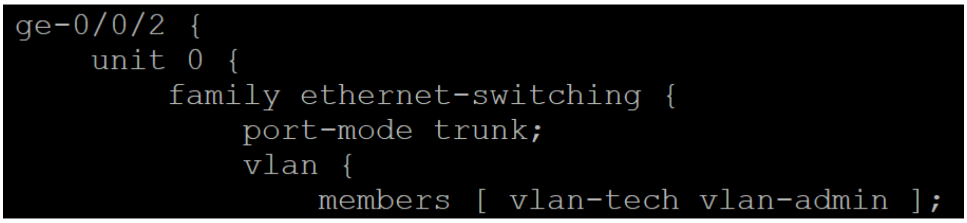
[9] http://computer.howstuffworks.com/lan-switch17.htm

VLAN Configuration:

VLAN Network Topology:

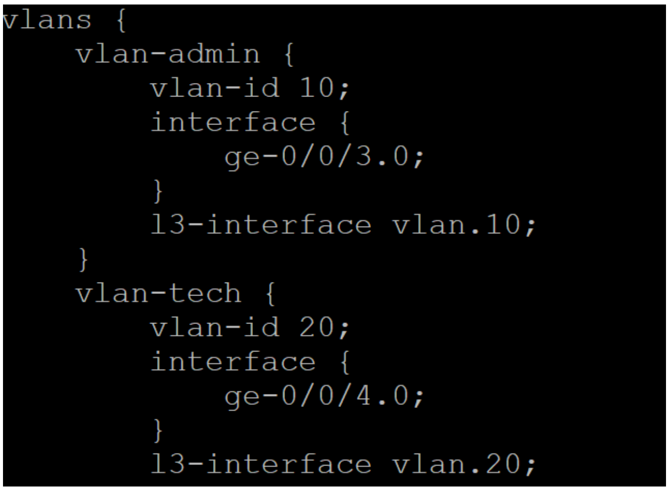


In this network topology, we have included two Juniper SRX routers to function as layer 3 switches. We have configured both routers to use their ge-0/0/2.0 to run on trunk mode. This is to allow connectivity between the two routers. using interface ge-0/0/2.0 as a trunk mode to make a connection between them.



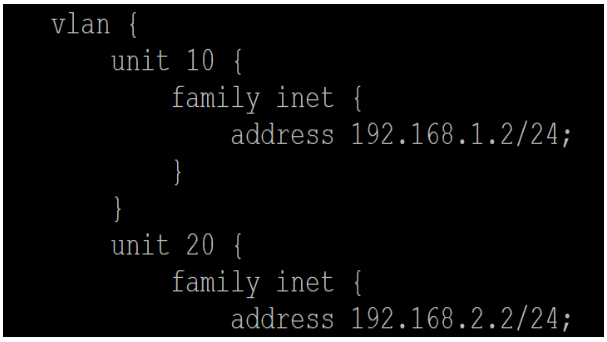
```
ge-0/0/2 {
    unit 0 {
        family ethernet-switching {
            port-mode trunk;
            vlan {
                members [ vlan-tech vlan-admin ];
```

We have configured VLAN admin with VLAN-ID and VLAN tech with VLAN-ID 20. We have attached admin to the interface ge-0/0/3, and tech to ge-0/0/4.0.



```
vlans {
    vlan-admin {
        vlan-id 10;
        interface {
            ge-0/0/3.0;
        }
        l3-interface vlan.10;
    }
    vlan-tech {
        vlan-id 20;
        interface {
            ge-0/0/4.0;
        }
        l3-interface vlan.20;
```

VLAN 10 (admin) us running on the network 192.168.1.2/24
VLAN 20 (tech) is running on the network 192.168.2.2/24



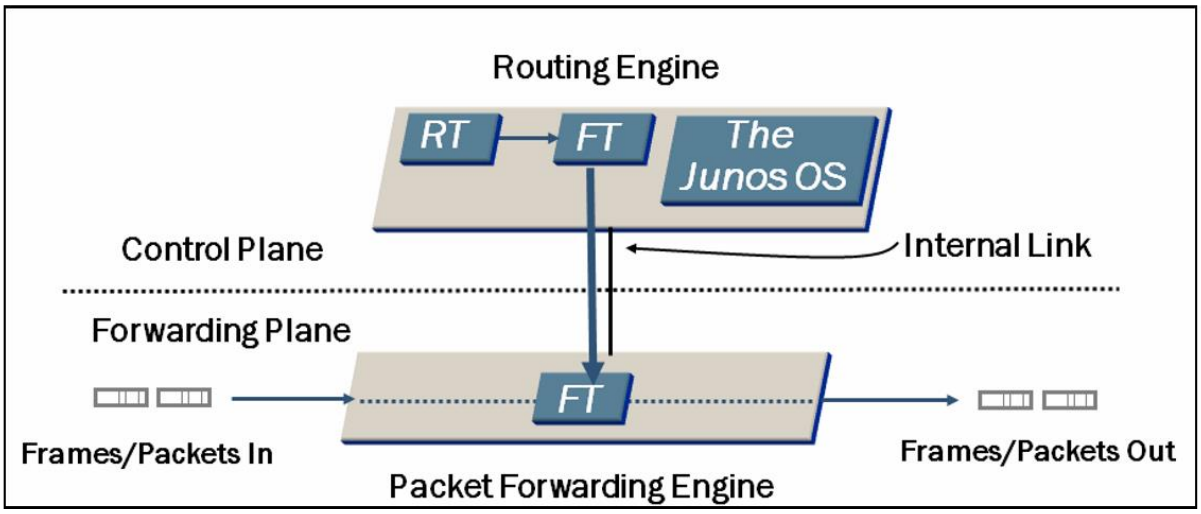
```
  vlan {
      unit 10 {
          family inet {
              address 192.168.1.2/24;
          }
      }
      unit 20 {
          family inet {
              address 192.168.2.2/24;
```

Because we did not have access to an EX-Switch, we decided to implement VLANs on a Juniper SRX router using another router as a routing-instance. By doing this, we could add the same VLANs on both routers.
The whole configuration of can be found in Appendix A.

Routing Engine and Packet Forwarding Engine

Summary

The Routing Engine (RE) is an essential part of a router that is used to control activities and build data in the Packet Forwarding Engine (PFE).



Routing Engine:
The Routing Engine (RE) is an essential part of a router that is used to control activities and build data in the Packet Forwarding Engine (PFE). All software processes that run on the Junos OS, are maintained by the RE. Processes such as maintaining the routing tables, managing the routing protocols, controlling of interfaces, controlling some of the chassis components, and provide interface for system management and user access to the router.

Packet Forwarding Engine:
The PFE is connected with the RE through an internal link, and is responsible for forwarding traffic. The PFE provides layer 2 and layer 3 packet switching, forwarding and route lookup functions. To ensure efficient movement of data, the PFE uses Application-Specific Integrated Circuits (ASICs). When RE builds up a routing table, a copy of the Forwarding Table (FT), is received by the PFE. When the PFE forwards traffic, it can either send traffic through a single port on a router or on all the ports, depending on whether the traffic is unicast or multicast.

Internet Protocol

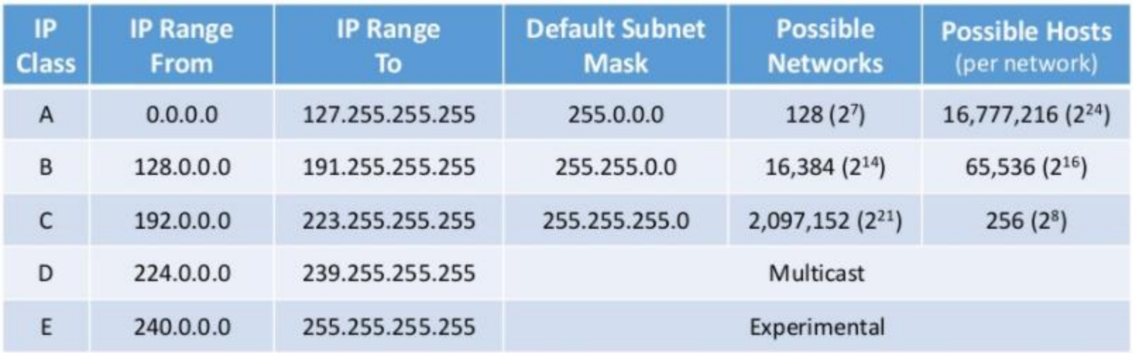
Summary

The Internet Protocol (IP) is a set of rules that by which data is sent from one host to another. Each host that is running on the internet has an IP address that is either an IPv4 or IPv6. The IP addresses uniquely identifies every hosts online such as a computer, a server, a router, and a phone. Along with addressing, one of the main functions of the IP protocol is that it consists of forwarding IP packets from source to destination. Since IP itself does not guarantee packet delivery, it is common to combine it with TCP, to work together in transmitting data over the internet. TCP will divide and bundle the packets it has received before forwarding them to IP, which then encapsulates these into IP packets.

Internet Protocol version 4 (IPv4)

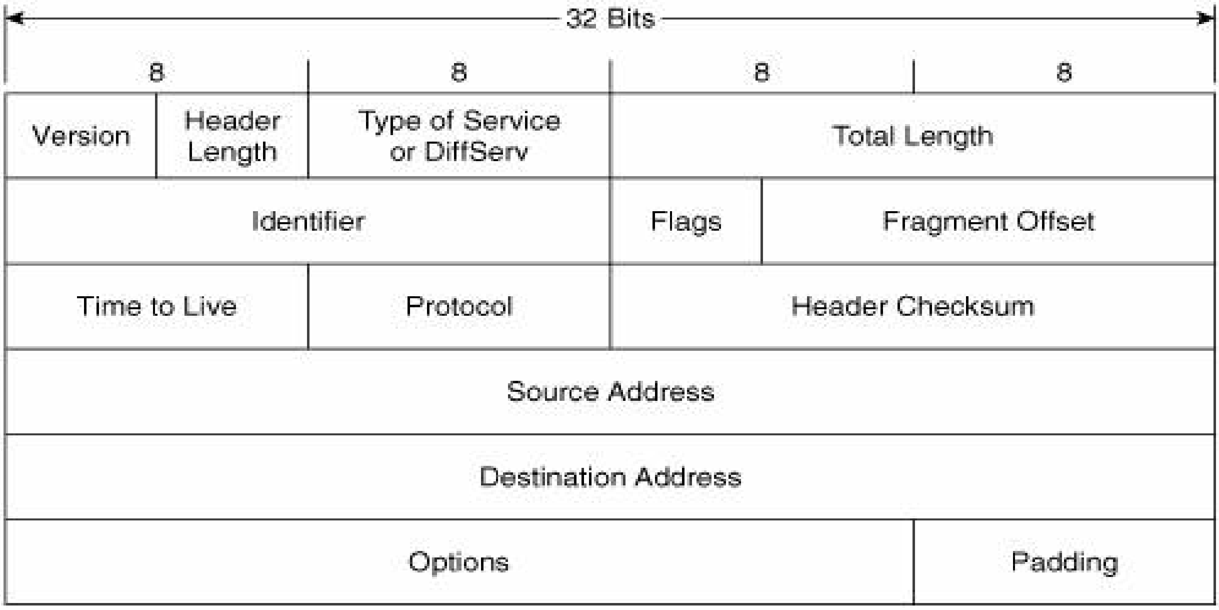
IPv4 was deployed in 1981 and comes with 32 bits (0’s and 1’s). It has four octets and is formatted into dotted decimal notation, for example: **192.149.252.76**. There are different class types of the addresses and some of them are reversed for private use. IPv4 provides approximately 4 billion addresses.

Figure I-1: This figure shows the different classes in IPv4.



IP Class	IP Range From	IP Range To	Default Subnet Mask	Possible Networks	Possible Hosts (per network)
A	0.0.0.0	127.255.255.255	255.0.0.0	128 (2^7)	16,777,216 (2^{24})
B	128.0.0.0	191.255.255.255	255.255.0.0	16,384 (2^{14})	65,536 (2^{16})
C	192.0.0.0	223.255.255.255	255.255.255.0	2,097,152 (2^{21})	256 (2^8)
D	224.0.0.0	239.255.255.255	Multicast		
E	240.0.0.0	255.255.255.255	Experimental		

IPv4 Header



<table>
<tr><td colspan="8">32 Bits</td></tr>
<tr><td colspan="2">8</td><td colspan="2">8</td><td colspan="2">8</td><td colspan="2">8</td></tr>
<tr><td>Version</td><td>Header Length</td><td colspan="2">Type of Service or DiffServ</td><td colspan="4">Total Length</td></tr>
<tr><td colspan="4">Identifier</td><td>Flags</td><td colspan="3">Fragment Offset</td></tr>
<tr><td colspan="2">Time to Live</td><td colspan="2">Protocol</td><td colspan="4">Header Checksum</td></tr>
<tr><td colspan="8">Source Address</td></tr>
<tr><td colspan="8">Destination Address</td></tr>
<tr><td colspan="6">Options</td><td colspan="2">Padding</td></tr>
</table>

An IPv4 packet header contains different relevant information about the packet. The packet header are 32 bits long and contains 20 bytes of data. Here is a description of what all the fields are:

- **Version:** Version number of the Internet Protocol used (e.g. IPv4).
- **IHL (Internet Header Length):** This field include the length of the IP header.
- **DSCP (Differentiated Services Code Point):** This is the Type of Service (ToS).
- **ECN (Explicit Congestion Notification):** Carries information about the congestion seen in the route.
- **Total Length:** Length of the IP Packet.
- **Identification:** If a packet during transmission is fragmented, all the fragments contains the same ID number. This ID number identifies the IP packets origin.
- **Flags:** This field find out if an IP packet that is too large and if it can be fragmented or not.
- **Fragment Offset:** Describes the exact position of the fragment in the original IP packet.
- **Time to Live:** Used to avoid looping. Every packet that is transmitted in a network has a TTL value set. Every hop, the TTL decrements this value, and when the value reaches 0 the packet is dropped.
- **Protocol:** Tells the destination host which protocol a packet belongs to (i.e. ICMP, TCP, UDP or OSPF).
- **Checksum:** This field check if the packet received is error-free.
- **Source Address:** Contains a 32-bit address of the source (sender of the packet).
- **Destination Address:** Contains a 32-bit address of the receiver (destination of the packet).
- **Options:** Optional field used for Security, Time Stamps etc... This field is only used if IHL (length of the IP packet) is greater than 5.
- **Padding:** This field is used to ensure that the internet header ends on a 32-bit boundary.[10]

[10] https://www.tutorialspoint.com/ipv4/ipv4_packet_structure.htm

Internet Protocol version 6 (IPv6)

IPv6 was deployed in 1999, and is in line to replace IPv4 in the future as IPv4 addresses are running out. IPv6 addresses comes with 128 bits and are formatted into a hexadecimal notation, consisting of numbers and letters. Each block out of the eight blocks in an IPv6 address represents 16 bits of a total 128 bits. These addresses can be expressed with fewer letters by suppressing leading and consecutive zeros. Most of the up-to-date devices which use IP protocols has included IPv6 and they only need activation. For the older devices, a software update is required.
Some of the benefits that comes with IPv6 over IPv4, is that it provides better and more efficient routing, multicast routing, and has a simpler fixed header format that is easier for administration. For many years industries have been stating that IPv4 addresses are running out and the situation becomes urgent each year, yet most companies fully operate on running IPv4 and new devices and services are added to their network all the time. In addition to much larger network address spaces that comes with IPv6, it also provides feature improvements over IPv4.

Some Notable Features in IPv6:

- **IPSec**: This is a key enhancement in IPv6. Authentication Header (AH) and Encapsulation Security Payload (ESP) are no longer treated as an upper layer protocol. IPv6 provides secure communication between endpoints by building a secure link without the use of intermediate VPN gateways.

- **Stateless Auto-configuration**: This is used as a new method to assign addresses. Endpoints in a network dynamically assigns itself with an IP address from route advertisements announced via the connected link. Stateless Auto-configuration is by default enabled on all operating systems running the IPv6 protocol stack so that the users do not need to manually or via DHCP assign addresses to their endpoints.

- **Flow-based QoS/CoS**: This field was added to allow routers on a IPv6 network to faster identify and provide QoS treatment of packets as they are forwarded to the downstream nodes, in the network.[11]

IPv6 Address Notation

Compared to IPv4 with its 32-bits addresses, IPv6 has 128 bits consisting of letters and numbers (hexadecimal characters). Because of their long 128 bits they tend to have many zeros and to simplify this address there are a few methods that can be used. The first abbreviation is to suppress leading zeroes, so:

This table shows how a normal 128-bit IPv6 address can be suppressed.

Normal IPv6 address	Suppressed leading zeroes
2001:0DB8:0000:0000:0202:B3FF:FE1E:8329	**2001:DB8:0:0:202:B3FF:FE1E:8329**

It is also possible to replace consecutive zeros within the address with a double colon such as:

2001:0DB8:0000:0000:0202:B3FF:FE1E:8329	**2001:DB8::202:B3FF:FE1E:8329**

[11] https://fronter.com/eal/links/files.phtml/2080432588$596013739$/2nd+Semester/Project+Network/IPv6/IPV6+Networking+Fundamentals.pdf

The double colon can only appear once in the address because the computer always uses a 128-bit binary representation of the address. When a computer finds a double colon in an IPv6 address, it expands it with as many leading zeros as need to get 128-bits.

Address Types in IPv6

There are three types of addressing in IPv6: Unicast, Multicast and Anycast.
Broadcast has been removed because multicast addresses are used instead.

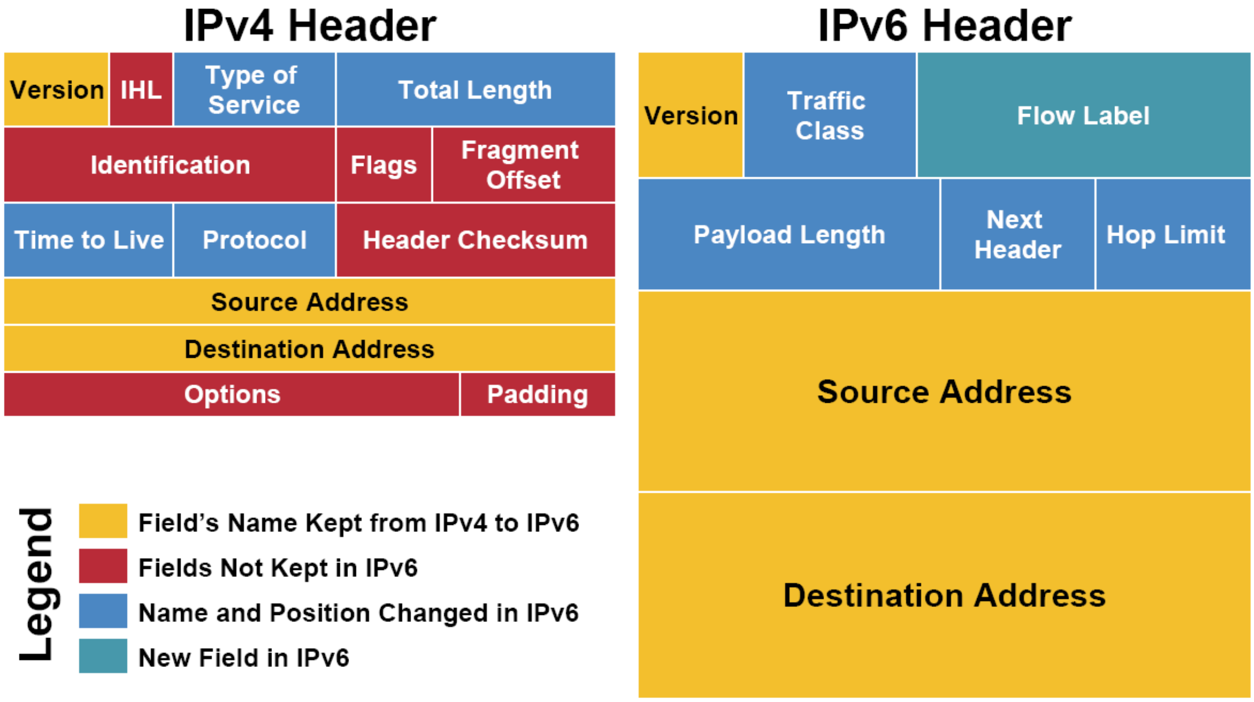
- **Unicast:** An identifier for an IPv6 node. Packets sent to a unicast address are delivered to the interface identified by that address.

- **Multicast:** An identifier for a group of interfaces belonging to different IPv6 nodes. A packet sent to a multicast address is delivered to all interfaces identified by that address.

- **Anycast:** An identifier for a group of interfaces belonging to different IPv6 nodes. A packet sent to an anycast address is delivered to only one of these interfaces and usually the closest one.

IPv6 Header

The IPv6 header has been simplified to have fewer fields. Below, figure I-2, shows which fields have been removed, added, kept, and changed. The fields that have been removed. These changes made to the IPv6 means that IPv6 is easier, faster, and more efficient to process packets. IPv6 was not only built because of the address range it provides, but also for better performance and routing efficiency.

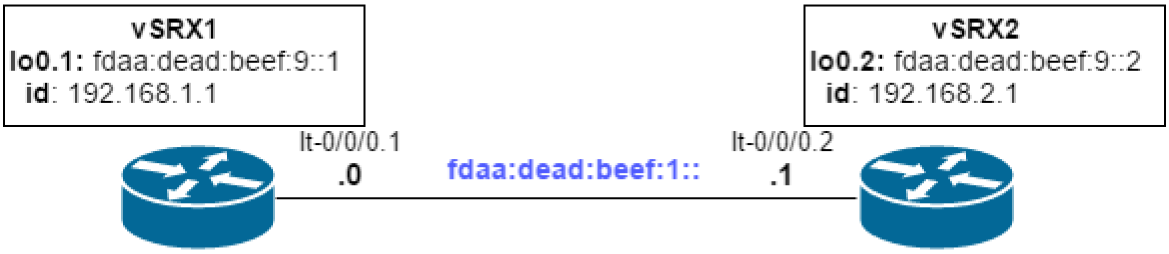


IPv6 Header

- **Source address (128 bits):** This field contains the IPv6 address of the originating node of the packet.
- **Destination address (128 bits):** This field contains the IPv6 address of the recipient node of the packet.
- **Version / IP version (4 bits):** Indicates the version of IPv6 protocol
- **Traffic Class/Packet priority (8 bits):** Indicates the class or priority of the IPv6 packet. This fields looks at the values in the ToS (Type of Service) field of the packet to know what services should be provided.
- **Flow Label/ QoS management (20 bits):** This label is used for latency-sensitive traffic such as Voice over IP (VoIP) or multimedia and used to identify in which communication a packet belongs to avoid re-ordering of the same packet.
- **Payload Length (16 bits):** This field tells IPv6 routers how much information a packet contains in its payload.
- **Next Header (8 bits):** This field indicates type of Extension Header or the upper layer PDUs such as TCP or UDP.
- **Hop Limit/TTL (8bits):** This field is used to prevent loops in a network. The value of Hop Limit is decremented by 1 after every hop. When the limit becomes 0, the packet will be discarded.[12]

IPv6 Configuration

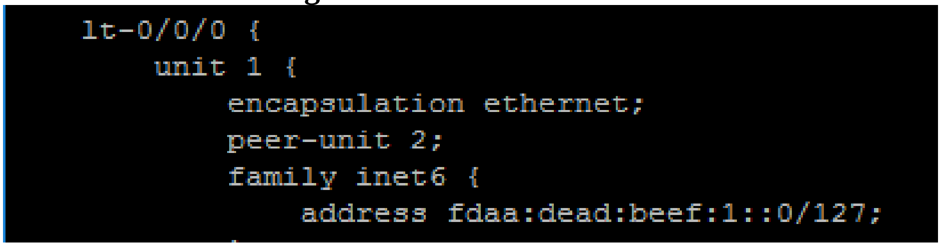
Network Topology



In this network topology, we have configured two routing-instances in a physical Juniper SRX router (SRX-A4). Our routing instances are vSRX1 and vSRX2. We have made a logical tunnel connection and peered both routers. Each routing-instance is given its own unique IPv6 address.

[12] http://www.tcpipguide.com/free/t_IPv6DatagramMainHeaderFormat.htm

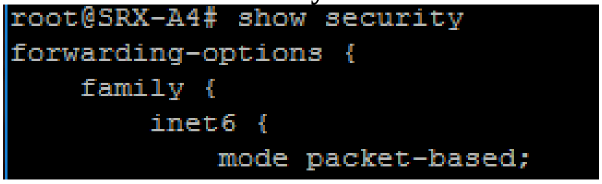
SRX-A4# show routing-instances vSRX1



```
    lt-0/0/0 {
        unit 1 {
            encapsulation ethernet;
            peer-unit 2;
            family inet6 {
                address fdaa:dead:beef:1::0/127;
```

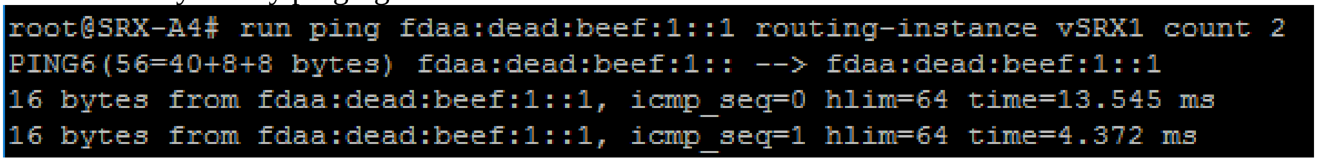
In security options under forwarding-options, we have configured inet6, mode packet-based on the physical router. This is done to describe how the packets are processed.

SRX-A4# show security



```
root@SRX-A4# show security
forwarding-options {
    family {
        inet6 {
            mode packet-based;
```

We can finally test by pinging from **vSRX-1** to **vSRX-2**.



```
root@SRX-A4# run ping fdaa:dead:beef:1::1 routing-instance vSRX1 count 2
PING6(56=40+8+8 bytes) fdaa:dead:beef:1:: --> fdaa:dead:beef:1::1
16 bytes from fdaa:dead:beef:1::1, icmp_seq=0 hlim=64 time=13.545 ms
16 bytes from fdaa:dead:beef:1::1, icmp_seq=1 hlim=64 time=4.372 ms
```

The whole configuration of IPv6 can be found in Appendix B.

Open Shortest Path First (OSPF)

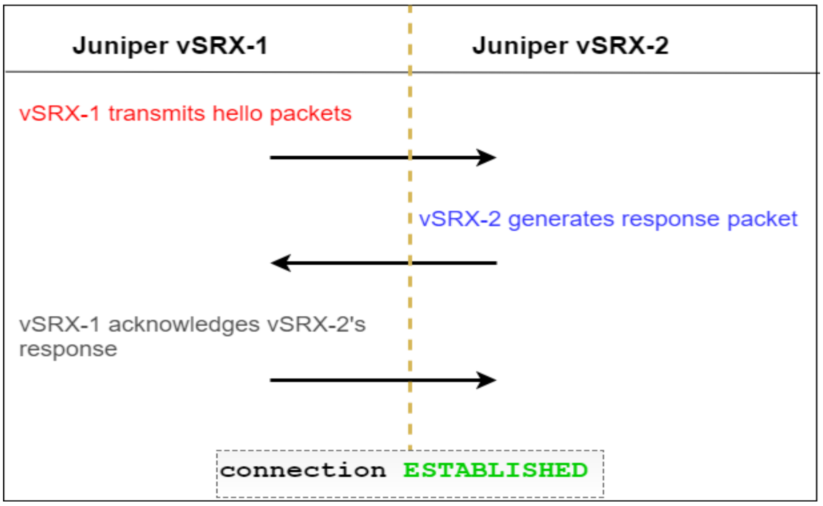
Summary

OSPF (Open Shortest Path First) is an Interior Gateway Protocol (IGP), and used to distribute IP routing information throughout a single Autonomous System (AS). The OSPF uses link-state information to make routing decisions using SPF (Shortest Path First) algorithm also referred to as Dijkstra's algorithm. Routers running OSPF floods Link State Advertisements (LSAs) throughout the network in the autonomous system to gain information about other routers local state such as usable interfaces and reachable neighbours. Through LSAs the routers also get a complete picture of the network topology and this allows them to calculate end-to-end paths within the autonomous system and select the shortest path to reach a destination. When a component in the network fails the routers update their routing table to easily recalculate and choose a different destination path. Routers running OSPF sends hello packets (IIHs), to discover new neighbours to establish adjacencies, which means that they will build up an identical database that describes the network topology. Despite OSPF being one of the most preferred routing protocol, the IGPs have their disadvantages. When the number of routers are increased in the network, the time of updating the network topology increases. Recalculation also becomes a time issue as all routers need to update their routing table. For these reasons IGPs such as OSPF are considered unsuitable for routing across large networks and are only used to route traffic within single autonomous systems. [13]

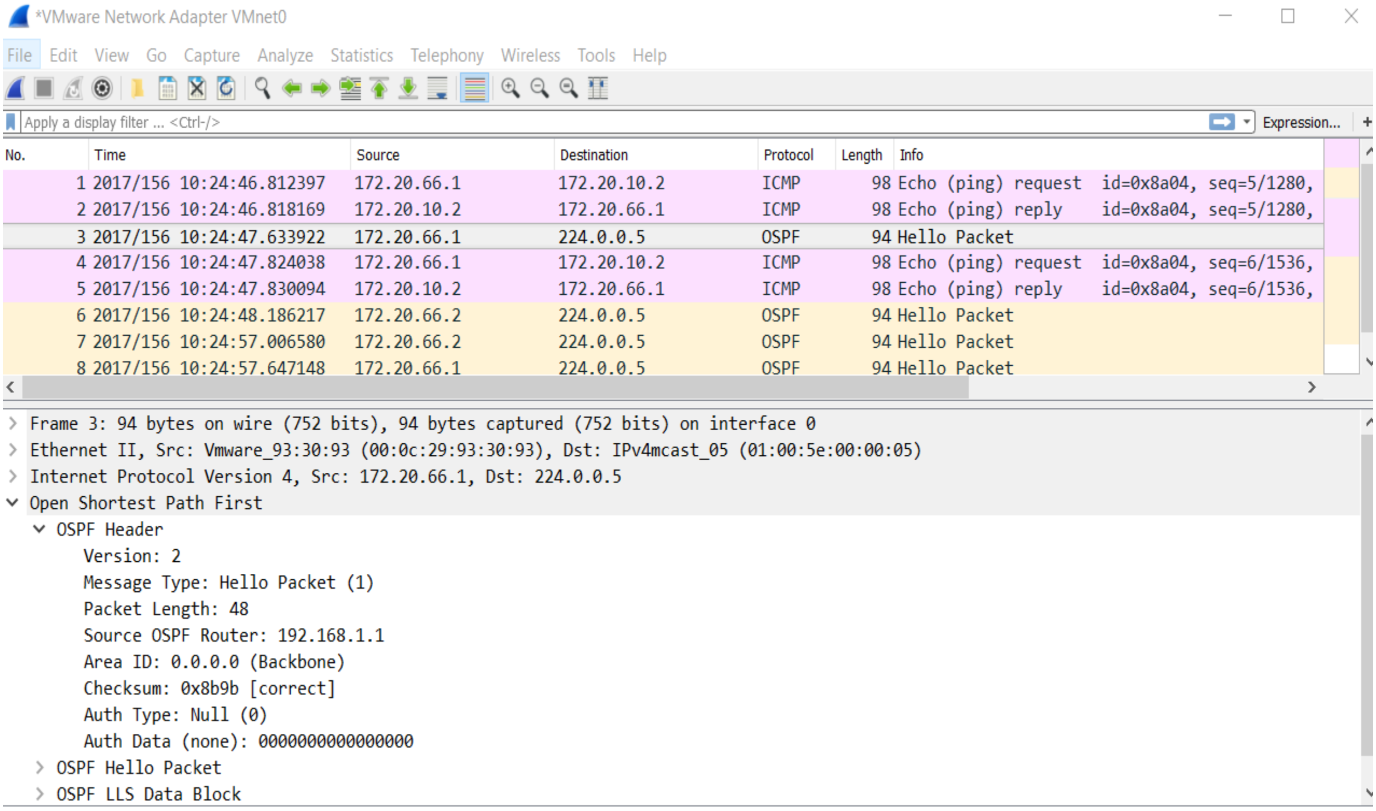
[13] http://www.metaswitch.com/resources/what-is-open-shortest-path-first-ospf

OSPF Three-Way Handshake

The Three-Way Handshake also known as the TCP three-way is a three-step method that requires two network nodes to exchange SYN(synchronize) and ACK (acknowledgement) packets before actual data communication begins. In an OSPF network, routers must periodically send multicast hello packets to all its enabled interfaces to establish and maintain neighbour relationships.


Juniper vSRX-1
Juniper vSRX-2
vSRX-1 transmits hello packets
vSRX-2 generates response packet
vSRX-1 acknowledges vSRX-2's response
connection ESTABLISHED

By Wireshark we can capture and monitor traffic on the interfaces of OSPF routers. Here we have used Wireshark which is a monitoring tool, to capture OSPF hello packets.

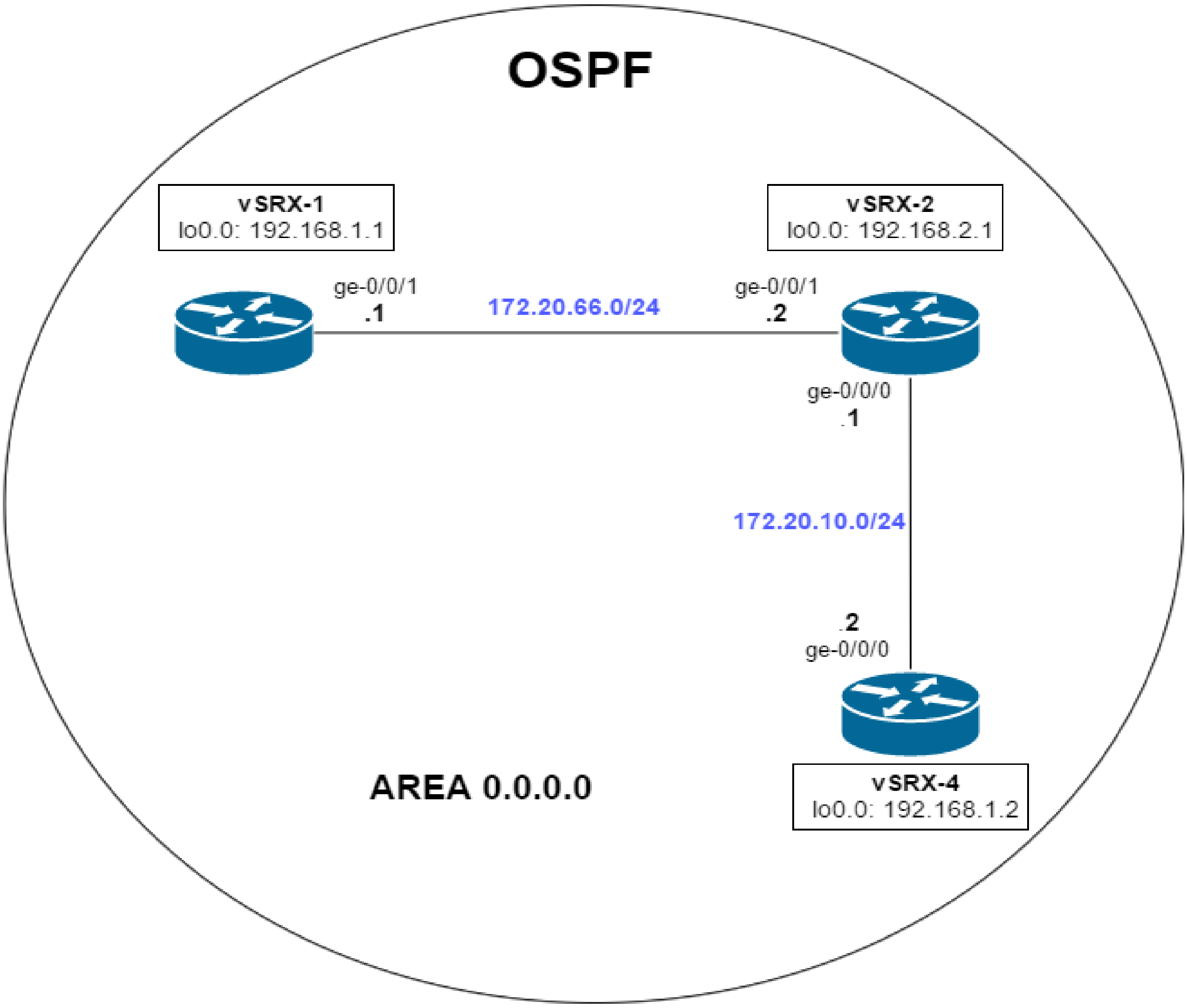

*VMware Network Adapter VMnet0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+

```
No.  Time                        Source        Destination   Protocol  Length  Info
   1 2017/156 10:24:46.812397    172.20.66.1   172.20.10.2   ICMP        98 Echo (ping) request  id=0x8a04, seq=5/1280,
   2 2017/156 10:24:46.818169    172.20.10.2   172.20.66.1   ICMP        98 Echo (ping) reply    id=0x8a04, seq=5/1280,
   3 2017/156 10:24:47.633922    172.20.66.1   224.0.0.5     OSPF        94 Hello Packet
   4 2017/156 10:24:47.824038    172.20.66.1   172.20.10.2   ICMP        98 Echo (ping) request  id=0x8a04, seq=6/1536,
   5 2017/156 10:24:47.830094    172.20.10.2   172.20.66.1   ICMP        98 Echo (ping) reply    id=0x8a04, seq=6/1536,
   6 2017/156 10:24:48.186217    172.20.66.2   224.0.0.5     OSPF        94 Hello Packet
   7 2017/156 10:24:57.006580    172.20.66.2   224.0.0.5     OSPF        94 Hello Packet
   8 2017/156 10:24:57.647148    172.20.66.1   224.0.0.5     OSPF        94 Hello Packet
```

```
> Frame 3: 94 bytes on wire (752 bits), 94 bytes captured (752 bits) on interface 0
> Ethernet II, Src: Vmware_93:30:93 (00:0c:29:93:30:93), Dst: IPv4mcast_05 (01:00:5e:00:00:05)
> Internet Protocol Version 4, Src: 172.20.66.1, Dst: 224.0.0.5
v Open Shortest Path First
  v OSPF Header
      Version: 2
      Message Type: Hello Packet (1)
      Packet Length: 48
      Source OSPF Router: 192.168.1.1
      Area ID: 0.0.0.0 (Backbone)
      Checksum: 0x8b9b [correct]
      Auth Type: Null (0)
      Auth Data (none): 0000000000000000
  > OSPF Hello Packet
  > OSPF LLS Data Block
```

OSPF Configuration

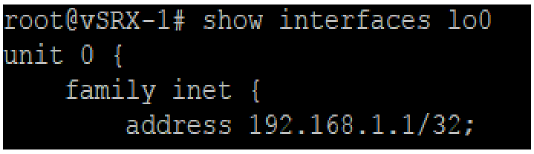
OSPF network topology



In this network topology, we have configured three virtual SRX routers to run OSPF in the same area. The main goal here is to allow **vSRX-1** to communicate with **vSRX-4** using OSPF only. We have tested this by disabling static routes and make sure that only OSPF is running on all the routers.

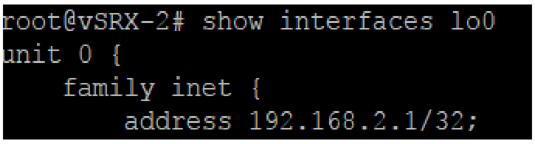
First, we have configured the loopback addresses on all the three routers.

vSRX-1# show interfaces lo0



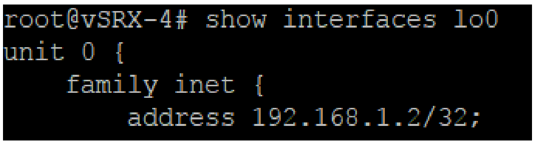
```
root@vSRX-1# show interfaces lo0
unit 0 {
    family inet {
        address 192.168.1.1/32;
```

vSRX-2# show interfaces lo0



```
root@vSRX-2# show interfaces lo0
unit 0 {
    family inet {
        address 192.168.2.1/32;
```

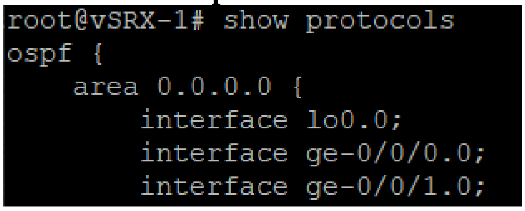
vSRX-4# show interfaces lo0



```
root@vSRX-4# show interfaces lo0
unit 0 {
    family inet {
        address 192.168.1.2/32;
```

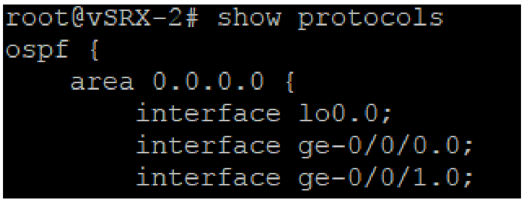
We have configured OSPF protocols by connecting all the routers to the same backbone area**,** area 0.0.0.0.

vSRX-1# show protocols



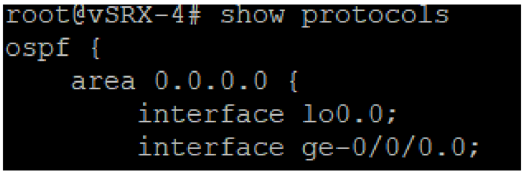
```
root@vSRX-1# show protocols
ospf {
    area 0.0.0.0 {
        interface lo0.0;
        interface ge-0/0/0.0;
        interface ge-0/0/1.0;
```

vSRX-2# show protocols



```
root@vSRX-2# show protocols
ospf {
    area 0.0.0.0 {
        interface lo0.0;
        interface ge-0/0/0.0;
        interface ge-0/0/1.0;
```

vSRX-4# show protocols

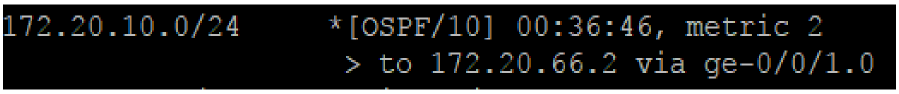


```
root@vSRX-4# show protocols
ospf {
    area 0.0.0.0 {
        interface lo0.0;
        interface ge-0/0/0.0;
```

OSPF Testing

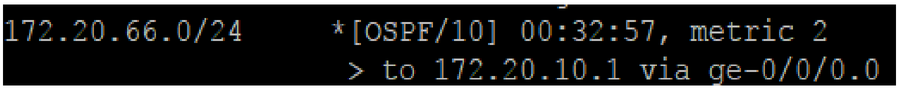
To verify that OSPF is active and running on the routers, we can in edit mode type '*run show route*' in the terminal. This will show us the routing table of the two routers.

vSRX-1# run show route



```
172.20.10.0/24      *[OSPF/10] 00:36:46, metric 2
                     > to 172.20.66.2 via ge-0/0/1.0
```

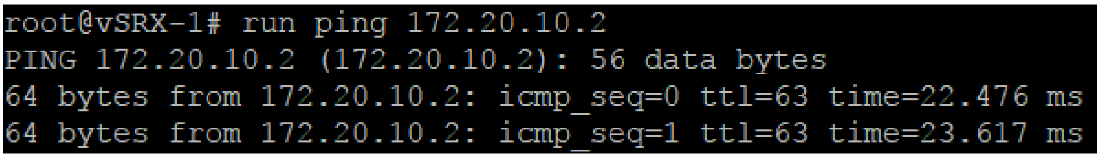
vSRX-4# run show route



```
172.20.66.0/24      *[OSPF/10] 00:32:57, metric 2
                     > to 172.20.10.1 via ge-0/0/0.0
```

Now that both routers know the routes to communicate we can start testing. On **vSRX-1** we will ping **vSRX-4** to verify communication.

vSRX-1# run ping 172.20.10.2

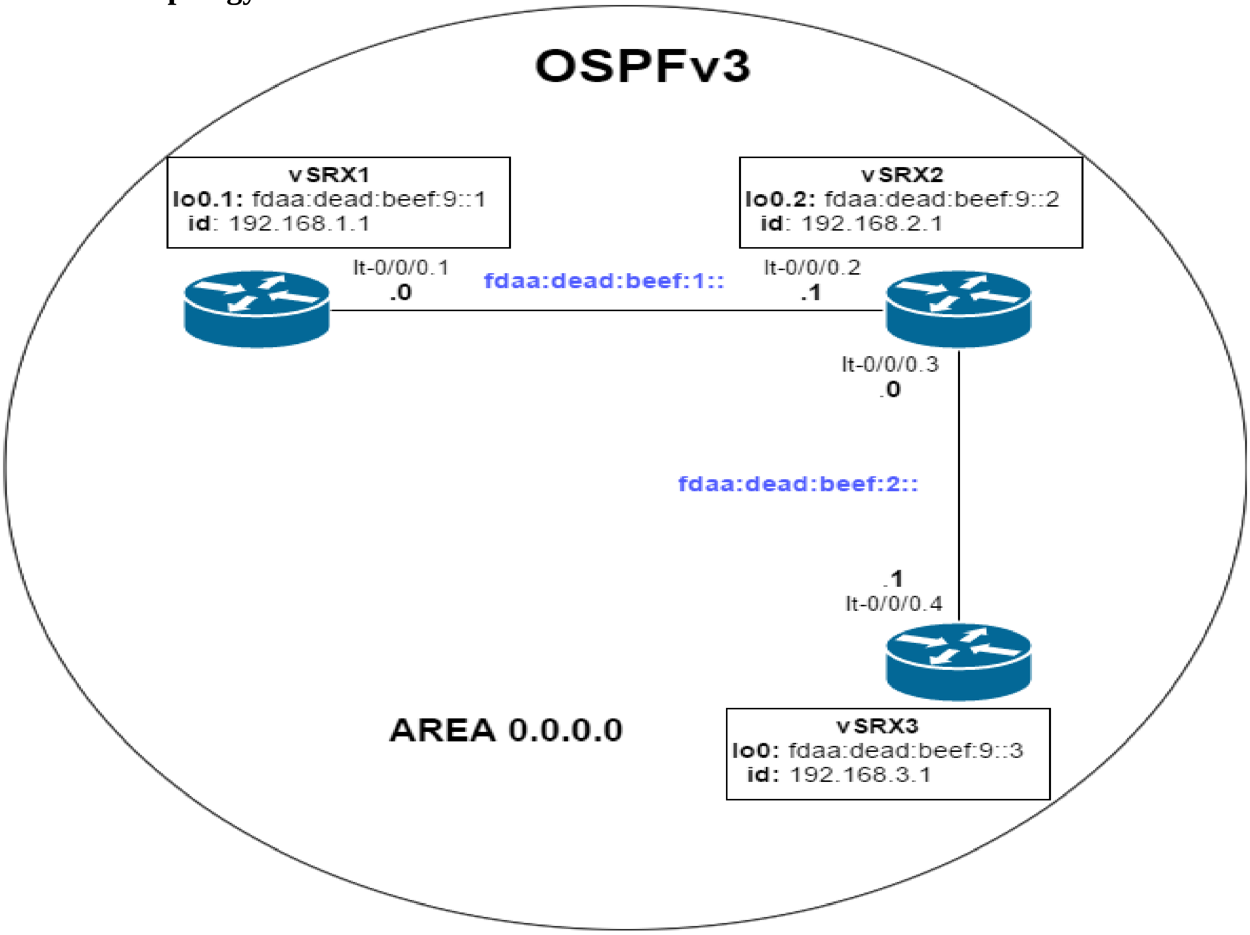


```
root@vSRX-1# run ping 172.20.10.2
PING 172.20.10.2 (172.20.10.2): 56 data bytes
64 bytes from 172.20.10.2: icmp_seq=0 ttl=63 time=22.476 ms
64 bytes from 172.20.10.2: icmp_seq=1 ttl=63 time=23.617 ms
```

We can verify that we have OSPF up and running and both routers can communicate. In this network topology we had 3 routers, but if we were to add more the configuration would be the same. Each router running OSPF must be connected to the same backbone area and each must have an IP-address on the interface that identifies it.
The whole configuration of OSPF can be found in Appendix C.

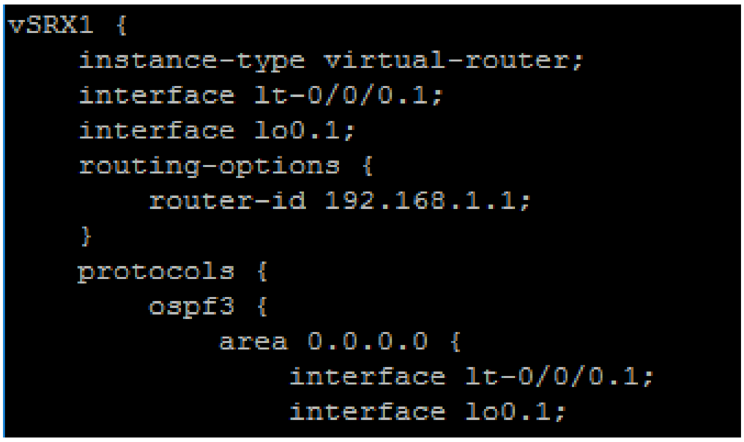
IPv6 Configuration - OSPFv3

Network Topology



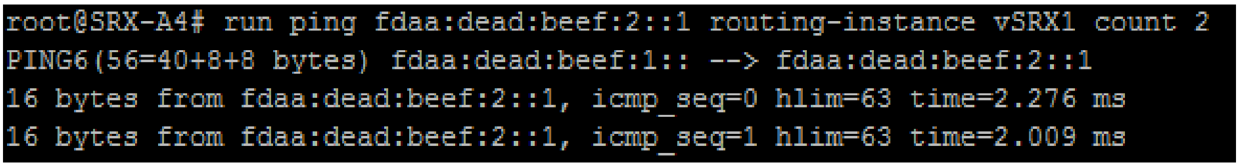
OSPFv3 is specifically created for IPv6 networks and works exactly like OSPFv2.
In this network topology, we have configured three routing-instances, vSRX1, vSRX2 and vSRX3, inside a physical Juniper SRX router (SRX-A4). The goal here is to establish communication between vSRX1 and vSRX3 using OSPF.

We have setup OSPFv3 protocol inside all three routing-instances, **vSRX1**, **vSRX2**, and **vSRX3**.
The Router-ID is given each router and used to specifically identify each of them.



```
vSRX1 {
    instance-type virtual-router;
    interface lt-0/0/0.1;
    interface lo0.1;
    routing-options {
        router-id 192.168.1.1;
    }
    protocols {
        ospf3 {
            area 0.0.0.0 {
                interface lt-0/0/0.1;
                interface lo0.1;
```

We can now test if both routers can communicate.
On **vSRX-1** we are pinging **vSRX-3**.



```
root@SRX-A4# run ping fdaa:dead:beef:2::1 routing-instance vSRX1 count 2
PING6(56=40+8+8 bytes) fdaa:dead:beef:1:: --> fdaa:dead:beef:2::1
16 bytes from fdaa:dead:beef:2::1, icmp_seq=0 hlim=63 time=2.276 ms
16 bytes from fdaa:dead:beef:2::1, icmp_seq=1 hlim=63 time=2.009 ms
```

As the screenshot shows, vSRX1 can ping vSRX3 without problems. We verified our work by testing that all the three routers can ping each other, and they could.
The whole configuration of OSPFv3 can be found in Appendix B.

Intermediate System to Intermediate System (IS-IS)

Summary

IS-IS (Intermediate System – Intermediate System) is an Interior Gateway Protocol in family of IP routing protocols. IS-IS is primarily used by ISPs to distribute routing information throughout a single autonomous system in an IP network. Just like OSPF, IS-IS is a link-state protocol, which means that routers exchange network topology information with their nearest neighbours. The link-state protocol uses SPF (Shortest Path First) algorithm also referred to as Dijkstra's algorithm, to determine route to reach each destination in a network. The main advantage of using a link-state protocol is that the routers can gain complete network topology information and whenever changes occur, the protocol recalculates routes and provides support for multiple paths. However, the disadvantages are, when an increasing number of routers are added in the same network, the size, frequency of topology updates and the length of time it takes to calculate end-to-end routes

increases. The lack of scalability means that IS-IS is preferably used in only one single autonomous system, to maintain a reliable network. Some of the operations of IS-IS is that each router distributes information about its local state such as usable interface, reachable neighbours, and the cost of using each interface. This operation is done by sending link-state PDUs. Just like OSPF, routers send hello packets to establish adjacencies with other routers and build up an identical database that describes the network topology in an autonomous system.
When configuring multiple IS-IS routers it is possible to "break" an autonomous system into multiple areas. This operation requires that each router either runs on level 1 or level 2 which indicates if it is a router that is operating in one area, or across multiple areas. Level 2 is disabled on a router if it is operating in only one single area whereas level 1 is disabled if the router is operating across multiple areas.[14] [15]

ISO Network Addresses

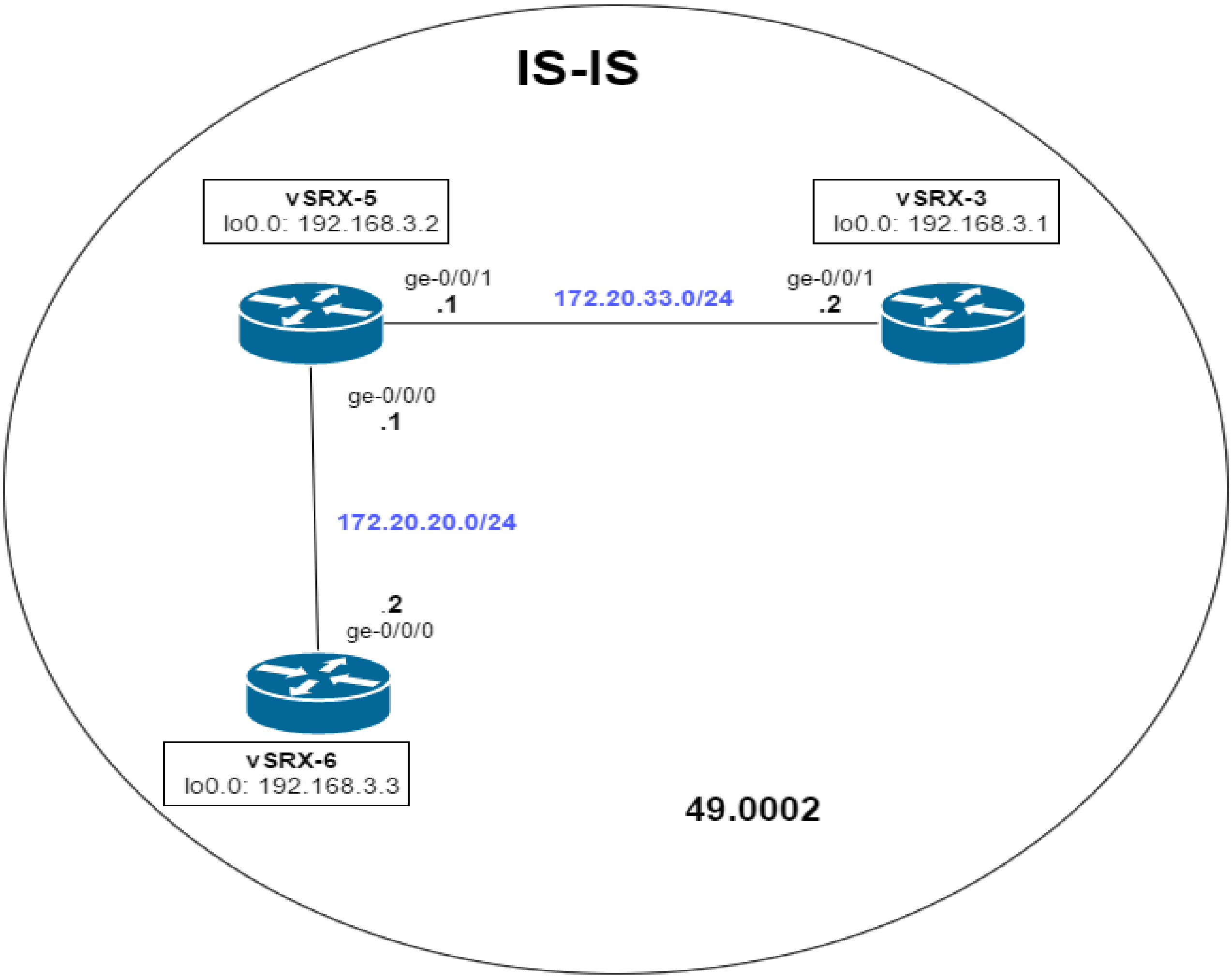
For a router to support IS-IS it is a requirement that an ISO network entity title (NET) address is configured on the routers interface. A unique ISO NET address identifies each router in a network and consist of different parts. Here is an example of an ISO NET address:
49.0001.1921.6800.1001.00

- 49 - **AFI** (The start value of 49, are considered private addresses and indicates the area number)
- 0001 – **ID** (This is the area ID)
- 1921.6800.1001 – **System identifier**
- 00 - **Selector**

[14] https://www.juniper.net/documentation/en_US/junos12.3/topics/concept/is-is-routing-overview.html
[15] http://www.metaswitch.com/resources/what-is-intermediate-system-to-intermediate-system-isis

IS-IS Configuration

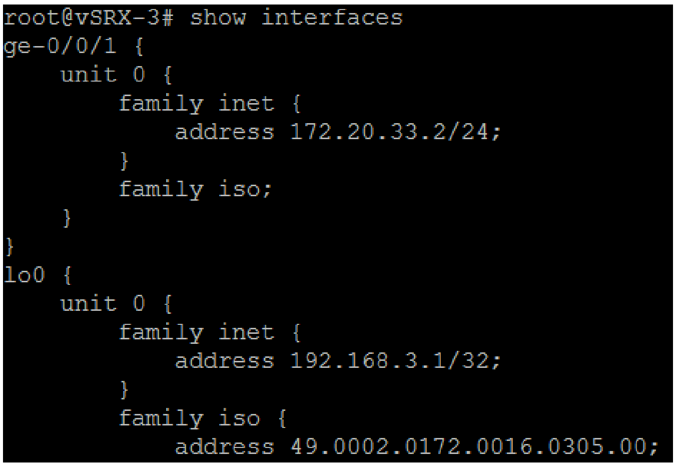
OSPF network topology



In this network topology, we have configured three virtual SRX routers in the same area, area 49. The goal is to allow **vSRX-3** to communicate with **vSRX-6** using IS-IS only.

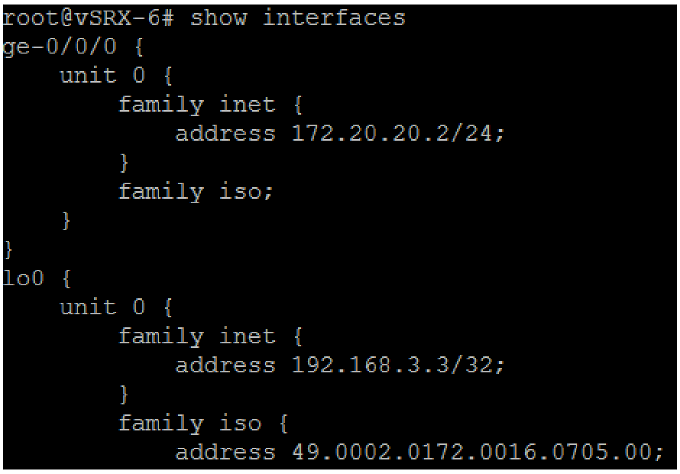
To configure IS-IS based on this topology we must assign IP addresses to the interfaces of the three routers including the loopback interface. The ISO NET address must also be assigned to each router with the same Area number and Area ID.

vSRX-3# show interfaces



```
root@vSRX-3# show interfaces
ge-0/0/1 {
    unit 0 {
        family inet {
            address 172.20.33.2/24;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.3.1/32;
        }
        family iso {
            address 49.0002.0172.0016.0305.00;
```

vSRX-6# show interfaces

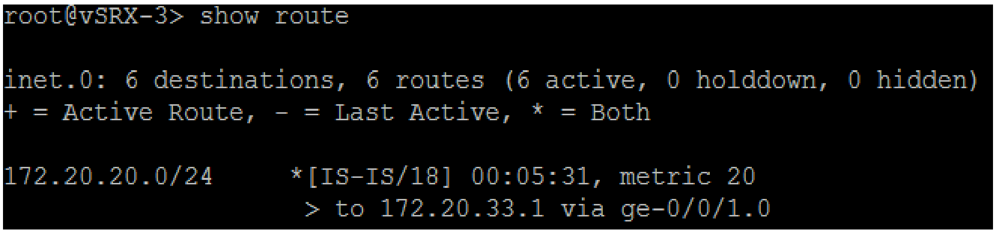


```
root@vSRX-6# show interfaces
ge-0/0/0 {
    unit 0 {
        family inet {
            address 172.20.20.2/24;
        }
        family iso;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 192.168.3.3/32;
        }
        family iso {
            address 49.0002.0172.0016.0705.00;
```

IS-IS Testing

After we have setup all the correct interfaces and ISO NET addresses, we can show the routes of **vSRX-3**, to find out if it knows the route to **vSRX-6**.
To do this, we can type: '*show route*', in operational mode.

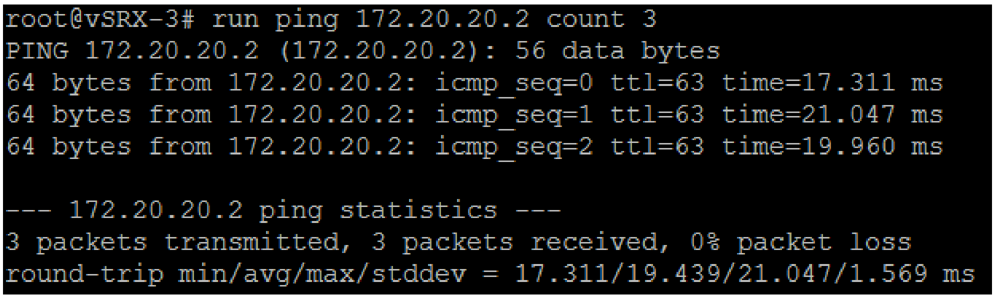
vSRX-3> show route



```
root@vSRX-3> show route

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

172.20.20.0/24     *[IS-IS/18] 00:05:31, metric 20
                    > to 172.20.33.1 via ge-0/0/1.0
```

We can see that **vSRX-3** knows the route to network 172.20.20.0/24, which is the network **vSRX-6** is operating on. The final step is to ping between these two devices to verify if they can communicate.

vSRX-3# run ping 172.20.20.2 count 3



```
root@vSRX-3# run ping 172.20.20.2 count 3
PING 172.20.20.2 (172.20.20.2): 56 data bytes
64 bytes from 172.20.20.2: icmp_seq=0 ttl=63 time=17.311 ms
64 bytes from 172.20.20.2: icmp_seq=1 ttl=63 time=21.047 ms
64 bytes from 172.20.20.2: icmp_seq=2 ttl=63 time=19.960 ms

--- 172.20.20.2 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 17.311/19.439/21.047/1.569 ms
```

We can verify that IS-IS is up and running and both routers can communicate. If now add more routers to this topology. Every new router that we add must be assigned with an ISO NET address that identifies it. However, the area number and area ID must remain the same.
The whole configuration of IS-IS can be found in Appendix D.

Border Gateway Protocol (BGP)

Summary

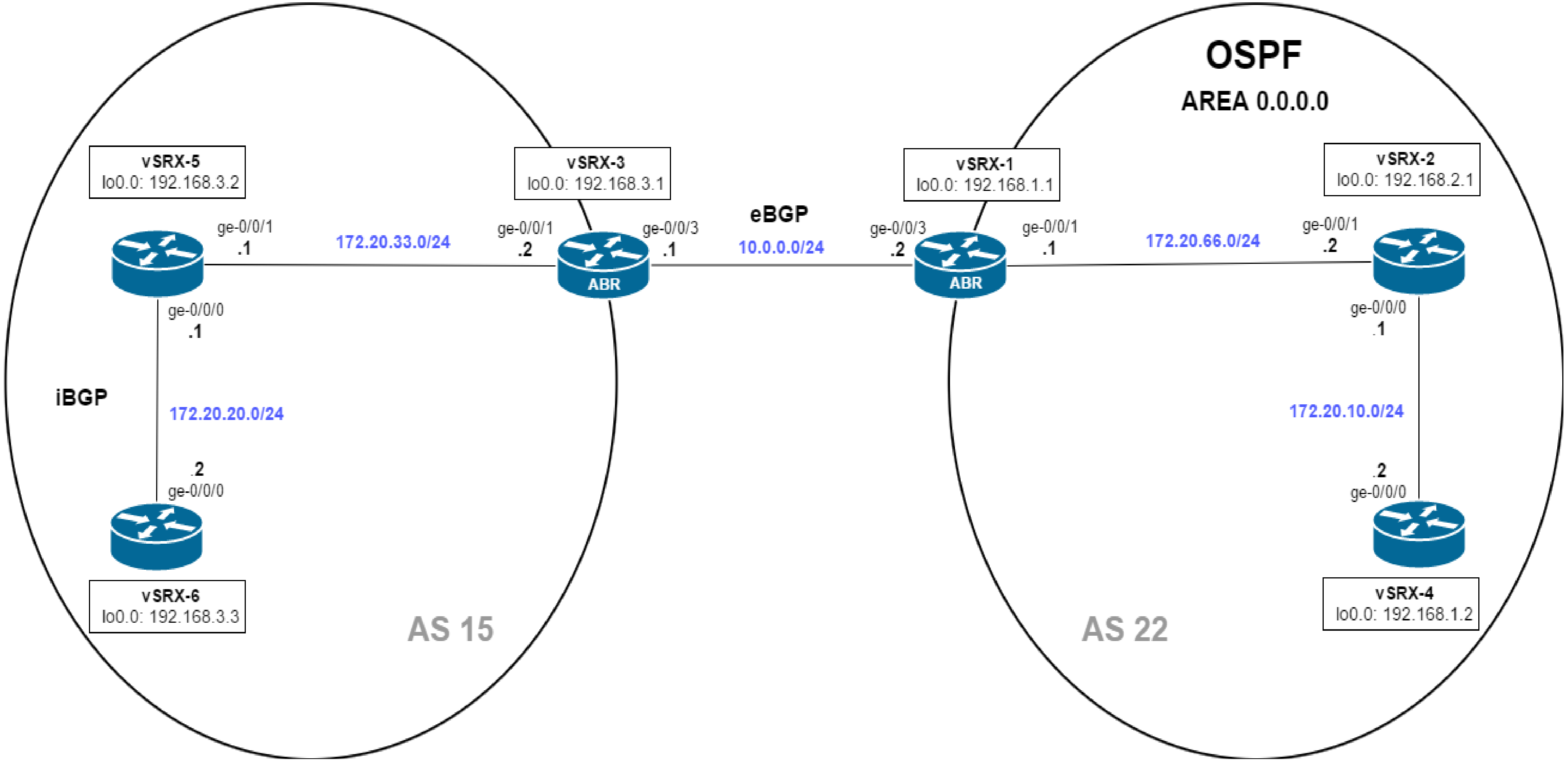
Border Gateway Protocol (BGP) is a routing protocol that manages how packets are routed across different Autonomous Systems (AS). Routers that run External BGP are known as the Area Border Routers (ABRs). The ABRs are commonly placed on the edge of an AS to route traffic to its neighbours that are also placed on the edge of another AS.
The protocol that enables the ABRs to this sort of communication is called, the External Border Gateway Protocol also known as (EBGP). Within the autonomous system itself, the routers use other protocols such as the Internal Border Gateway Protocol (IBGP), or Interior Gateway Protocols (IGPs) such as OSPF and IS-IS.
All these protocols are responsible for handling routing information and transmission between the nodes in an AS, often using the link-state protocol to choose the shortest path first.
An AS is often held by those with large network infrastructures with many smaller networks, such as, ISP, governments, educational institutions, or large enterprises. For the ABRs to address routing packets, every AS must have a unique ASN (Autonomous System Number) that identifies it. This number is given when the AS is officially registered by the Regional Internet Registry (RIR).[16]
The 16-bit Autonomous System numbers, from 64512-65534 are reserved for private use. The 32-bits numbers from 4.200.000.000 – 4.294.967.294 are for private use.[17]

[16] http://searchtelecom.techtarget.com/definition/BGP

[17] https://www.apnic.net/get-ip/faqs/asn/

BGP Configuration

BGP and OSPF network topology

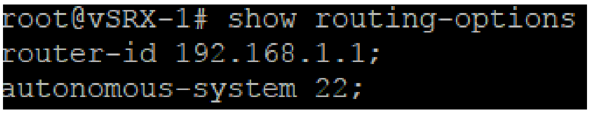


In this network topology, we have configured two virtual SRX routers to run as ABRs. Both routers are responsible for exchanging routing information across the two autonomous systems and they must allow communication for all devices.
For internal communication, we are using iBGP in AS 15 and in OSPF in AS 22.

Configuring External BGP

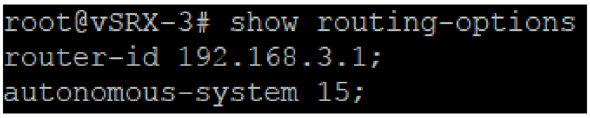
On both ABRs we have configured a router-id and an autonomous system number that indicates where it will operate. The address we have given the router-id is the same as the loopback address. We have done this because a loopback address never goes down, unless the router is shut off.

vSRX-1 (ABR)



```
root@vSRX-1# show routing-options
router-id 192.168.1.1;
autonomous-system 22;
```

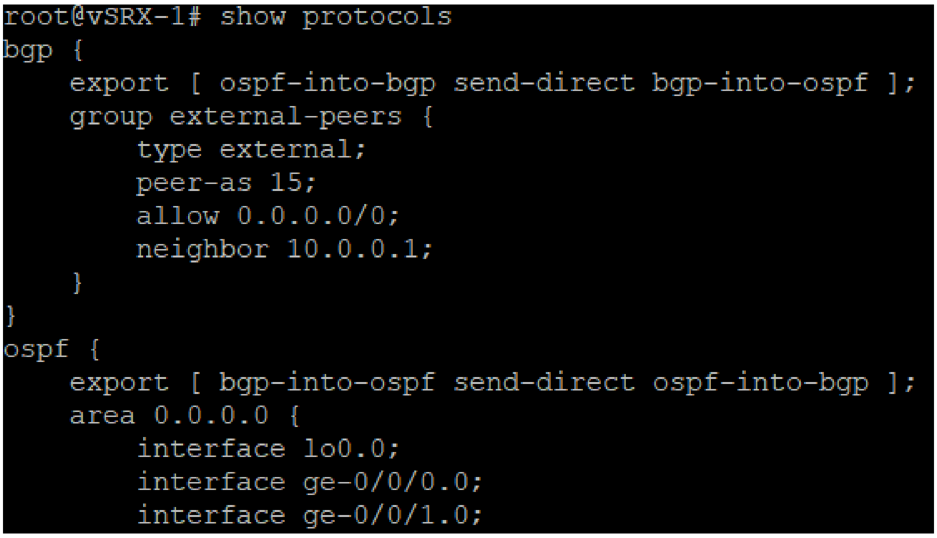
vSRX-3 (ABR)



```
root@vSRX-3# show routing-options
router-id 192.168.3.1;
autonomous-system 15;
```

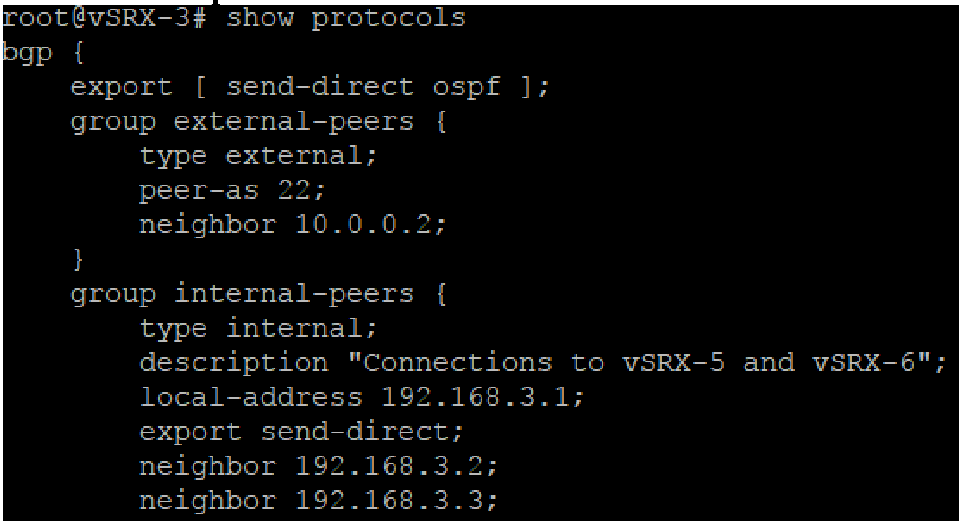
On both the **vSRX-1** and **vSRX-3**, we have redistributed Direct routes and OSPF routes. This allows for communication across the two autonomous systems and the routers can send packets to all devices.

vSRX-1# show protocols



```
root@vSRX-1# show protocols
bgp {
    export [ ospf-into-bgp send-direct bgp-into-ospf ];
    group external-peers {
        type external;
        peer-as 15;
        allow 0.0.0.0/0;
        neighbor 10.0.0.1;
    }
}
ospf {
    export [ bgp-into-ospf send-direct ospf-into-bgp ];
    area 0.0.0.0 {
        interface lo0.0;
        interface ge-0/0/0.0;
        interface ge-0/0/1.0;
```

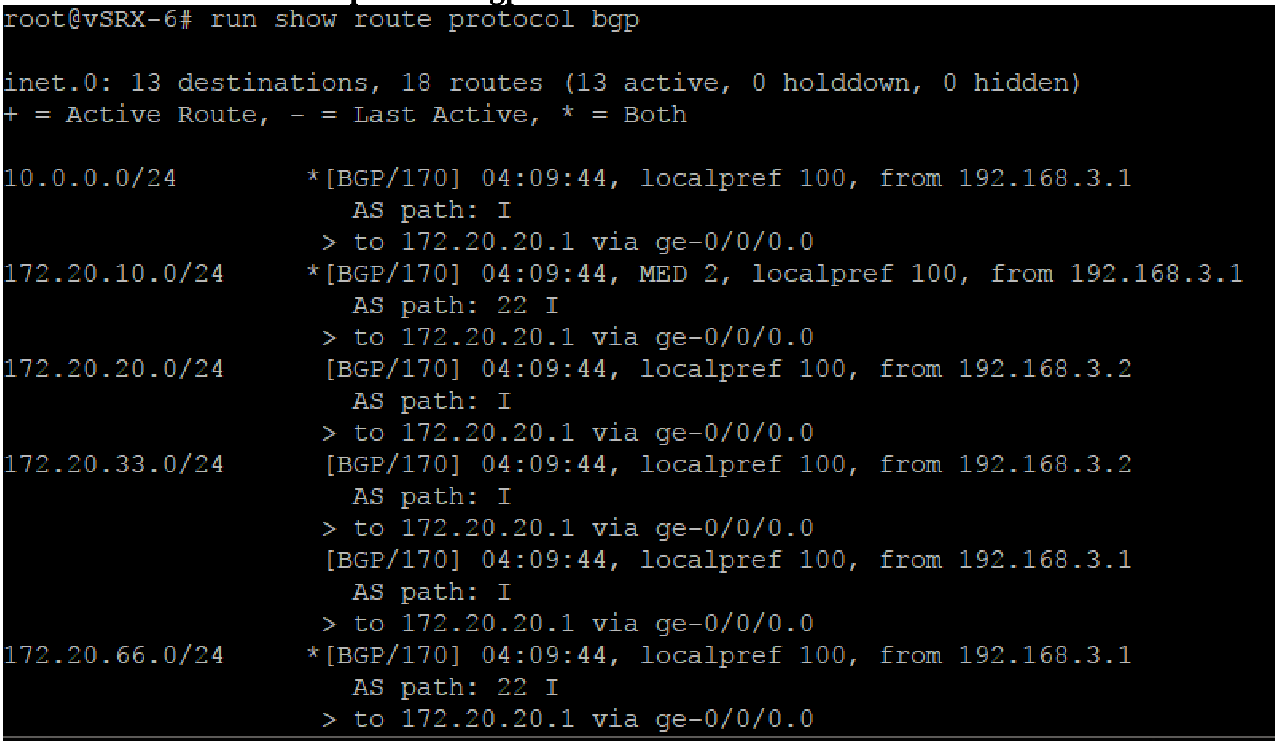
vSRX-3# show protocols



```
root@vSRX-3# show protocols
bgp {
    export [ send-direct ospf ];
    group external-peers {
        type external;
        peer-as 22;
        neighbor 10.0.0.2;
    }
    group internal-peers {
        type internal;
        description "Connections to vSRX-5 and vSRX-6";
        local-address 192.168.3.1;
        export send-direct;
        neighbor 192.168.3.2;
        neighbor 192.168.3.3;
```

To verify if **vSRX-6** knows route to all networks in AS 22, we can in edit mode type: '*run show route protocol bgp'*. If the network is shown with BGP as the preference, it means that the router knows about the routes and can start sending packets.

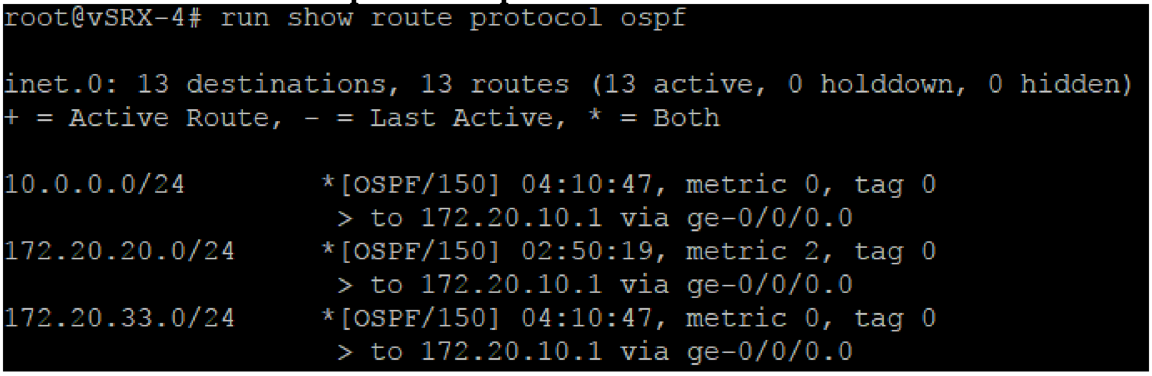
vSRX-6# run show route protocol bgp



```
root@vSRX-6# run show route protocol bgp

inet.0: 13 destinations, 18 routes (13 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/24        *[BGP/170] 04:09:44, localpref 100, from 192.168.3.1
                      AS path: I
                    > to 172.20.20.1 via ge-0/0/0.0
172.20.10.0/24     *[BGP/170] 04:09:44, MED 2, localpref 100, from 192.168.3.1
                      AS path: 22 I
                    > to 172.20.20.1 via ge-0/0/0.0
172.20.20.0/24      [BGP/170] 04:09:44, localpref 100, from 192.168.3.2
                      AS path: I
                    > to 172.20.20.1 via ge-0/0/0.0
172.20.33.0/24      [BGP/170] 04:09:44, localpref 100, from 192.168.3.2
                      AS path: I
                    > to 172.20.20.1 via ge-0/0/0.0
                    [BGP/170] 04:09:44, localpref 100, from 192.168.3.1
                      AS path: I
                    > to 172.20.20.1 via ge-0/0/0.0
172.20.66.0/24     *[BGP/170] 04:09:44, localpref 100, from 192.168.3.1
                      AS path: 22 I
                    > to 172.20.20.1 via ge-0/0/0.0
```

To verify if **vSRX-4** knows route to all networks in AS 15, we can in edit mode type: '*run show route'*
If the network is shown with OSPF as the preference, it means that the router knows about the routes and can start sending packets.

vSRX-4# run show route protocol ospf

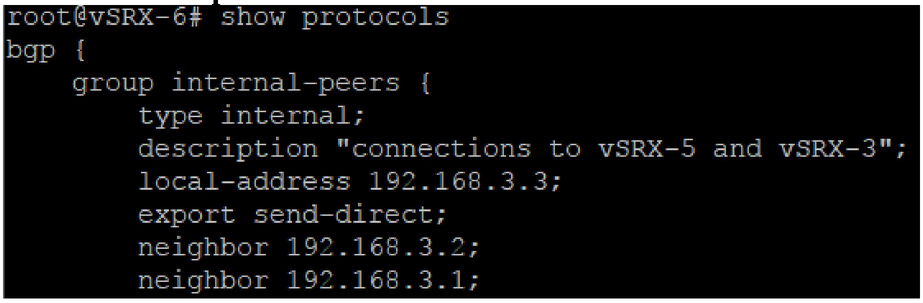


```
root@vSRX-4# run show route protocol ospf

inet.0: 13 destinations, 13 routes (13 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

10.0.0.0/24        *[OSPF/150] 04:10:47, metric 0, tag 0
                    > to 172.20.10.1 via ge-0/0/0.0
172.20.20.0/24     *[OSPF/150] 02:50:19, metric 2, tag 0
                    > to 172.20.10.1 via ge-0/0/0.0
172.20.33.0/24     *[OSPF/150] 04:10:47, metric 0, tag 0
                    > to 172.20.10.1 via ge-0/0/0.0
```

Configuring Internal BGP (iBGP):

Inside the protocol BGP, we have created an internal-peers group. Inside of this group we have configured vSRX-5 and vSRX-6 as its neighbours.

vSRX-6# show protocols

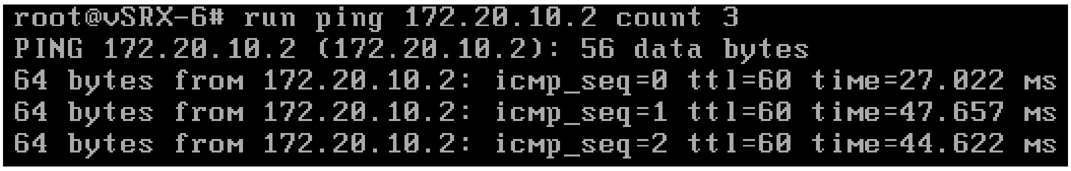


```
root@vSRX-6# show protocols
bgp {
    group internal-peers {
        type internal;
        description "connections to vSRX-5 and vSRX-3";
        local-address 192.168.3.3;
        export send-direct;
        neighbor 192.168.3.2;
        neighbor 192.168.3.1;
```

BGP Testing

Now that we have configured all the routers we can test to see if the routers can communicate across the two autonomous systems.
-Here we have managed to ping **vSRX-4** from **vSRX-6.**

vSRX-6# run ping 172.20.10.2 count 3



```
root@vSRX-6# run ping 172.20.10.2 count 3
PING 172.20.10.2 (172.20.10.2): 56 data bytes
64 bytes from 172.20.10.2: icmp_seq=0 ttl=60 time=27.022 ms
64 bytes from 172.20.10.2: icmp_seq=1 ttl=60 time=47.657 ms
64 bytes from 172.20.10.2: icmp_seq=2 ttl=60 time=44.622 ms
```

The main goal of this exercise was to get all the routers to communicate across the two autonomous systems. We can successfully verify that iBGP, eBGP and OSPF are all active and running and we can ping all routers internal and external across the two autonomous systems.
The whole configuration of BGP can be found in Appendix E.

Class of Service (COS)

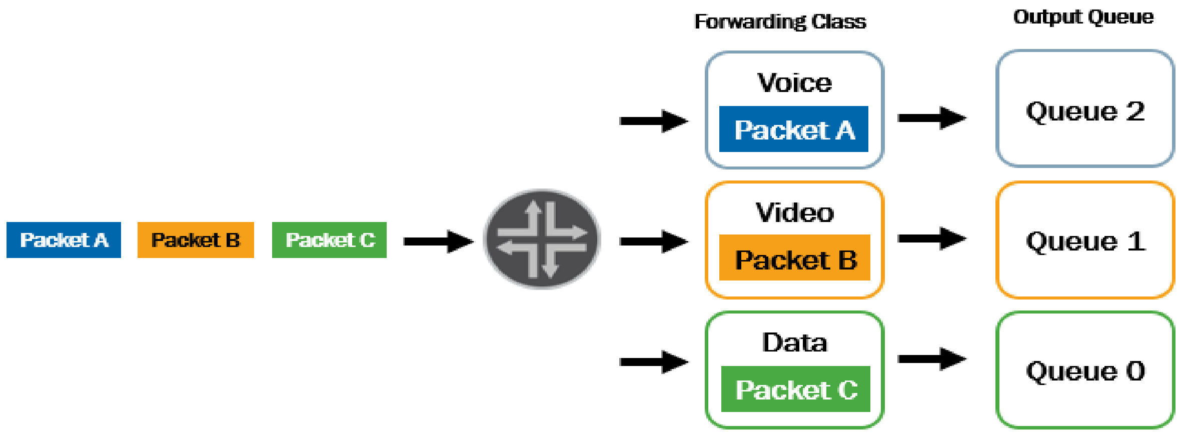
Summary

Class of Service (CoS) is a term that is used to describe the process of managing different kinds of traffic within a network. It provides mechanisms for categorizing traffic and meeting performance requirements within a network, by prioritizing latency-sensitive traffic such as Voice over IP (VoIP) or multimedia. With CoS implemented in a network, packets that must be dropped due to congestions, can only be dropped on a term of rules that are set by the network administrators or ISPs. This is done by dividing different type of traffic such as voice, video or data into classes and allocating bandwidth to them.
By default, a Juniper router comes with CoS implemented in the hardware rather than the software. This allows the router to treat all traffic equally and control congestion by dropping packets using Random Early Detection (RED) algorithm. CoS does not make a network faster or reduce congestions but it does however improve the performance of voice or multimedia traffic, by prioritizing it over normal data traffic.[18]

Forwarding classes and Queues

On a Juniper router, incoming traffic is associated with a forwarding class during the output. The forwarding class is, depending on its membership linked with a queue. Once the traffic is placed into the correct queue, a scheduler defines how the interfaces should process that traffic. The number of queues supported on outbound interfaces are hardware dependent and each queue describe the importance of traffic. There are 4 default queue numbers from 0-3 in most devices running Junos OS.

Figure C-1: Traffic and forwarding classes

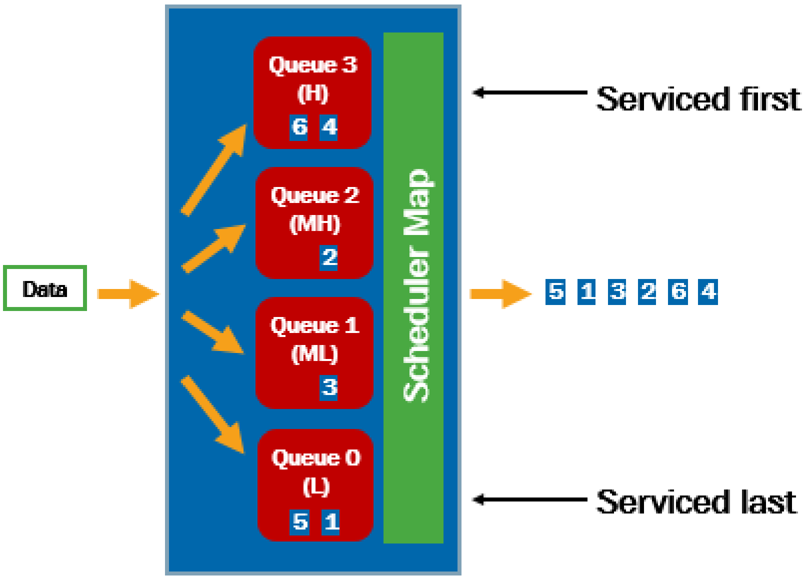


[18] https://fronter.com/eal/links/files.phtml/2080432588$596013739$/2nd+Semester/Project+Network/Class+of+Service+_prcent_28CoS_prcent_29/Junos+Routing+Essentials+_prcent_28JRE_prcent_29/JNAA-JRE-Junosphere-12.c-R_SG+Revised+v2.pdf

Scheduling

A scheduler is associated with a queue and forwarding class through a scheduler map. When configuring a scheduler on a juniper router we can specify components such as: priority, transmission rate, buffer size and RED. Priority and transmission rate define the order in which packet transmit and how much of bandwidth that forwarding class gets. Buffer size and RED define how many packets can be held, and which should be dropped during congestion.

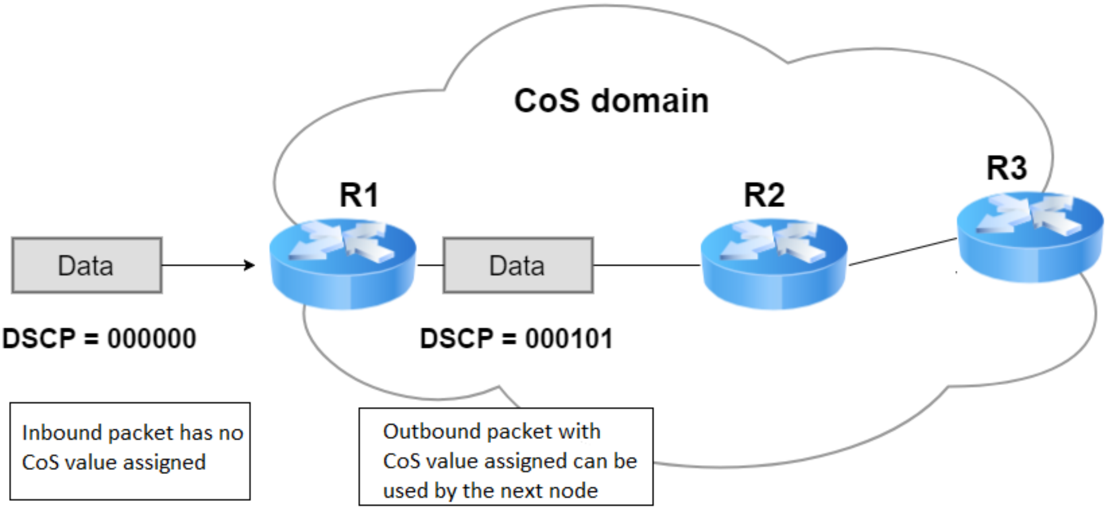
Figure C-2: Queues and scheduler map



Deployment Method: Behavior Aggregate (BA)

When an edge router receives an incoming packet, the router can be configured to set behavior aggregate (BA) marking on that packet. BA is a set of CoS values (bits) that are given to a packet IP header in the Type of Service (ToS) field. This process takes place before the packet leaves the outbound interface. Once BA marking is applied to a packet, the same set of rules will be used by the downstream nodes in the same network.

Figure C-3: Behavior aggregate (BA) classifier

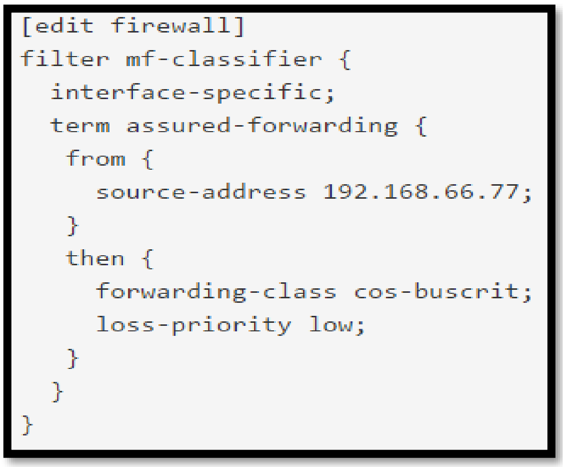


CoS domain
R1
R2
R3
Data
Data
DSCP = 000000
DSCP = 000101
Inbound packet has no CoS value assigned
Outbound packet with CoS value assigned can be used by the next node

Deployment Method: Multifield Classifier

Multifield Classifier (MF) is used to classify packets by looking at different fields such as source address and destination address. When a packet is received by 192.168.66.77, MF places that packet into a forwarding class with different terms such as transmission rate and high priority. The packet will then be forwarded to another router, but the terms that the receiving router made will no longer exist. Forwarding classes and priorities are configured under firewall filters on a Juniper router. Multifield Classifier does not place BA marking on packets, and is configured on a only a single Juniper router.

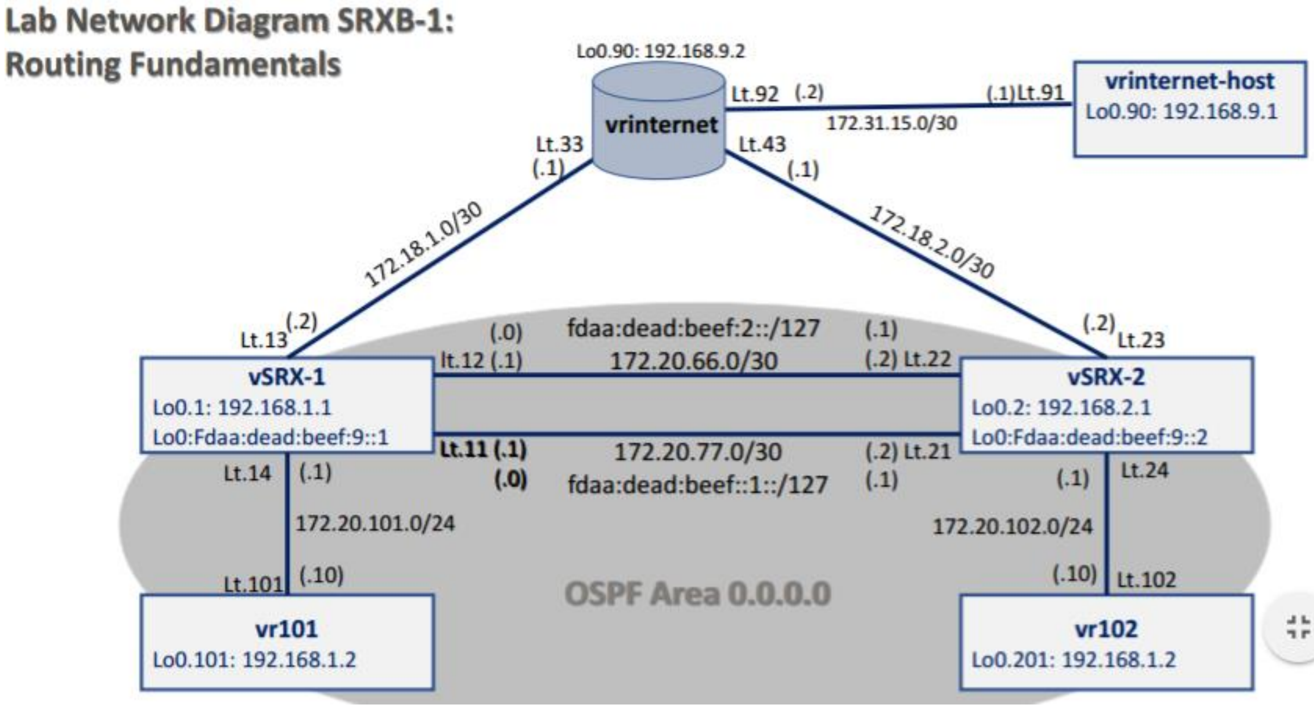
Figure C-4: MF Classifier configuration



```
[edit firewall]
filter mf-classifier {
  interface-specific;
  term assured-forwarding {
   from {
     source-address 192.168.66.77;
   }
   then {
     forwarding-class cos-buscrit;
     loss-priority low;
   }
  }
}
```

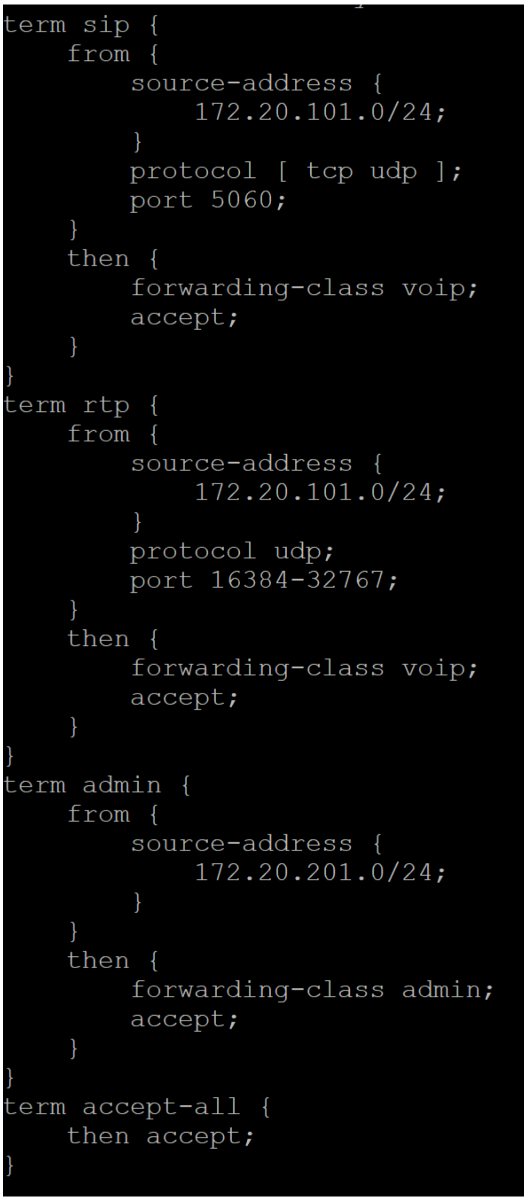
Class of Service Configuration

Network Topology



This configuration shows implementation of CoS on two single juniper SRX routers.
We have configured vSRX-1 to forward incoming packets from vr101, and from 172.20.201.0/24 sources, onto forwarding classes. On this router, we have under firewall filters, different terms for incoming packets.

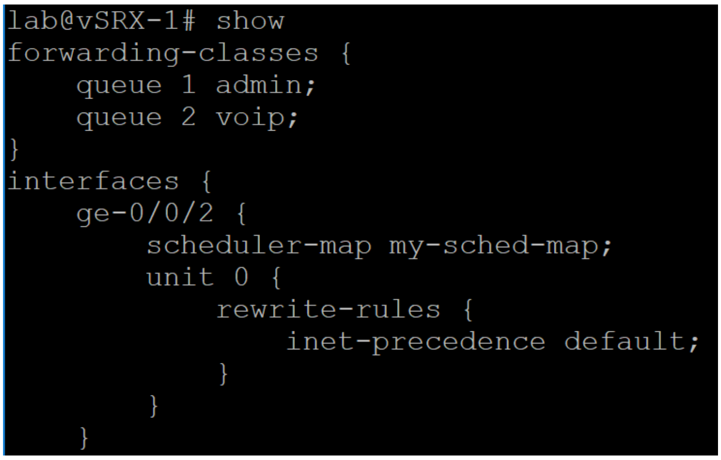
vSRX-1 Ingress Multifield Classifier



```
term sip {
    from {
        source-address {
            172.20.101.0/24;
        }
        protocol [ tcp udp ];
        port 5060;
    }
    then {
        forwarding-class voip;
        accept;
    }
}
term rtp {
    from {
        source-address {
            172.20.101.0/24;
        }
        protocol udp;
        port 16384-32767;
    }
    then {
        forwarding-class voip;
        accept;
    }
}
term admin {
    from {
        source-address {
            172.20.201.0/24;
        }
    }
    then {
        forwarding-class admin;
        accept;
    }
}
term accept-all {
    then accept;
}
```

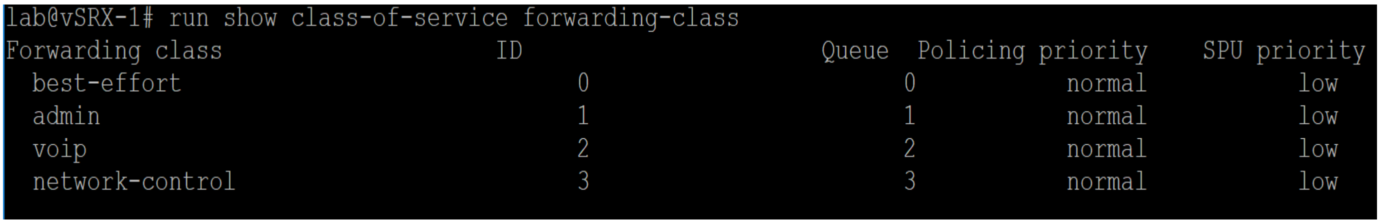
We have two forwarding classes, admin and voip that we can configured. On Junos routers, default forwarding classes are, best-effort and network-control.

vSRX-1 Behavior Aggregate Rewrite and Scheduler Application:



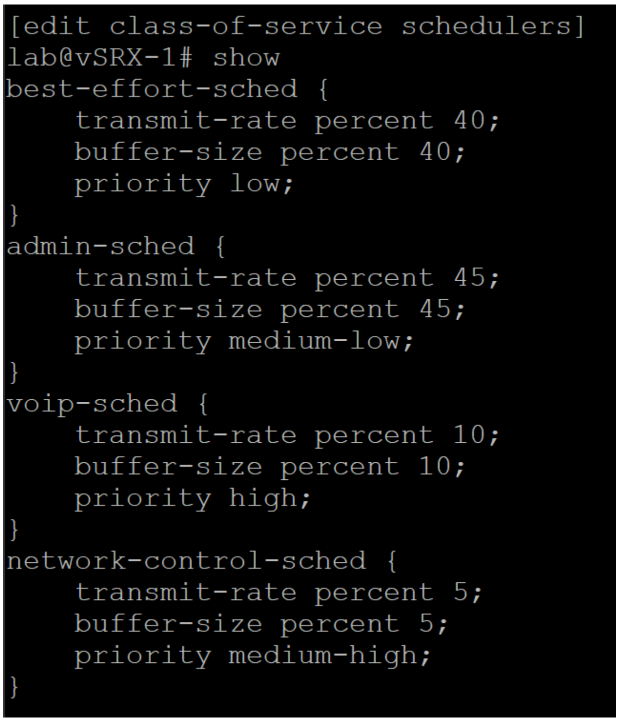
```
lab@vSRX-1# show
forwarding-classes {
    queue 1 admin;
    queue 2 voip;
}
interfaces {
    ge-0/0/2 {
        scheduler-map my-sched-map;
        unit 0 {
            rewrite-rules {
                inet-precedence default;
            }
        }
    }
```

This picture displays about current queue and forwarding class mapping here:



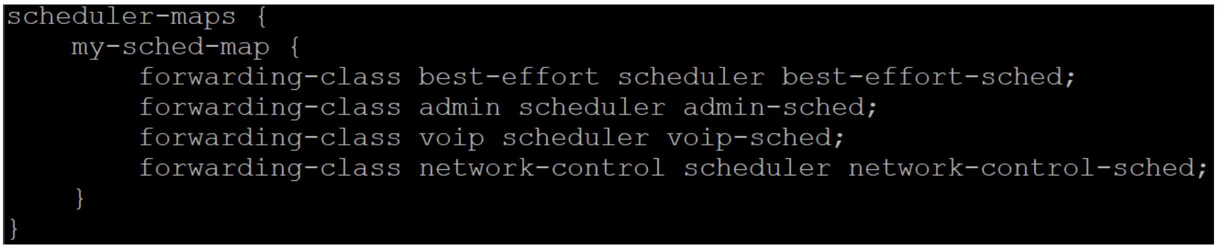
```
lab@vSRX-1# run show class-of-service forwarding-class
Forwarding class                        ID                      Queue  Policing priority    SPU priority
  best-effort                                 0                       0            normal             low
  admin                                       1                       1            normal             low
  voip                                        2                       2            normal             low
  network-control                             3                       3            normal             low
```

Schedulers are defining properties of the output queues:



```
[edit class-of-service schedulers]
lab@vSRX-1# show
best-effort-sched {
    transmit-rate percent 40;
    buffer-size percent 40;
    priority low;
}
admin-sched {
    transmit-rate percent 45;
    buffer-size percent 45;
    priority medium-low;
}
voip-sched {
    transmit-rate percent 10;
    buffer-size percent 10;
    priority high;
}
network-control-sched {
    transmit-rate percent 5;
    buffer-size percent 5;
    priority medium-high;
}
```

Scheduler-maps displays the mapping of schedulers to the forwarding classes:



```
scheduler-maps {
    my-sched-map {
        forwarding-class best-effort scheduler best-effort-sched;
        forwarding-class admin scheduler admin-sched;
        forwarding-class voip scheduler voip-sched;
        forwarding-class network-control scheduler network-control-sched;
    }
}
```

The whole configuration of CoS can be found in Appendix F.

Conclusion

By working with different components and protocols of computer networking, we learned that everything is one way or another related. Protocols such as OSPF, and IS-IS uses link-state algorithm to make routing calculations. BGP is used to connect two, or more autonomous systems together, and route packets within the network and across different autonomous systems. We have learned how Ethernet works and how we can set up multiple VLANs to control traffic on different broadcast domains. We have also learned how each layer of the TCP/IP model functions, and how they are used to allow one computer to talk with the other. We have learned the similarities and differences between IPv4 and IPv6, and the advantages that comes with IPv6.

By working on this report, we realized the need of diving tasks to each group member so we could ensure that we all would reach our deadlines. Some of the challenges that we faced in our group were especially the configuration of BGP with route-redistribution, and VLANs as we did not have access to switches and had to use Juniper SRX routers as layer-3 switches.

Researching on Class of Service was also a difficult part as many sources that could be found on the internet were either too detailed or poor. However, staying in contact with our lecturer allowed us to be in the right path and was helpful in many different aspects throughout this report.

Appendix

Appendix A

Configuration of: srxD-1

```
## Last changed: 2017-06-04 13:14:47 UTC
version 12.1X46-D30.2;
system {
    host-name srxD-1;
    root-authentication {
        encrypted-password "$1$ZpRtRPU8$u4ezHJ37Y5J0k.hMclXoA/"; ## SECRET-DATA
        ssh-dsa "ssh-dss
AAAAB3NzaC1kc3MAAACBAMQrfP2bZyBXJ6PC7XXZ+MzErI8Jl6jah5L4/O8BsfP2hC7EvRfNoX7Mqbr
tCX/9gUH9gChVuBCB+ERULMdgRvM5uGhC/gs4UX+4dBbfBgKYYwgmisM8EoT25m7qI8ybpl2YZvHNznvO8h7kr4kp
YuQEpKvgsTdH/Jl
e4Uqnjv7DAAAAFQDZaqA6QAgbW3O/zveaLCIDj6p0dwAAAIB1iL+krWrXiD8NPpY+w4dWXEqaV3bnobzPC4eyxQKB
UCOr80Q5YBlWXVB
Hx9elwBWZwj0SF4hLKHznExnLerVsMuTMA846RbQmSz62vM6kGM13HFonWeQvWia0TDr78+rOEgWF2KHBSIxL51lm
IDW8Gql9hJfD/Dr
/NKP97w3L0wAAAIEAr3FkWU8XbYytQYEKxsIN9P1UQ1ERXB3G40YwqFO484SlyKyYCfaz+yNsaAJu2C8UebDIR3Gi
eyNcOAKf3inCG8j
QwjLvZskuZwrvlsz/xtcxSoAh9axJcdUfSJYMW/g+mD26JK1Cliw5rwp2nH9kUrJxeI7IReDp4egNkM4i15o=
configurator@serve
r1.he"; ## SECRET-DATA
    }
    login {
        user lab {
            uid 2000;
            class super-user;
            authentication {
                encrypted-password "$1$84J5Maes$cni5Hrazbd/IEHr/50oY30"; ## SECRET-DATA
            }
        }
    }
    services {
        ftp;
        ssh;
        telnet;
        netconf {
            ssh;
        }
        web-management {
            http {
                interface ge-0/0/0.0;
            }
            https {
                system-generated-certificate;
                interface all;
            }
        }
    }
    syslog {
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
}
interfaces {
    ge-0/0/0 {
        description "MGMT Interface - DO NOT DELETE";
        unit 0 {
            family inet {
                address 10.210.14.149/24;
            }
        }
    }
    ge-0/0/2 {
        unit 0 {
            family ethernet-switching {
                port-mode trunk;
                vlan {
                    members [ vlan-tech vlan-admin ];
                }
            }
        }
    }
    ge-0/0/3 {
        unit 0 {
            family ethernet-switching {
                port-mode access;
                vlan {
                    members vlan-admin;
                }
            }
        }
    }
    ge-0/0/4 {
        unit 0 {
            family ethernet-switching {
                port-mode access;
                vlan {
                    members vlan-tech;
                }
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.1.2/24;
            }
        }
    }
    vlan {
        unit 10 {
            family inet {
                address 192.168.1.2/24;
            }
        }
        unit 20 {
            family inet {
                address 192.168.2.2/24;
            }
        }
    }
}
security {
    forwarding-options {
        family {
            iso {
                mode packet-based;
            }
        }
    }
}
```

```
routing-instances {
    vrvlan {
        instance-type virtual-router;
        interface vlan.10;
    }
    vrvlan2 {
        instance-type virtual-router;
        interface vlan.20;
    }
}
vlans {
    vlan-admin {
        vlan-id 10;
        interface {
            ge-0/0/3.0;
        }
        l3-interface vlan.10;
    }
    vlan-tech {
        vlan-id 20;
        interface {
            ge-0/0/4.0;
        }
        l3-interface vlan.20;
    }
}
```

Configuration of: srxD-2

```
## Last changed: 2017-06-04 13:03:32 UTC
version 12.1X46-D30.2;
system {
    host-name srxD-2;
    root-authentication {
        encrypted-password "$1$ZpRtRPU8$u4ezHJ37Y5J0k.hMclXoA/"; ## SECRET-DATA
        ssh-dsa "ssh-dss
AAAAB3NzaC1kc3MAAACBAMQrfP2bZyBXJ6PC7XXZ+MzErI8Jl6jah5L4/O8BsfP2hC7EvRfNoX7Mqbr
tCX/9gUH9gChVuBCB+ERULMdgRvM5uGhC/gs4UX+4dBbfBgKYYwgmisM8EoT25m7qI8ybpl2YZvHNznvO8h7kr4kp
YuQEpKvgsTdH/Jl
e4Uqnjv7DAAAAFQDZaqA6QAgbW3O/zveaLCIDj6p0dwAAAIB1iL+krWrXiD8NPpY+w4dWXEqaV3bnobzPC4eyxQKB
UCOr80Q5YBlWXVB
Hx9elwBWZwj0SF4hLKHznExnLerVsMuTMA846RbQmSz62vM6kGM13HFonWeQvWia0TDr78+rOEgWF2KHBSIxL51lm
IDW8Gql9hJfD/Dr
/NKP97w3L0wAAAIEAr3FkWU8XbYytQYEKxsIN9P1UQ1ERXB3G40YwqFO484SlyKyYCfaz+yNsaAJu2C8UebDIR3Gi
eyNcOAKf3inCG8j
QwjLvZskuZwrvlsz/xtcxSoAh9axJcdUfSJYMW/g+mD26JK1Cliw5rwp2nH9kUrJxeI7IReDp4egNkM4i15o=
configurator@serve
r1.he"; ## SECRET-DATA
    }
    login {
        user lab {
            uid 2000;
            class super-user;
            authentication {
                encrypted-password "$1$84J5Maes$cni5Hrazbd/IEHr/50oY30"; ## SECRET-DATA
            }
        }
    }
    services {
        ftp;
        ssh;
        telnet;
        netconf {
            ssh;
        }
        web-management {
            http {
                interface ge-0/0/0.0;
            }
            https {
                system-generated-certificate;
                interface all;
            }
        }
    }
    syslog {
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
}
interfaces {
    ge-0/0/0 {
        description "MGMT Interface - DO NOT DELETE";
        unit 0 {
            family inet {
                address 10.210.14.150/24;
            }
        }
    }
    ge-0/0/2 {
        unit 0 {
            family ethernet-switching {
                port-mode trunk;
                vlan {
                    members [ vlan-tech vlan-admin ];
                }
            }
        }
    }
    ge-0/0/3 {
        unit 0 {
            family ethernet-switching {
                port-mode access;
                vlan {
                    members vlan-admin;
                }
            }
        }
    }
    ge-0/0/4 {
        unit 0 {
            family ethernet-switching {
                vlan {
                    members vlan-tech;
                }
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.1.1/32;
            }
        }
    }
    vlan {
        unit 10 {
            family inet {
                address 192.168.1.1/24;
            }
        }
        unit 20 {
            family inet {
                address 192.168.2.1/24;
            }
        }
    }
}
security {
    forwarding-options {
        family {
            iso {
                mode packet-based;
            }
        }
    }
}
routing-instances {
    vrvlan {
        instance-type virtual-router;
        interface vlan.10;
    }
    vrvlan2 {
        instance-type virtual-router;
        interface vlan.20;
    }
}
vlans {
    vlan-admin {
        vlan-id 10;
        interface {
            ge-0/0/3.0;
        }
        l3-interface vlan.10;
    }
    vlan-tech {
        vlan-id 20;
        interface {
            ge-0/0/4.0;
        }
        l3-interface vlan.20;
    }
}
```

Appendix B

Configuration of: vSRX-A4

```
## Last changed: 2017-06-02 00:51:31 UTC
version 12.1X46-D30.2;
system {
    host-name SRX-A4;
    root-authentication {
        encrypted-password "$1$VHyIWhsl$GpERroh6a22orYL3it6XV/"; ## SECRET-DATA
    }
    login {
        user nick {
            uid 2001;
            class super-user;
            authentication {
                encrypted-password "$1$GQl.iPWl$nmn6pK./0lPyQX./Oh1CD/"; ## SECRET-DATA
            }
        }
    }
    services {
        ssh;
        telnet;
        web-management {
            http;
        }
    }
}
interfaces {
    ge-0/0/0 {
        description "DO NOT DELETE";
        unit 0 {
            family inet {
                address 10.210.14.136/26;
            }
        }
    }
    lt-0/0/0 {
        unit 1 {
            encapsulation ethernet;
            peer-unit 2;
            family inet6 {
                address fdaa:dead:beef:1::0/127;
            }
        }
        unit 2 {
            encapsulation ethernet;
            peer-unit 1;
            family inet6 {
                address fdaa:dead:beef:1::1/127;
            }
        }
        unit 3 {
            encapsulation ethernet;
            peer-unit 4;
            family inet6 {
                address fdaa:dead:beef:2::0/127;
            }
        }
        unit 4 {
            encapsulation ethernet;
            peer-unit 3;
            family inet6 {
                address fdaa:dead:beef:2::1/127;
            }
        }
    }
    lo0 {
        unit 1 {
            family inet6 {
                address fdaa:dead:beef:9::1/128;
            }
        }
        unit 2 {
            family inet6 {
                address fdaa:dead:beef:9::2/128;
            }
        }
        unit 3 {
            family inet6 {
                address fdaa:dead:beef:9::3/128;
            }
        }
    }
}
security {
    forwarding-options {
        family {
            inet6 {
                mode packet-based;
            }
            mpls {
                mode packet-based;
            }
        }
    }
}
routing-instances {
    vSRX1 {
        instance-type virtual-router;
        interface lt-0/0/0.1;
        interface lo0.1;
        routing-options {
            router-id 192.168.1.1;
        }
        protocols {
            ospf3 {
                area 0.0.0.0 {
                    interface lt-0/0/0.1;
                    interface lo0.1;
                }
            }
        }
    }
    vSRX2 {
        instance-type virtual-router;
        interface lt-0/0/0.2;
        interface lt-0/0/0.3;
        interface lo0.2;
        routing-options {
            router-id 192.168.2.1;
        }
        protocols {
            ospf3 {
                area 0.0.0.0 {
                    interface lt-0/0/0.2;
                    interface lt-0/0/0.3;
                    interface lo0.2;
                }
            }
        }
    }
    vSRX3 {
        instance-type virtual-router;
        interface lt-0/0/0.4;
        interface lo0.3;
        routing-options {
            router-id 192.168.3.1;
        }
        protocols {
            ospf3 {
                area 0.0.0.0 {
                    interface lt-0/0/0.4;
                    interface lo0.3;
                }
            }
        }
    }
```

```
}
```

Appendix C

Configuration of: vSRX-1

```
## Last changed: 2017-06-01 18:23:44 UTC
version 12.1X47-D15.4;
groups {
    global;
}
system {
    host-name vSRX-1;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
        telnet;
        xnm-clear-text;
        web-management {
            http {
                interface vlan.0;
            }
            https {
                system-generated-certificate;
                interface vlan.0;
            }
        }
        dhcp {
            router {
                192.168.1.1;
            }
            pool 192.168.1.0/24 {
                address-range low 192.168.1.2 high 192.168.1.254;
            }
            propagate-settings ge-0/0/0.0;
        }
    }
    syslog {
        archive size 100k files 3;
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    ##
    ## Warning: statement ignored: unsupported platform (firefly-perimeter)
    ##
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-0/0/1 {
        unit 0 {
            family inet {
                address 172.20.66.1/24;
            }
        }
    }
    ge-0/0/3 {
        unit 0 {
            description to-vSRX-3;
            family inet {
                address 10.0.0.2/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.1.1/32;
            }
        }
    }
}
routing-options {
    router-id 192.168.1.1;
    autonomous-system 22;
}
protocols {
    bgp {
        export [ direct-into-bgp ospf-into-bgp ];
        group external-peers {
            type external;
            peer-as 15;
            allow 0.0.0.0/0;
            neighbor 10.0.0.1;
        }
    }
    ospf {
        export [ bgp-into-ospf direct-into-bgp ];
        area 0.0.0.0 {
            interface lo0.0;
            interface ge-0/0/0.0;
            interface ge-0/0/1.0;
        }
    }
}
policy-options {
    policy-statement bgp-into-ospf {
        term bgp-only {
            from protocol bgp;
            then accept;
        }
    }
    policy-statement direct-into-bgp {
        term direct-only {
            from protocol direct;
            then accept;
        }
    }
    policy-statement ospf-into-bgp {
        term ospf-only {
            from {
                protocol ospf;
                area 0.0.0.0;
            }
            then accept;
        }
    }
}
security {
    forwarding-options {
        family {
            inet6 {
                mode packet-based;
            }
            mpls {
                mode packet-based;
            }
        }
    }
}
```

Configuration of: vSRX-2

```
## Last changed: 2017-06-01 18:23:47 UTC
version 12.1X47-D15.4;
groups {
    global;
}
system {
    host-name vSRX-2;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
        telnet;
        xnm-clear-text;
        web-management {
            http {
                interface vlan.0;
            }
            https {
                system-generated-certificate;
                interface vlan.0;
            }
        }
        dhcp {
            router {
                192.168.1.1;
            }
            pool 192.168.1.0/24 {
                address-range low 192.168.1.2 high 192.168.1.254;
            }
            propagate-settings ge-0/0/0.0;
        }
    }
    syslog {
        archive size 100k files 3;
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    ##
    ## Warning: statement ignored: unsupported platform (firefly-perimeter)
    ##
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 172.20.10.1/24;
            }
        }
    }
    ge-0/0/1 {
        unit 0 {
            family inet {
                address 172.20.66.2/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.2.1/32;
            }
        }
    }
}
protocols {
    ospf {
        area 0.0.0.0 {
            interface lo0.0;
            interface ge-0/0/0.0;
            interface ge-0/0/1.0;
        }
    }
}
security {
    forwarding-options {
        family {
            inet6 {
                mode packet-based;
            }
            mpls {
                mode packet-based;
            }
        }
    }
}
```

Configuration of: vSRX-4

```
## Last changed: 2017-06-01 18:23:35 UTC
version 12.1X47-D15.4;
```

```
groups {
    global;
}
system {
    host-name vSRX-4;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
        telnet;
        xnm-clear-text;
        web-management {
            http {
                interface vlan.0;
            }
            https {
                system-generated-certificate;
                interface vlan.0;
            }
        }
        dhcp {
            router {
                192.168.1.1;
            }
            pool 192.168.1.0/24 {
                address-range low 192.168.1.2 high 192.168.1.254;
            }
            propagate-settings ge-0/0/0.0;
        }
    }
    syslog {
        archive size 100k files 3;
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    ##
    ## Warning: statement ignored: unsupported platform (firefly-perimeter)
    ##
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 172.20.10.2/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.1.2/32;
            }
        }
    }
}
protocols {
    ospf {
        area 0.0.0.0 {
            interface lo0.0;
            interface ge-0/0/0.0;
        }
    }
}
security {
    forwarding-options {
        family {
            inet6 {
                mode packet-based;
            }
            mpls {
                mode packet-based;
            }
        }
    }
}
```

Appendix D

Configuration of: vSRX-3

```
## Last changed: 2017-06-01 17:14:58 UTC
version 12.1X47-D15.4;
groups {
    global;
}
system {
    host-name vSRX-3;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
        telnet;
        xnm-clear-text;
        web-management {
            http {
                interface vlan.0;
            }
            https {
                system-generated-certificate;
                interface vlan.0;
            }
        }
        dhcp {
            router {
                192.168.1.1;
            }
            pool 192.168.1.0/24 {
                address-range low 192.168.1.2 high 192.168.1.254;
            }
            propagate-settings ge-0/0/0.0;
        }
    }
    syslog {
        archive size 100k files 3;
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    ##
    ## Warning: statement ignored: unsupported platform (firefly-perimeter)
    ##
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-0/0/1 {
        unit 0 {
            family inet {
                address 172.20.33.2/24;
            }
            family iso;
        }
    }
    ge-0/0/3 {
        unit 0 {
            description to-vSRX-1;
            family inet {
                address 10.0.0.1/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.3.1/32;
            }
            family iso {
                address 49.0002.0172.0016.0305.00;
            }
        }
    }
}
routing-options {
    router-id 192.168.3.1;
    autonomous-system 15;
}
protocols {
    bgp {
        export [ isis-into-bgp direct-into-bgp ];
        group external-peers {
            type external;
            peer-as 22;
            neighbor 10.0.0.2;
        }
    }
    isis {
        export [ bgp-into-isis direct-into-bgp ];
        interface ge-0/0/1.0 {
            level 1 disable;
        }
        interface lo0.0;
    }
}
policy-options {
    policy-statement bgp-into-isis {
        term bgp-only {
            from protocol bgp;
            then accept;
        }
    }
    policy-statement direct-into-bgp {
        term direct-only {
            from protocol direct;
            then accept;
        }
    }
    policy-statement isis-into-bgp {
        term isis-only {
            from protocol isis;
            then accept;
        }
    }
}
security {
    forwarding-options {
        family {
            mpls {
                mode packet-based;
            }
            iso {
                mode packet-based;
            }
        }
    }
}
```

Configuration of: vSRX-5

```
## Last changed: 2017-06-01 17:15:00 UTC
version 12.1X47-D15.4;
groups {
    global;
}
system {
    host-name vSRX-5;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
```

```
            telnet;
            xnm-clear-text;
            web-management {
                http {
                    interface vlan.0;
                }
                https {
                    system-generated-certificate;
                    interface vlan.0;
                }
            }
            dhcp {
                router {
                    192.168.1.1;
                }
                pool 192.168.1.0/24 {
                    address-range low 192.168.1.2 high 192.168.1.254;
                }
                propagate-settings ge-0/0/0.0;
            }
        }
        syslog {
            archive size 100k files 3;
            user * {
                any emergency;
            }
            file messages {
                any critical;
                authorization info;
            }
            file interactive-commands {
                interactive-commands error;
            }
        }
        max-configurations-on-flash 5;
        ##
        ## Warning: statement ignored: unsupported platform (firefly-perimeter)
        ##
        max-configuration-rollbacks 5;
        license {
            autoupdate {
                url https://ae1.juniper.net/junos/key_retrieval;
            }
        }
    }
    interfaces {
        ge-0/0/0 {
            unit 0 {
                family inet {
                    address 172.20.20.1/24;
                }
                family iso;
            }
        }
        ge-0/0/1 {
            unit 0 {
                family inet {
                    address 172.20.33.1/24;
                }
                family iso;
            }
        }
        lo0 {
            unit 0 {
                family inet {
```

```
                address 192.168.3.2/32;
            }
            family iso {
                address 49.0002.0172.0016.0505.00;
            }
        }
    }
}
protocols {
    isis {
        interface ge-0/0/0.0 {
            level 1 disable;
        }
        interface ge-0/0/1.0 {
            level 1 disable;
        }
        interface lo0.0;
    }
}
security {
    forwarding-options {
        family {
            mpls {
                mode packet-based;
            }
            iso {
                mode packet-based;
            }
        }
    }
}
```

Configuration of: vSRX-6

```
## Last changed: 2017-06-01 17:15:02 UTC
version 12.1X47-D15.4;
groups {
    global;
}
system {
    host-name vSRX-6;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
        telnet;
        xnm-clear-text;
        web-management {
            http {
                interface vlan.0;
            }
            https {
                system-generated-certificate;
                interface vlan.0;
            }
        }
        dhcp {
            router {
                192.168.1.1;
            }
            pool 192.168.1.0/24 {
                address-range low 192.168.1.2 high 192.168.1.254;
            }
            propagate-settings ge-0/0/0.0;
        }
    }
    syslog {
        archive size 100k files 3;
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    ##
    ## Warning: statement ignored: unsupported platform (firefly-perimeter)
    ##
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            family inet {
                address 172.20.20.2/24;
            }
            family iso;
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.3.3/32;
            }
            family iso {
                address 49.0002.0172.0016.0705.00;
            }
        }
    }
}
protocols {
    isis {
        interface ge-0/0/0.0 {
            level 1 disable;
        }
        interface lo0.0;
    }
}
security {
    forwarding-options {
        family {
            mpls {
                mode packet-based;
            }
            iso {
                mode packet-based;
                }
            }
        }
}
```

Appendix E

Configuration of: vSRX-1(ABR)

```
## Last changed: 2017-05-24 08:49:54 UTC
version 12.1X47-D15.4;
groups {
    global;
}
system {
    host-name vSRX-1;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
        telnet;
        xnm-clear-text;
        web-management {
            http {
                interface vlan.0;
            }
            https {
                system-generated-certificate;
                interface vlan.0;
            }
        }
        dhcp {
            router {
                192.168.1.1;
            }
            pool 192.168.1.0/24 {
                address-range low 192.168.1.2 high 192.168.1.254;
            }
            propagate-settings ge-0/0/0.0;
        }
    }
    syslog {
        archive size 100k files 3;
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    ##
    ## Warning: statement ignored: unsupported platform (firefly-perimeter)
    ##
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-0/0/1 {
        unit 0 {
            family inet {
                address 172.20.66.1/24;
            }
        }
    }
    ge-0/0/3 {
        unit 0 {
            description to-vSRX-3;
            family inet {
                address 10.0.0.2/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.1.1/32;
            }
        }
    }
}
routing-options {
    router-id 192.168.1.1;
    autonomous-system 22;
}
protocols {
    bgp {
        export [ direct-into-bgp ospf-into-bgp ];
        group external-peers {
            type external;
            peer-as 15;
            allow 0.0.0.0/0;
            neighbor 10.0.0.1;
        }
    }
    ospf {
        export [ bgp-into-ospf direct-into-bgp ];
        area 0.0.0.0 {
            interface lo0.0;
            interface ge-0/0/0.0;
            interface ge-0/0/1.0;
        }
    }
}
policy-options {
    policy-statement bgp-into-ospf {
        term bgp-only {
            from protocol bgp;
            then accept;
        }
    }
    policy-statement direct-into-bgp {
        term direct-only {
            from protocol direct;
            then accept;
        }
    }
    policy-statement ospf-into-bgp {
        term ospf-only {
            from {
                protocol ospf;
                area 0.0.0.0;
            }
            then accept;
        }
    }
}
security {
    forwarding-options {
        family {
            inet6 {
                mode packet-based;
            }
            mpls {
                mode packet-based;
            }
        }
    }
}
```

Configuration of: vSRX-3(ABR)

```
## Last changed: 2017-06-05 08:17:54 UTC
version 12.1X47-D15.4;
groups {
    global;
}
system {
    host-name vSRX-3;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
        telnet;
        xnm-clear-text;
        web-management {
            http {
                interface vlan.0;
            }
            https {
                system-generated-certificate;
                interface vlan.0;
            }
        }
        dhcp {
            router {
                192.168.1.1;
            }
            pool 192.168.1.0/24 {
                address-range low 192.168.1.2 high 192.168.1.254;
            }
            propagate-settings ge-0/0/0.0;
        }
    }
    syslog {
        archive size 100k files 3;
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    ##
    ## Warning: statement ignored: unsupported platform (firefly-perimeter)
    ##
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-0/0/1 {
        unit 0 {
            family inet {
                address 172.20.33.2/24;
            }
        }
    }
    ge-0/0/3 {
        unit 0 {
            description to-vSRX-1;
            family inet {
                address 10.0.0.1/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.3.1/32;
            }
        }
    }
}
routing-options {
    router-id 192.168.3.1;
    autonomous-system 15;
}
protocols {
    bgp {
        export [ send-direct ospf ];
        group external-peers {
            type external;
            peer-as 22;
            neighbor 10.0.0.2;
        }
        group internal-peers {
            type internal;
            description "Connections to vSRX-5 and vSRX-6";
            local-address 192.168.3.1;
            export send-direct;
            neighbor 192.168.3.2;
            neighbor 192.168.3.3;
        }
    }
    ospf {
        area 0.0.0.1 {
            interface lo0.0 {
                passive;
            }
            interface ge-0/0/1.0;
        }
    }
}
policy-options {
    policy-statement bgp {
        term 1 {
            from protocol bgp;
            then accept;
        }
    }
    policy-statement ospf {
        term 3 {
            from protocol ospf;
            then accept;
        }
    }
    policy-statement send-direct {
        term 2 {
            from protocol direct;
            then accept;
        }
    }
}
security {
    forwarding-options {
        family {
            mpls {
                mode packet-based;
            }
            iso {
                mode packet-based;
            }
        }
    }
}
```

Configuration of: vSRX-5

```
## Last changed: 2017-06-05 08:20:21 UTC
version 12.1X47-D15.4;
groups {
    global;
}
system {
    host-name vSRX-5;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
        telnet;
        xnm-clear-text;
        web-management {
            http {
                interface vlan.0;
}
            }
            https {
                system-generated-certificate;
                interface vlan.0;
            }
        }
        dhcp {
            router {
                192.168.1.1;
            }
            pool 192.168.1.0/24 {
                address-range low 192.168.1.2 high 192.168.1.254;
            }
            propagate-settings ge-0/0/0.0;
        }
    }
    syslog {
        archive size 100k files 3;
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    ##
    ## Warning: statement ignored: unsupported platform (firefly-perimeter)
    ##
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            description to-vSRX6;
            family inet {
                address 172.20.20.1/24;
            }
        }
    }
    ge-0/0/1 {
        unit 0 {
            description to-vSRX3;
            family inet {
                address 172.20.33.1/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.3.2/32;
            }
        }
    }
}
```

Configuration of: vSRX-6

```
## Last changed: 2017-06-05 08:20:23 UTC
version 12.1X47-D15.4;
groups {
    global;
}
system {
    host-name vSRX-6;
    root-authentication {
        encrypted-password "$1$JIJjCz84$phS6QY3sgimoMOi/kjFN20"; ## SECRET-DATA
    }
    name-server {
        208.67.222.222;
        208.67.220.220;
    }
    services {
        ssh;
        telnet;
        xnm-clear-text;
        web-management {
            http {
                interface vlan.0;
            }
            https {
                system-generated-certificate;
                interface vlan.0;
            }
        }
        dhcp {
            router {
                192.168.1.1;
            }
            pool 192.168.1.0/24 {
                address-range low 192.168.1.2 high 192.168.1.254;
            }
            propagate-settings ge-0/0/0.0;
        }
    }
    syslog {
        archive size 100k files 3;
        user * {
            any emergency;
        }
        file messages {
            any critical;
            authorization info;
        }
        file interactive-commands {
            interactive-commands error;
        }
    }
    max-configurations-on-flash 5;
    ##
    ## Warning: statement ignored: unsupported platform (firefly-perimeter)
    ##
    max-configuration-rollbacks 5;
    license {
        autoupdate {
            url https://ae1.juniper.net/junos/key_retrieval;
        }
    }
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            description to-vSRX5;
            family inet {
                address 172.20.20.2/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.3.3/32;
            }
        }
    }
}
routing-options {
    router-id 192.168.3.3;
    autonomous-system 15;
}
protocols {
    bgp {
        group internal-peers {
            type internal;
            description "connections to vSRX-5 and vSRX-3";
            local-address 192.168.3.3;
            export send-direct;
            neighbor 192.168.3.2;
            neighbor 192.168.3.1;
        }
    }
    ospf {
        area 0.0.0.1 {
            interface lo0.0 {
                passive;
            }
            interface ge-0/0/0.0;
        }
    }
}
policy-options {
    policy-statement send-direct {
        term 2 {
            from protocol direct;
            then accept;
        }
    }
}
security {
    forwarding-options {
        family {
            mpls {
                mode packet-based;
            }
            iso {
                mode packet-based;
            }
        }
    }
}
```

Appendix F

Configuration of: vSRX-1

```
## Last changed: 2017-06-05 08:45:54 UTC
```

```
version 12.1X47-D15.4;
system {
    host-name vSRX-1;
    root-authentication {
        encrypted-password "$1$3wX27FEA$8/YMqSk092wAbqkexszJ71"; ## SECRET-DATA
        ssh-dsa "ssh-dss
AAAAB3NzaC1kc3MAAACBAMQrfP2bZyBXJ6PC7XXZ+MzErI8Jl6jah5L4/O8BsfP2hC7EvRfNoX7MqbrtCX/9gUH9g
ChVuBCB+ERULMdgRvM5uGhC/gs4UX+4dBbfBgKYYwgmisM8EoT25m7qI8ybpl2YZvHNznvO8h7kr4kpYuQEpKvgsT
dH/Jle4Uqnjv7DAAAAFQDZaqA6QAgbW3O/zveaLCIDj6p0dwAAAIB1iL+krWrXiD8NPpY+w4dWXEqaV3bnobzPC4e
yxQKBUCOr80Q5YBlWXVBHx9elwBWZwj0SF4hLKHznExnLerVsMuTMA846RbQmSz62vM6kGM13HFonWeQvWia0TDr7
8+rOEgWF2KHBSIxL51lmIDW8Gql9hJfD/Dr/NKP97w3L0wAAAIEAr3FkWU8XbYytQYEKxsIN9P1UQ1ERXB3G40Ywq
FO484SlyKyYCfaz+yNsaAJu2C8UebDIR3GieyNcOAKf3inCG8jQwjLvZskuZwrvlsz/xtcxSoAh9axJcdUfSJYMW/
g+mD26JK1Cliw5rwp2nH9kUrJxeI7IReDp4egNkM4i15o= configurator@server1.he"; ## SECRET-DATA
    }
    login {
        user lab {
            uid 2000;
            class super-user;
            authentication {
                encrypted-password "$1$84J5Maes$cni5Hrazbd/IEHr/50oY30"; ## SECRET-DATA
            }
        }
    }
    services {
        ftp;
        ssh;
        telnet;
        web-management {
            http {
                interface fxp0.0;
            }
            https {
                system-generated-certificate;
                interface all;
            }
        }
    }
    syslog {
        file messages {
            any any;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
    ntp {
        boot-server 172.25.11.254;
        server 172.25.11.254;
    }
}
interfaces {
    ge-0/0/0 {
        description "MGMT Interface - DO NOT DELETE";
        unit 0 {
            family inet {
                address 172.25.11.1/24;
            }
        }
    }
    ge-0/0/1 {
        disable;
        unit 0 {
            family inet {
                address 172.20.77.1/30;
            }
            family inet6 {
                address fdaa:dead:beef:1::1/127;
            }
        }
    }
    ge-0/0/2 {
        unit 0 {
            family inet {
                address 172.20.66.1/30;
            }
            family inet6 {
                address fdaa:dead:beef:2::1/127;
            }
        }
    }
    ge-0/0/3 {
        unit 0 {
            family inet {
                address 172.18.1.2/30;
            }
        }
    }
    ge-0/0/4 {
        unit 0 {
            family inet {
                filter {
                    input classify-traffic;
                }
                address 172.20.101.1/24;
            }
        }
    }
    ge-0/0/5 {
        unit 0 {
            family inet {
                filter {
                    input classify-traffic;
                }
                address 172.20.201.1/24;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 192.168.1.1/32;
            }
            family inet6 {
                address fdaa:dead:beef:9::1/128;
            }
        }
    }
}
routing-options {
    static {
        route 0.0.0.0/0 next-hop 172.18.1.1;
        route 192.168.1.2/32 next-hop 172.20.101.10;
        route 172.21.0.0/24 next-hop 172.20.101.10;
        route 172.21.1.0/24 next-hop 172.20.101.10;
        route 172.21.2.0/24 next-hop 172.20.101.10;
        route 192.168.1.3/32 next-hop 172.20.201.10;
    }
}
protocols {
    ospf {
        export ospf-export;
        area 0.0.0.0 {
            interface ge-0/0/1.0;
            interface ge-0/0/2.0;
            interface lo0.0;
        }
    }
    ospf3 {
        area 0.0.0.0 {
            interface lo0.0;
            interface ge-0/0/1.0;
            interface ge-0/0/2.0;
        }
    }
}
policy-options {
    policy-statement ospf-export {
        term match-default-static-route {
            from {
                protocol static;
                route-filter 0.0.0.0/0 exact;
            }
            then accept;
        }
        term match-interface-routes {
            from {
                route-filter 172.18.1.0/30 exact;
                route-filter 172.20.101.0/24 exact;
                route-filter 172.20.201.0/24 exact;
            }
            then accept;
        }
        term match-other-static-routes {
            from {
                protocol static;
                route-filter 172.21.0.0/24 exact;
                route-filter 172.21.1.0/24 exact;
                route-filter 172.21.2.0/24 exact;
            }
            then accept;
        }
    }
}
class-of-service {
    forwarding-classes {
        queue 1 admin;
        queue 2 voip;
    }
    interfaces {
        ge-0/0/2 {
            scheduler-map my-sched-map;
            unit 0 {
                rewrite-rules {
                    inet-precedence default;
                }
            }
        }
        ge-0/0/3 {
            scheduler-map my-sched-map;
        }
        ge-0/0/4 {
            scheduler-map my-sched-map;
        }
        ge-0/0/5 {
            scheduler-map my-sched-map;
        }
    }
    scheduler-maps {
        my-sched-map {
            forwarding-class best-effort scheduler best-effort-sched;
            forwarding-class admin scheduler admin-sched;
            forwarding-class voip scheduler voip-sched;
            forwarding-class network-control scheduler network-control-sched;
        }
    }
    schedulers {
        best-effort-sched {
            transmit-rate percent 40;
            buffer-size percent 40;
            priority low;
        }
        admin-sched {
            transmit-rate percent 45;
            buffer-size percent 45;
            priority medium-low;
        }
        voip-sched {
            transmit-rate percent 10;
            buffer-size percent 10;
            priority high;
        }
        network-control-sched {
            transmit-rate percent 5;
            buffer-size percent 5;
            priority medium-high;
        }
    }
}
security {
    forwarding-options {
        family {
            inet6 {
                mode packet-based;
            }
            mpls {
                mode packet-based;
            }
        }
    }
}
firewall {
    family inet {
        filter classify-traffic {
            term sip {
                from {
                    source-address {
                        172.20.101.0/24;
                    }
                    protocol [ tcp udp ];
                    port 5060;
                }
                then {
                    forwarding-class voip;
                    accept;
                }
            }
            term rtp {
                from {
                    source-address {
                        172.20.101.0/24;
                    }
                    protocol udp;
                    port 16384-32767;
                }
                then {
                    forwarding-class voip;
                    accept;
                }
            }
            term admin {
                from {
                    source-address {
                        172.20.201.0/24;
                    }
                }
                then {
                    forwarding-class admin;
                    accept;
                }
            }
            term accept-all {
                then accept;
            }
        }
    }
}
```

Configuration of: vSRX-2

```
## Last changed: 2017-06-05 08:45:52 UTC
version 12.1X47-D15.4;
system {
    host-name vSRX-2;
    root-authentication {
        encrypted-password "$1$rUleI7B.$lzOAmvzh2qAcDdLXri2Cb/"; ## SECRET-DATA
        ssh-dsa "ssh-dss
AAAAB3NzaC1kc3MAAACBAMQrfP2bZyBXJ6PC7XXZ+MzErI8Jl6jah5L4/O8BsfP2hC7EvRfNoX7MqbrtCX/9gUH9g
ChVuBCB+ERULMdgRvM5uGhC/gs4UX+4dBbfBgKYYwgmisM8EoT25m7qI8ybpl2YZvHNznvO8h7kr4kpYuQEpKvgsT
dH/Jle4Uqnjv7DAAAAFQDZaqA6QAgbW3O/zveaLCIDj6p0dwAAAIB1iL+krWrXiD8NPpY+w4dWXEqaV3bnobzPC4e
yxQKBUCOr80Q5YBlWXVBHx9elwBWZwj0SF4hLKHznExnLerVsMuTMA846RbQmSz62vM6kGM13HFonWeQvWia0TDr7
8+rOEgWF2KHBSIxL51lmIDW8Gql9hJfD/Dr/NKP97w3L0wAAAIEAr3FkWU8XbYytQYEKxsIN9P1UQ1ERXB3G40Ywq
FO484SlyKyYCfaz+yNsaAJu2C8UebDIR3GieyNcOAKf3inCG8jQwjLvZskuZwrvlsz/xtcxSoAh9axJcdUfSJYMW/
g+mD26JK1Cliw5rwp2nH9kUrJxeI7IReDp4egNkM4i15o= configurator@server1.he"; ## SECRET-DATA
    }
    login {
        user lab {
            uid 2000;
            class super-user;
            authentication {
                encrypted-password "$1$84J5Maes$cni5Hrazbd/IEHr/50oY30"; ## SECRET-DATA
            }
        }
    }
    services {
        ftp;
        ssh;
        telnet;
        web-management {
            http {
                interface ge-0/0/0.0;
            }
            https {
                system-generated-certificate;
                interface all;
            }
        }
    }
    syslog {
        file messages {
            any any;
            authorization info;
        }
        file interactive-commands {
            interactive-commands any;
        }
    }
    ntp {
        boot-server 172.25.11.254;
        server 172.25.11.254;
    }
}
interfaces {
    ge-0/0/0 {
        description "MGMT Interface - DO NOT DELETE";
        unit 0 {
            family inet {
                address 172.25.11.2/24;
            }
        }
    }
    lt-0/0/0 {
        per-unit-scheduler;
        unit 10 {
            encapsulation frame-relay;
            dlci 100;
            peer-unit 20;
            family inet {
                address 172.31.15.2/30;
            }
        }
        unit 20 {
            encapsulation frame-relay;
            dlci 100;
            peer-unit 10;
            family inet {
                address 172.31.15.1/30;
            }
        }
        unit 30 {
            encapsulation frame-relay;
            dlci 200;
            peer-unit 40;
            family inet {
                address 172.18.2.1/30;
            }
        }
        unit 40 {
            encapsulation frame-relay;
            dlci 200;
            peer-unit 30;
            family inet {
                address 172.18.2.2/30;
            }
        }
    }
    ge-0/0/1 {
        unit 0 {
            family inet {
                address 172.20.77.2/30;
            }
            family inet6 {
                address fdaa:dead:beef:1::0/127;
            }
        }
    }
    ge-0/0/2 {
        unit 0 {
            family inet {
                address 172.20.66.2/30;
            }
            family inet6 {
                address fdaa:dead:beef:2::0/127;
            }
        }
    }
    ge-0/0/3 {
        unit 0 {
            family inet {
                address 172.18.1.1/30;
            }
        }
    }
    ge-0/0/4 {
        unit 0 {
            family inet {
                address 172.20.101.10/24;
            }
        }
    }
    ge-0/0/5 {
        vlan-tagging;
        unit 1 {
            vlan-id 1;
            family inet {
                address 172.21.0.1/24;
            }
        }
        unit 2 {
            vlan-id 2;
            family inet {
                address 172.21.1.1/24;
            }
        }
        unit 3 {
            vlan-id 3;
            family inet {
                address 172.21.2.1/24;
            }
        }
    }
    ge-0/0/6 {
        unit 0 {
            family inet {
                address 172.20.201.10/24;
            }
        }
    }
    ge-0/0/7 {
        unit 0 {
            family inet {
                address 172.20.102.1/24;
            }
        }
    }
    ge-0/0/8 {
        unit 0 {
            family inet {
                address 172.20.102.10/24;
            }
        }
    }
    lo0 {
        unit 0 {
            description "loopback for main routing instance of vSRX-2";
            family inet {
                address 192.168.2.1/32;
            }
            family inet6 {
                address fdaa:dead:beef:9::2/128;
            }
        }
        unit 1 {
            description "vr101 loopback";
            family inet {
                address 192.168.1.2/32;
            }
        }
        unit 2 {
            description "vr102 loopback";
            family inet {
                address 192.168.2.2/32;
            }
        }
        unit 3 {
            description "VR_Internet Loopback";
            family inet {
                address 192.168.3.1/32;
            }
        }
        unit 4 {
            description "VR_Host Loopback";
            family inet {
                address 192.168.4.1/32;
            }
        }
        unit 5 {
            description "vr201 loopback";
            family inet {
                address 192.168.1.3/32;
            }
        }
    }
}
routing-options {
    static {
        route 0.0.0.0/0 next-hop 172.18.2.1;
        route 192.168.1.2/32 {
            next-hop 172.20.66.1;
            qualified-next-hop 172.20.77.1 {
                preference 6;
            }
        }
        route 192.168.1.1/32 {
            next-hop 172.20.66.1;
            qualified-next-hop 172.20.77.1 {
                preference 6;
            }
        }
        route 172.20.101.0/24 {
            next-hop 172.20.66.1;
            qualified-next-hop 172.20.77.1 {
                preference 6;
            }
        }
        route 192.168.2.2/32 next-hop 172.20.102.10;
        route 192.168.1.3/32 {
            next-hop 172.20.66.1;
            qualified-next-hop 172.20.77.1 {
                preference 6;
            }
        }
        route 172.20.201.0/24 {
            next-hop 172.20.66.1;
            qualified-next-hop 172.20.77.1 {
                preference 6;
            }
        }
    }
}
protocols {
    ospf {
        area 0.0.0.0 {
            interface ge-0/0/1.0;
            interface ge-0/0/2.0;
            interface lo0.0;
            interface ge-0/0/7.0;
        }
    }
    ospf3 {
        area 0.0.0.0 {
            interface lo0.0;
            interface ge-0/0/1.0;
            interface ge-0/0/2.0;
        }
    }
}
class-of-service {
    forwarding-classes {
        queue 1 admin;
        queue 2 voip;
    }
    interfaces {
        ge-0/0/* {
            scheduler-map my-sched-map;
        }
        ge-0/0/2 {
            scheduler-map my-sched-map;
            unit 0 {
                classifiers {
                    inet-precedence default;
                }
            }
        }
    }
    scheduler-maps {
        my-sched-map {
            forwarding-class best-effort scheduler best-effort-sched;
            forwarding-class admin scheduler admin-sched;
            forwarding-class voip scheduler voip-sched;
            forwarding-class network-control scheduler network-control-sched;
        }
    }
    schedulers {
        class-of-service;
        best-effort-sched {
            transmit-rate percent 40;
            buffer-size percent 40;
            priority low;
        }
        admin-sched {
            transmit-rate percent 45;
            buffer-size percent 45;
            priority medium-low;
        }
        voip-sched {
            transmit-rate percent 10;
            buffer-size percent 10;
            priority high;
        }
        network-control-sched {
            transmit-rate percent 5;
            buffer-size percent 5;
            priority medium-high;
        }
    }
}
security {
    forwarding-options {
        family {
            inet6 {
                mode packet-based;
            }
            mpls {
                mode packet-based;
            }
        }
    }
}
routing-instances {
    VR_Host {
        instance-type virtual-router;
        interface lt-0/0/0.20;
        interface lo0.4;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop 172.31.15.2;
            }
        }
    }
    VR_Internet {
        instance-type virtual-router;
        interface lt-0/0/0.10;
        interface lt-0/0/0.30;
        interface ge-0/0/3.0;
        interface lo0.3;
        routing-options {
            static {
                route 192.168.1.1/32 next-hop 172.18.1.2;
                route 192.168.1.2/32 next-hop 172.18.1.2;
                route 192.168.2.1/32 next-hop 172.18.2.2;
                route 192.168.2.2/32 next-hop 172.18.2.2;
                route 172.20.101.0/24 next-hop 172.18.1.2;
                route 172.20.102.0/24 next-hop 172.18.2.2;
            }
        }
    }
    vr101 {
        instance-type virtual-router;
        interface ge-0/0/4.0;
        interface ge-0/0/5.1;
        interface ge-0/0/5.2;
        interface ge-0/0/5.3;
        interface lo0.1;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop 172.20.101.1;
            }
        }
        protocols {
            ospf {
                area 0.0.0.0 {
                    interface lo0.1;
                    interface ge-0/0/4.0;
                }
            }
        }
    }
    vr102 {
        instance-type virtual-router;
        interface ge-0/0/8.0;
        interface lo0.2;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop 172.20.102.1;
            }
        }
        protocols {
            ospf {
                area 0.0.0.0 {
                    interface lo0.2;
                    interface ge-0/0/8.0;
                }
            }
        }
    }
    vr201 {
        instance-type virtual-router;
        interface ge-0/0/6.0;
        interface lo0.5;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop 172.20.201.1;
            }
        }
    }
}
```