BSc in Telecommunications Engineering

TEL3214

Computer Communication Networks

Lecture 07a
Routing in IPv4 with OSPFv2

Eng Diarmuid O'Briain, CEng, CISSP

Department of Electrical and Computer Engineering,
College of Engineering, Design, Art and Technology,
Makerere University

Table of Contents

Illustration Index

This page is intentionally blank

1. Introduction to Routing

Routing refers to selection of paths in a computer network along which to send data. Routing directs forwarding, the passing of logically addressed packets from their source network, toward their ultimate destination through intermediary nodes; typically hardware devices called routers. The routing process usually directs forwarding on the basis of routing tables which maintain a record of the best routes to various network destinations. Thus constructing routing tables, which are held in the routers' memory, becomes very important for efficient routing.

Routing differs from bridging in its assumption that address-structures imply the proximity of similar addresses within the network, thus allowing a single routing-table entry to represent the route to a group of addresses. Therefore, routing outperforms bridging in large networks, and it has become the dominant form of path-discovery on the Internet.

Small networks may involve manually configured routing tables, while larger networks involve complex topologies and may change constantly, making the manual construction of routing tables very problematic. A good example is the Public Switched Telephone Network (PSTN) which uses pre-computed routing tables, with fallback routes if the most direct route becomes blocked. Dynamic routing attempts to solve this problem by constructing routing tables automatically, based on information carried by routing protocols, and allowing the network to act nearly autonomously in avoiding network failures and blockages. Dynamic routing dominates the Internet. However, the configuration of the routing protocols often requires a skilled touch, one should not suppose that networking technology has developed to the point of the complete automation of routing.

Traditional Internet Protocol (IP) routing stays relatively simple because it uses next-hop routing where the router only needs to consider where it sends the packet, and does not need to consider the subsequent path of the packet on the remaining hops. However, more complex routing strategies can be, and are, often used in systems such as Multi-Protocol Label Switching (MPLS), Asynchronous Transfer Mode (ATM) or Frame Relay, which are sometimes used as underlying technologies to support IP networks.

1.1 Standard Routing Model

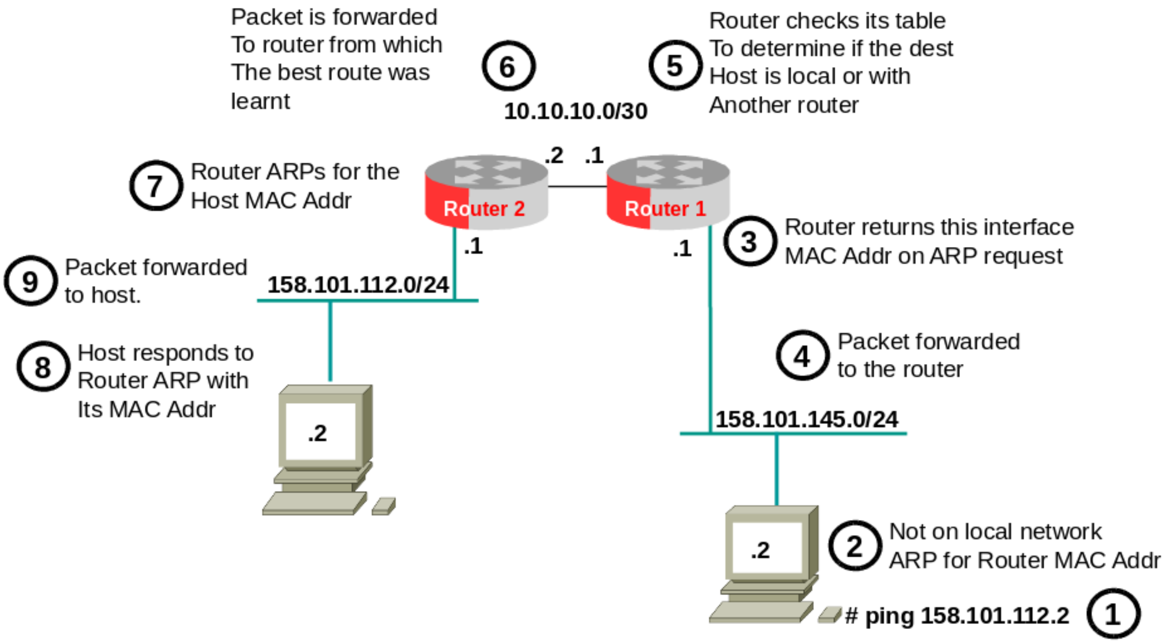

Packet is forwarded To router from which The best route was learnt
6
5
Router checks its table To determine if the dest Host is local or with Another router
10.10.10.0/30
.2
.1
7
Router ARPs for the Host MAC Addr
Router 2
Router 1
.1
.1
3
Router returns this interface MAC Addr on ARP request
9
Packet forwarded to host.
158.101.112.0/24
4
Packet forwarded to the router
8
Host responds to Router ARP with Its MAC Addr
.2
158.101.145.0/24
.2
2
Not on local network ARP for Router MAC Addr
# ping 158.101.112.2
1

Illustration 1: Standard Routing model

A router must be connected to at least two networks, or it will have nothing to route. A special variety of router is the one-armed router used to route packets in a VLAN environment. In the case of a one-armed router the multiple attachments to different networks are all over the same physical link.

A router creates and/or maintains a table, called a "routing table" that stores the best routes to certain network destinations and the "routing metrics" associated with those routes.

Knowing where to send packets requires knowledge of the structure of the network. In small networks, routing can be very simple, and is often configured by hand called Static Routing. In large networks the topology of the network is complex, and constantly changing, making the problem of constructing the routing tables very complex.

As the best routes can only be recalculated very slowly relative to the rate of arrival of packets, routers keep a routing table that maintains a record of only the best possible routes to certain network destinations and the routing metrics associated with those routes.

In the above diagram the host 158.101.145.2 pings 158.101.112.2. The IP Stack in the host determines that this address is not on its own network by doing a bitwise AND on the address with its own subnet mask. It then looks up its Default Gateway, or in other words the router connected to the LAN which can route to the rest of the internetwork. It sends out an Address Resolution Protocol (ARP) request to 158.101.145.1 for the MAC address of the routers interface on its own network. The router responds in kind and the host forwards the packet in a frame to the router.

The router strips off the frame and does a routing table lookup for the destination IP address in the packet header. In this case it does not have an interface in the destination network. It determines from its routing table that router2 has declared it can reach the destination network. It places the packet in a frame and forwards it to the peer router2.

Router2 receives the frame and removes the packet from it. It determines that it is connected on its other interface to the destination network so it does an ARP request on that LAN for the MAC address of the host 158.101.112.2. The host responds to the request with its MAC address and router2 encapsulates the packet in a new frame which it forwards to the destination host.

1.2 Routing Tables

Routers maintain Routing Tables to determine if it can reach a requested route. Routing tables can take many forms, but here is a simple model that can explain most Internet routing. Each entry in a routing table has at least two fields - IP Address Prefix and Next Hop. The Next Hop is the IP address of another host or router that is directly reachable via an Ethernet, serial link, or some other physical connection. The IP Address Prefix specifies a set of destinations for which the routing entry is valid for. In order to be in this set, the beginning of the destination IP address must match the IP Address Prefix, which can have from 0 to 32 significant bits. For example, a IP Address Prefix of 128.8.0.0/16 would match any IP Destination Address of the form 128.8.X.X. Bridged and switched networks are regarded as single connections.

If no routing table entries match a packet's Destination Address, the packet is discarded as undeliverable (possibly with an ICMP notification to the sender). If multiple routing tables entries match, the longest match is preferred. The longest match is the entry with the most 1 bits in its Routing Mask.

2. Quagga

2.1 Introduction

Quagga Routing Software Suite is a GNU General Public License (GPL) licensed advanced routing software package that provides a suite of TCP/IP based routing protocols.

- RIP
- RIPng
- OSPFv2
- OSPFv3
- Babel
- BGP-4

The Quagga architecture consists of a core routing daemon, *zebra* and a number of routing protocol daemons. *zebra* itself is the kernel interface, handles static routes and is the zserv FTP server for the transferring of zebra files. Each routing protocol has its own daemon, with for example *opdfd* daemon handling OSPFv2 operations and *ospf6d* the OPSFv3 operations.

Having mastered the application of IP Networking on a GNU/Linux OS in the role of a single host on a network we will now look at the use of Quagga to allow the GNU/Linux host with multiple network interfaces to act as a router in a single area OSPF network for both IPv4 and IPv6.

3. Introduction to DHCP

Dynamic Host Configuration Protocol (DHCP) is a mechanism for the automatic configuration of hosts. DHCP is an extension of the original autoconfiguration protocol called the Bootstrap Protocol (BOOTP) and elements of BOOTP still exist within DHCP.

The DHCP protocol allows a host to contact a central server which maintains a list of IP addresses which may be assigned on one or more subnets. A DHCP client may request an address from this pool and also get other details about the network to which it is attached, such as the address of the gateway router, the address of the DNS Server, etc..

There are two versions of the DHCP protocol DHCPv4 and DHCPv6 though IPv6 has and alternative mechanism for auto-configuration called StateLess Address Auto Configuration (SLAAC) which will be considered later in this lecture.

3.1 DHCP flow

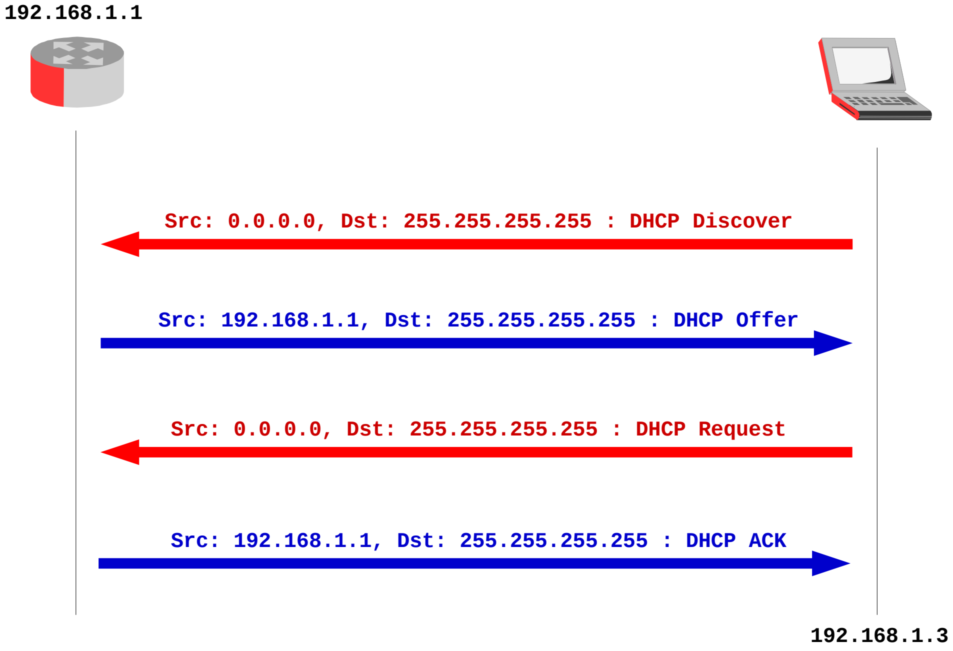

192.168.1.1
Src: 0.0.0.0, Dst: 255.255.255.255 : DHCP Discover
Src: 192.168.1.1, Dst: 255.255.255.255 : DHCP Offer
Src: 0.0.0.0, Dst: 255.255.255.255 : DHCP Request
Src: 192.168.1.1, Dst: 255.255.255.255 : DHCP ACK
192.168.1.3

Illustration 2: DHCP Flow

A DHCP **Discover** is broadcast from the host requiring an IP address. The DHCP Server responds with an **Offer**. If the client is happy it sends another broadcast **Request** for the IP address from the offer. The DHCP Server sends an **ACK**nowledgement along with the optional elements like DNS Server and Gateway router address.

3.2 DHCP Lab

- Run the network: **TEL3214-DHCP-Example.imn** in the NTE emulator.
- Run Wireshark on Host n4.
- From the bash prompt on Host n2:

```
root@n2:/tmp/pycore.57892/n2.conf# dhclient -v eth0
Internet Systems Consortium DHCP Client 4.3.1
Copyright 2004-2014 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/00:00:00:aa:00:01
Sending on   LPF/eth0/00:00:00:aa:00:01
Sending on   Socket/fallback
DHCPREQUEST on eth0 to 255.255.255.255 port 67
DHCPACK from 192.168.1.1
bound to 192.168.1.3 -- renewal in 293 seconds.
```

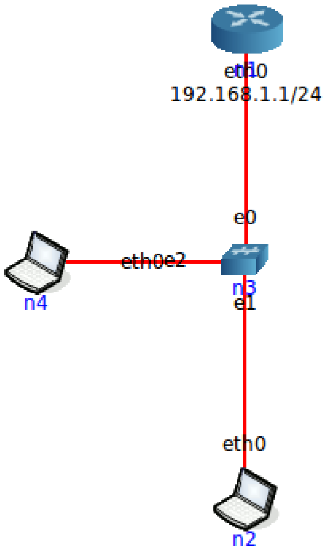
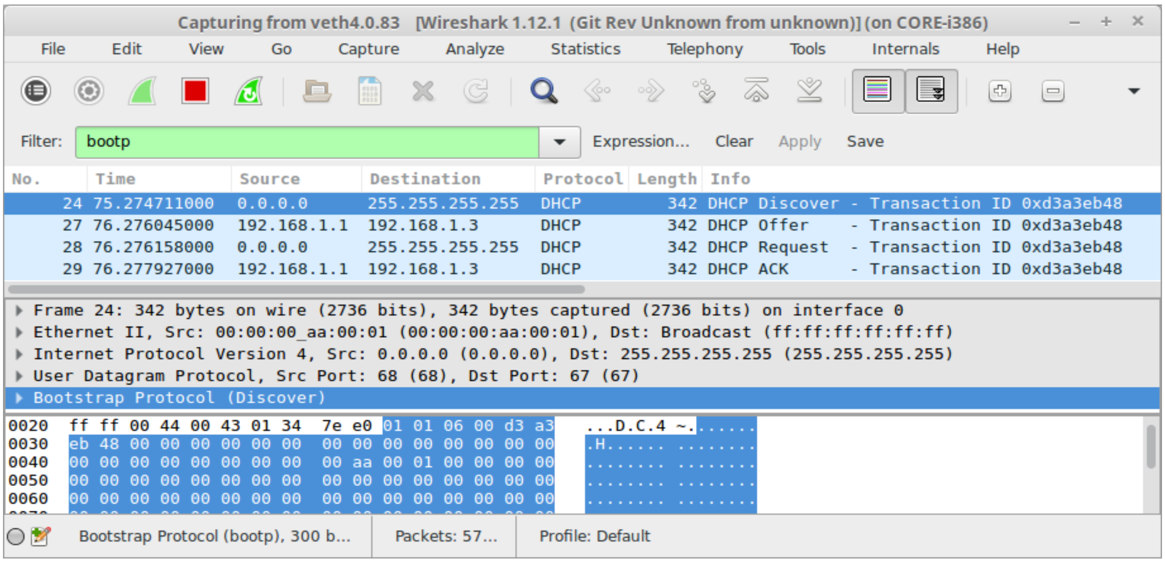


Illustration 3: DHCP Lab

- Spend some time understanding the packets in the trace.



Filter: bootp

```
No.   Time           Source        Destination      Protocol Length Info
   24 75.274711000   0.0.0.0       255.255.255.255  DHCP        342 DHCP Discover - Transaction ID 0xd3a3eb48
   27 76.276045000   192.168.1.1   192.168.1.3      DHCP        342 DHCP Offer    - Transaction ID 0xd3a3eb48
   28 76.276158000   0.0.0.0       255.255.255.255  DHCP        342 DHCP Request  - Transaction ID 0xd3a3eb48
   29 76.277927000   192.168.1.1   192.168.1.3      DHCP        342 DHCP ACK      - Transaction ID 0xd3a3eb48

▸ Frame 24: 342 bytes on wire (2736 bits), 342 bytes captured (2736 bits) on interface 0
▸ Ethernet II, Src: 00:00:00_aa:00:01 (00:00:00:aa:00:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▸ Internet Protocol Version 4, Src: 0.0.0.0 (0.0.0.0), Dst: 255.255.255.255 (255.255.255.255)
▸ User Datagram Protocol, Src Port: 68 (68), Dst Port: 67 (67)
▸ Bootstrap Protocol (Discover)

0020  ff ff 00 44 00 43 01 34  7e e0 01 01 06 00 d3 a3   ...D.C.4 ~.......
0030  eb 48 00 00 00 00 00 00  00 00 00 00 00 00 00 00   .H...... ........
0040  00 00 00 00 00 00 00 00  00 aa 00 01 00 00 00 00   ........ ........
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
```

Bootstrap Protocol (bootp), 300 b... | Packets: 57... | Profile: Default

Illustration 4: DHCP Lab: Wireshark trace

4. Build an IPv4 routed network

Here is an IPv4 addressed network with 2 routers. Configure for OSPF.

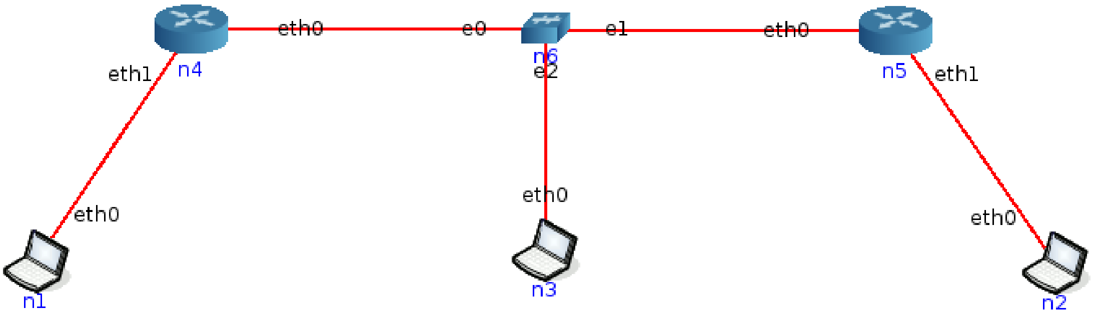


Illustration 5: IPv4 Routed network

Start the network and then select the *vtysh* shell by right-click on the ***icon > Shell window > bash***. At the bash prompt type **vtysh**. (Starting vtysh directly has a pager problem).

```
root@n4:/tmp/pycore.42270/n4.conf# vtysh

Hello, this is Quagga (version 0.99.23.1).
Copyright 1996-2005 Kunihiro Ishiguro, et al.

n4#
```

For the purpose of the exercise the link between routers n4 and n5 will be configured with the network 10.10.10.0/30 and a local interface address in the range 10.0.0.0/8.

4.1 The basics

The sh run command gives you the configuration running on the device at that point in time.

```
n4# show run
Building configuration...

Current configuration:
!
!
service integrated-vtysh-config
!
interface eth0
 ipv6 nd suppress-ra
!
interface eth1
 ipv6 nd suppress-ra
!
interface lo
!
router ospf
!
ip forwarding
ipv6 forwarding
!
line vty
!
end
```

The ***copy running-config startup-config*** (can be abbreviated to ***copy run start***) saves the running configuration to file. However while this is exactly how you save configurations on real Quagga routers, within the NTE simulator this overwrites the /etc/quagga/Quagga.conf file and will mess up future configurations. So if a configuration needs to be saved on NTE use the ***File > Save as imn...*** option.

```
n4# copy run start
Building Configuration...
Integrated configuration saved to /etc/quagga/Quagga.conf
[OK]
```

If this command was in fact done, and Quagga.conf have been written to, NTE can be recovered by removing the file and creating a new soft link as follows.

```
nte@NTE-i386:~$ cd /etc/quagga
nte@NTE-i386:/etc/quagga$ sudo rm Quagga.conf
nte@NTE-i386:/etc/quagga$ sudo ln -s /usr/local/etc/quagga/Quagga.conf
Quagga.conf
```

To configure a router it is necessary to change into a configuration terminal. This is achieved with the ***configure terminal*** command (can be abbreviated to ***conf t***).

```
n4# conf t
n4(config)#

The first action is alyays to set the hostname. This is a simple identifier
for the device and will appear in the prompt.

n4(config)# hostname RTR_n4
RTR_n4(config)#
```

4.2 Configure Router n4

Configure interface eth0 with the IP address 10.0.0.1/30. Complete with a ***no shutdown*** command (can be abbreviated to ***no shut***).

```
RTR_n4(config)# interface eth0
RTR_n4(config-if)# ip address 10.10.10.1/30
RTR_n4(config-if)# no shut
```

Now configure eth1 in the same manner. Note the abbreviations used from the previous example.

```
RTR_n4(config-if)# int eth1
RTR_n4(config-if)# ip addr 192.168.1.1/24
RTR_n4(config-if)# no shut
```

Configure an IP address on the local interface. This is good practice and it can be used as the router ID in various places.

```
RTR_n4(config-if)# int lo
RTR_n4(config-if)# ip addr 10.0.0.1/32
RTR_n4(config-if)# exit
```

Now review the configuration thus far. Exit from the configuration mode and review.

```
RTR_n4(config)# exit
RTR_n4# show run
Building configuration...

Current configuration:
!
hostname RTR_n4
!
service integrated-vtysh-config
!
interface eth0
 ip address 10.10.10.1/24
 ipv6 nd suppress-ra
!
interface eth1
 ip address 192.168.1.1/24
 ipv6 nd suppress-ra
!
interface lo
 ip address 10.0.0.1/32
!
router ospf
!
ip forwarding
ipv6 forwarding
!
line vty
!
end
```

4.3 DHCP Server on Router n4

Configure the Dynamic Host Configuration Protocol (DHCP) Server. Before starting the network right click on Router n4 and select ***Services...***

Under the ***Utility*** column click ***DHCP*** and the spanner symbol to the right of it. Add the following to the end of the text and click ***Apply***

```
subnet 192.168.1.0 netmask 255.255.255.0 {
    pool {
        range 192.168.1.2 192.168.1.254;
        default-lease-time 600;
        option routers 192.168.1.1;
        option domain-name-servers 8.8.8.8;
    }
}
```

4.4 Configure Router n5

Following the same logic configure Router n5.

```
n5(config)# hostname RTR_n5
RTR_n5(config)# int eth0
RTR_n5(config-if)# ip addr 10.10.10.2/30
RTR_n5(config-if)# no shut
RTR_n5(config-if)# int eth1
RTR_n5(config-if)# ip addr 192.168.2.1/24
RTR_n5(config-if)# no shut
RTR_n5(config-if)# int lo
RTR_n5(config-if)# ip addr 10.0.0.2/32
RTR_n5(config-if)# no shut
```

Review the configuration

```
RTR_n5# show run
Building configuration...

Current configuration:
!
hostname RTR_n5
!
service integrated-vtysh-config
!
interface eth0
 ip address 10.10.10.2/30
 ipv6 nd suppress-ra
!
interface eth1
 ip address 192.168.2.1/24
 ipv6 nd suppress-ra
!
interface lo
 ip address 10.0.0.2/32
!
router ospf
!
ip forwarding
ipv6 forwarding
!
line vty
!
end
```

4.5 DHCP Server on Router n5

Carry out the same action for Router n5.

```
subnet 192.168.2.0 netmask 255.255.255.0 {
    pool {
        range 192.168.2.2 192.168.2.254;
        default-lease-time 600;
        option routers 192.168.2.1;
        option domain-name-servers 8.8.8.8;
    }
}
```

4.5.1 Confirm DHCP Server is running

Start the network and verify the DHCP Servers are running. Open a bash shell and use the following command.

```
root@n4:/tmp/pycore.39622/n4.conf# service isc-dhcp-server status
Status of ISC DHCP server: dhcpd is running.
```

If it is not running:

```
root@n4:/tmp/pycore.39622/n4.conf# service isc-dhcp-server status
Status of ISC DHCP server: dhcpd is not running.
```

issue the command:

```
root@n4:/tmp/pycore.39622/n4.conf# service isc-dhcp-server status
Starting ISC DHCP server: dhcpd.
```

Note: Using the command **`systemctl status isc-dhcp-server`** fails with a D-Bus connection error. This is due to the way CORE builds LinuX Containers (LXC).

4.5.2 Check the clients have an IP Address

If the client Hosts n1 and n2 do not have IP addresses trigger requests to the servers for IP addresses.

```
root@n1:/tmp/pycore.39622/n1.conf# dhclient -v eth0
Internet Systems Consortium DHCP Client 4.3.1
Copyright 2004-2014 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/00:00:00:aa:00:06
Sending on   LPF/eth0/00:00:00:aa:00:06
Sending on   Socket/fallback
DHCPREQUEST on eth0 to 255.255.255.255 port 67
DHCPACK from 192.168.1.1
RTNETLINK answers: File exists
bound to 192.168.1.3 -- renewal in 221 seconds.
```

4.5.3 Review the packets on the wire at Hub n6

DHCP Discover

```
Frame: 342 bytes on wire (2736 bits) on interface 0
Ethernet II, Src: 00:00:00_aa:00:06, Dst: ff:ff:ff:ff:ff:ff
Internet Protocol Version 4, Src: 0.0.0.0, Dst: 255.255.255.255
User Datagram Protocol, Src Port: 68, Dst Port: 67
Bootstrap Protocol (Discover)
    Message type: Boot Request (1)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x3833224d
    Seconds elapsed: 48
    Bootp flags: 0x0000 (Unicast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 0.0.0.0
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: 00:00:00_aa:00:06
    Client hardware address padding: 00000000000000000000
    Server host name not given
    Boot file name not given
    Magic cookie: DHCP
    Option: (53) DHCP Message Type (Discover)
        Length: 1
        DHCP: Discover (1)
    Option: (12) Host Name
        Length: 2
        Host Name: n1
    Option: (55) Parameter Request List
        Length: 13
        Parameter Request List Item: (1) Subnet Mask
        Parameter Request List Item: (28) Broadcast Address
        Parameter Request List Item: (2) Time Offset
        Parameter Request List Item: (3) Router
        Parameter Request List Item: (15) Domain Name
        Parameter Request List Item: (6) Domain Name Server
        Parameter Request List Item: (119) Domain Search
        Parameter Request List Item: (12) Host Name
        Parameter Request List Item: (44) NetBIOS over TCP/IP Name Server
        Parameter Request List Item: (47) NetBIOS over TCP/IP Scope
        Parameter Request List Item: (26) Interface MTU
        Parameter Request List Item: (121) Classless Static Route
        Parameter Request List Item: (42) Network Time Protocol Servers
    Option: (255) End
        Option End: 255
    Padding
```

DHCP Offer

```
Frame 17: 342 bytes on wire (2736 bits)on interface 0
Ethernet II, Src: 00:00:00_aa:00:05, Dst: 00:00:00_aa:00:06
Internet Protocol Version 4, Src: 192.168.1.1, Dst: 192.168.1.3
User Datagram Protocol, Src Port: 67, Dst Port: 68
Bootstrap Protocol (Offer)
    Message type: Boot Reply (2)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x5b7d5971
    Seconds elapsed: 0
    Bootp flags: 0x0000 (Unicast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 192.168.1.3
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: 00:00:00_aa:00:06
    Client hardware address padding: 00000000000000000000
    Server host name not given
    Boot file name not given
    Magic cookie: DHCP
    Option: (53) DHCP Message Type (Offer)
        Length: 1
        DHCP: Offer (2)
    Option: (54) DHCP Server Identifier
        Length: 4
        DHCP Server Identifier: 192.168.1.1
    Option: (51) IP Address Lease Time
        Length: 4
        IP Address Lease Time: (600s) 10 minutes
    Option: (1) Subnet Mask
        Length: 4
        Subnet Mask: 255.255.255.0
    Option: (3) Router
        Length: 4
        Router: 192.168.1.1
    Option: (6) Domain Name Server
        Length: 4
        Domain Name Server: 8.8.8.8
    Option: (255) End
        Option End: 255
    Padding
```

DHCP Request

```
Frame: 342 bytes on wire (2736 bits) on interface 0
Ethernet II, Src: 00:00:00_aa:00:06, Dst: ff:ff:ff:ff:ff:ff
Internet Protocol Version 4, Src: 0.0.0.0, Dst: 255.255.255.255
User Datagram Protocol, Src Port: 68, Dst Port: 67
Bootstrap Protocol (Request)
    Message type: Boot Request (1)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x5b7d5971
    Seconds elapsed: 0
    Bootp flags: 0x0000 (Unicast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 0.0.0.0
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: 00:00:00_aa:00:06
    Client hardware address padding: 00000000000000000000
    Server host name not given
    Boot file name not given
    Magic cookie: DHCP
    Option: (53) DHCP Message Type (Request)
        Length: 1
        DHCP: Request (3)
    Option: (54) DHCP Server Identifier
        Length: 4
        DHCP Server Identifier: 192.168.1.1
    Option: (50) Requested IP Address
        Length: 4
        Requested IP Address: 192.168.1.3
    Option: (12) Host Name
        Length: 2
        Host Name: n1
    Option: (55) Parameter Request List
        Length: 13
        Parameter Request List Item: (1) Subnet Mask
        Parameter Request List Item: (28) Broadcast Address
        Parameter Request List Item: (2) Time Offset
        Parameter Request List Item: (3) Router
        Parameter Request List Item: (15) Domain Name
        Parameter Request List Item: (6) Domain Name Server
        Parameter Request List Item: (119) Domain Search
        Parameter Request List Item: (12) Host Name
        Parameter Request List Item: (44) NetBIOS over TCP/IP Name Server
        Parameter Request List Item: (47) NetBIOS over TCP/IP Scope
        Parameter Request List Item: (26) Interface MTU
        Parameter Request List Item: (121) Classless Static Route
        Parameter Request List Item: (42) Network Time Protocol Servers
    Option: (255) End
        Option End: 255
    Padding
```

DHCP ACKnowledgement

```
Frame: 342 bytes on wire (2736 bits) on interface 0
Ethernet II, Src: 00:00:00_aa:00:05, Dst: 00:00:00_aa:00:06
Internet Protocol Version 4, Src: 192.168.1.1, Dst: 192.168.1.3
User Datagram Protocol, Src Port: 67, Dst Port: 68
Bootstrap Protocol (ACK)
    Message type: Boot Reply (2)
    Hardware type: Ethernet (0x01)
    Hardware address length: 6
    Hops: 0
    Transaction ID: 0x5b7d5971
    Seconds elapsed: 0
    Bootp flags: 0x0000 (Unicast)
    Client IP address: 0.0.0.0
    Your (client) IP address: 192.168.1.3 (192.168.1.3)
    Next server IP address: 0.0.0.0
    Relay agent IP address: 0.0.0.0
    Client MAC address: 00:00:00_aa:00:06
    Client hardware address padding: 00000000000000000000
    Server host name not given
    Boot file name not given
    Magic cookie: DHCP
    Option: (53) DHCP Message Type (ACK)
        Length: 1
        DHCP: ACK (5)
    Option: (54) DHCP Server Identifier
        Length: 4
        DHCP Server Identifier: 192.168.1.1
    Option: (51) IP Address Lease Time
        Length: 4
        IP Address Lease Time: (600s) 10 minutes
    Option: (1) Subnet Mask
        Length: 4
        Subnet Mask: 255.255.255.0
    Option: (3) Router
        Length: 4
        Router: 192.168.1.1
    Option: (6) Domain Name Server
        Length: 4
        Domain Name Server: 8.8.8.8
    Option: (255) End
        Option End: 255
    Padding
```

4.6 Review the route tables in each router

4.6.1 Routes table in Router n4

```
RTR_n4# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

C>* 10.0.0.1/32 is directly connected, lo
C>* 10.10.10.0/30 is directly connected, eth0
C>* 127.0.0.0/8 is directly connected, lo
C>* 192.168.1.0/24 is directly connected, eth1
```

4.6.2 Routes table in Router n5

```
RTR_n5# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

C>* 10.0.0.2/32 is directly connected, lo
C>* 10.10.10.0/30 is directly connected, eth0
C>* 127.0.0.0/8 is directly connected, lo
C>* 192.168.2.0/24 is directly connected, eth1
```

Reviewing the routing tables it is clear that Router n4 cannot see the network 192.168.2.0/24 and Router n5 cannot see network 192.168.1.0/24.

4.7 Static Routes

In each of the routers define a static route that points to the network that is unknown to it.

```
RTR_n4(config)# ip route 192.168.2.0/24 10.10.10.2
```

Note in the routes table a new route that has the prefix ***S***. This is the display of the Static route.

Router n4

```
RTR_n4(config)# exit
RTR_n4# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

C>* 10.0.0.1/32 is directly connected, lo
C>* 10.10.10.0/30 is directly connected, eth0
C>* 127.0.0.0/8 is directly connected, lo
C>* 192.168.1.0/24 is directly connected, eth1
S>* 192.168.2.0/24 [1/0] via 10.10.10.2, eth0
```

Router n5

```
RTR_n5(config)# ip route 192.168.1.0/24 10.10.10.1

RTR_n5(config)# exit
RTR_n5# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

C>* 10.0.0.2/32 is directly connected, lo
C>* 10.10.10.0/30 is directly connected, eth0
C>* 127.0.0.0/8 is directly connected, lo
S>* 192.168.1.0/24 [1/0] via 10.10.10.1, eth0
C>* 192.168.2.0/24 is directly connected, eth1
```

Now test the configuration by testing from Host n1 to Host n2.

```
root@n1:/tmp/pycore.48245/n1.conf# ping -c1 192.168.2.3
PING 192.168.2.3 (192.168.2.3) 56(84) bytes of data.
64 bytes from 192.168.2.3: icmp_seq=1 ttl=62 time=0.036 ms

--- 192.168.2.3 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.036/0.036/0.036/0.000 ms
```

OK so there is connectivity, what is the path of the connection. The traceroute command can reveal this. Check it against the network diagram.

```
root@n1:/tmp/pycore.48245/n1.conf# traceroute 192.168.2.3
traceroute to 192.168.2.3 (192.168.2.3), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  0.035 ms  0.008 ms  0.005 ms
 2  10.10.10.2 (10.10.10.2)  0.022 ms  0.011 ms  0.010 ms
 3  192.168.2.3 (192.168.2.3)  0.022 ms  0.014 ms  0.014 ms
```

Well this is fine for small networks as the number of routers increase then the number of static route increase exponentially. Therefore there is a need for dynamic routing.

5. Open Shortest Path First (OSPF)

Traditional IP networks in the past used the RIP or the Cisco Interior Gateway Routing Protocols (IGRP). These are Distance Vector based or hop by hop based on a distance metric of hop counts. Each router regularly passes its routing table to its neighbouring routers. Cisco developed an Enhanced version of IGRP (EIGRP) to take account of more parameters and to reduce the traffic between routers to only that the neighbours did not have already. It also incorporated Classless Inter-Domain Routing (CIDR) and Message Digest 5 (MD5) functionality. However in 1991 a DRAFT Internet Engineering Task Force (IETF) Standard RFC1247 - OSPF Version 2 let to a series of RFCs that eventually produced a new Interior Gateway Protocol (IGP) RFC2328 - Open Shortest Path First Version 2. standard in 1998. This new IGP is not based on Distance Vector but is a Link State protocol that has been the mainstay of interior networks ever since. In 2008 RFC5340 - OSPF for IPv6 incorporated support for IPv6. It is generally known as OSPFv3.

5.1 OSPF Overview

OSPF is an IGP most suited for use in large networks. OSPF uses a link-state algorithm to exchange routing information between routers in an Autonomous System (AS). An AS is a collection of routers and networks administratively configured to belong to a single organisation. OSPF enables the routers to quickly synchronise their topological databases, topology information for the AS only floods in response to topological change.

5.2 Benefits of Using OSPF versus Distance Vector protocols

Compared to other distance vector protocols like RIP and IGRP, OSPF:

- Chooses the least costly path as the best path
- Can calculate equal cost multiple paths to a destination
- Distributes external information independently
- Propagates routing information quickly and stably
- Handles Variable Length Subnet Masks (VLSM)
- Supports multicasting
- Responds quickly to topological changes by utilising reliable flooding to minimise routing traffic
- Is loop free
- Supports large metrics, external route tags and authentication of protocol exchanges

5.3 OSPF Concepts

5.3.1 Overview

To better understand the OSPF implementation, it is important to define some terms and concepts.

Link State Databases

Each OSPF router originates one or more Link State Advertisements (LSA) to describe its local part of the routing domain. The advertised link state describes the router’s local interface and adjacent neighbours. Collectively, the total LSAs in the routing domain generated by OSPF routers form what is referred to as the Link State Database (LSDB). The LSDB describes the routing topology - the collection of routers and networks in the routing domain and how they are interconnected. LSDBs are exchanged between neighbouring routers soon after the routers discover each other.

5.3.2 Reliable Flooding

The LSDB synchronises via reliable flooding to ensure each router has an identical LSDB. When a router's link state changes, a technique called reliable flooding occurs wherein an OSPF router floods its updated LSA out of all of its interfaces. The neighbouring routers receive the updated LSA, update their own LSDB, and replicate this action out of all of their interfaces (except the interface where the LSA originated).

5.3.3 Shortest Path First Algorithm

Using the LSDB as input, each OSPF router runs Dijkstra's Shortest Path First (SPF) algorithm to compute the shortest path from the calculating router to all destinations. The shortest path destinations discovered by the SPF algorithm are then updated into the IP routing table.

5.3.4 Adjacency

Adjacency is a relationship an OSPF establishes with other routers attached to its local interface. Full adjacency is achieved once these two events take place:

- Hello packets are exchanged, allowing neighbouring routers to discover each other.
- LSDBs are synchronised between neighbouring routers.

5.3.5 Network Types

- Point-to-Point networks
  - This type of network connects a single pair of routers. Point-to-Point interfaces include PPP, Serial Line Internet Protocol (SLIP), ATM, and Frame Relay.
- Broadcast networks
  - This type of network allows more than two routers to share a common network, and is able to address a single message to all attached devices. Broadcast interfaces include 10, 100 and 1000 Mbps Ethernet.

5.3.6 Designated Router

A single router, called the Designated Router (DR), is used on Broadcast networks such as Ethernet and ATM. Once the DR establishes adjacency with other routers in an area, it is responsible for generating the network link state advertisements for the broadcast network and distributing this information to other parts of the routing domain. Conversely, the DR also receives routing information from other parts of the routing domain and distributes it to other routers on its network.

By default, the first OSPF router configured on an IP subnet is the DR. When a second router is added, it becomes the Backup Designated Router (BDR). Additional routers added to the subnet defer to the existing DR and BDR. The only time this designation changes is when the existing DR or BDR fails, in which case other routers on the subnet will participate in a designated router election.

5.3.7 Authentication

All OSPF protocol exchanges are authenticated. OSPF authentication ensures routers exchange information only with trusted neighbours.

- Simple password
  - Configures a password included in all OSPF messages on an interface-by-interface basis.
  - When a router receives a message on an interface configured for simple password authentication, it checks the incoming OSPF message to ensure the proper password is included in the message. If the password is not included, the message is dropped.
  - The simple password is clear text, case sensitive, and is not encrypted.

- Cryptographic
  - This authentication method, also referred to as MD5 authentication uses a shared secret key that is configured in all routers attached to a common network or subnet.
  - Each key is identified by the combination of an interface and Key ID. A default key ID of 0 is automatically set when an interface is configured for cryptographic authentication.
  - An interface can have multiple active keys.
  - Each key has four time constants associated with it, governing the use of the key during specific time periods.

5.4 SPF Algorithm

The SFP routing algorithm is the basis for OSPF operations. When an SPF router is powered up, it initialises its routing-protocol data structures and then waits for indications from lower-layer protocols that its interfaces are functional.

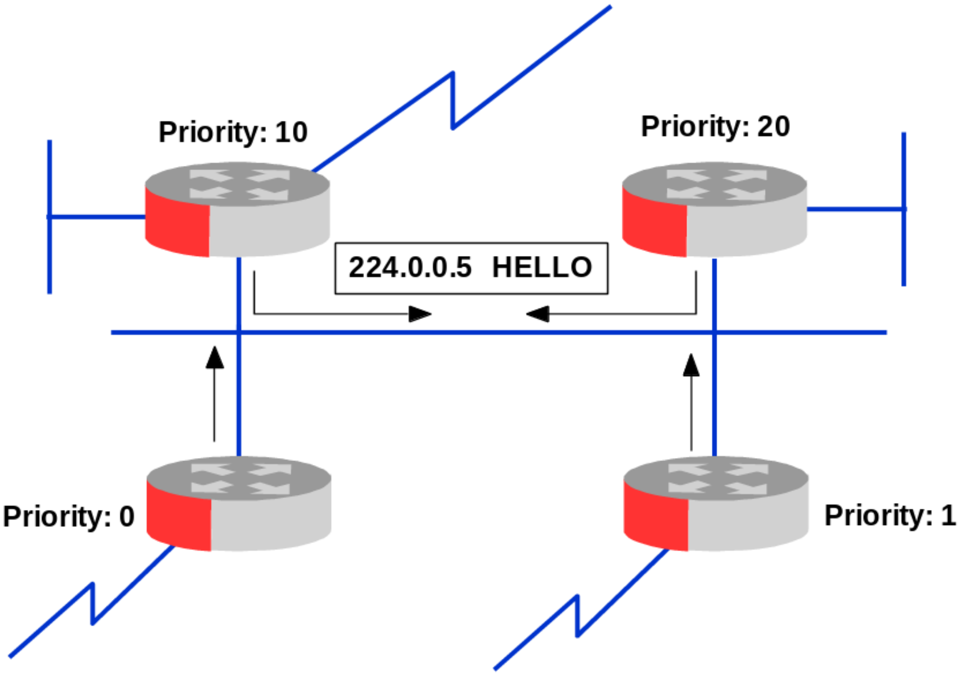


Illustration 6: OSPF HELLO

5.4.1 Hello Protocol

After a router is assured that its interfaces are functioning, it uses the OSPF Hello protocol to acquire neighbours by multicasting to *224.0.0.5*, which is to all routers with interfaces to a common network. The router sends *hello* packets to its neighbours and receives their *hello* packets. In addition to helping acquire neighbours, hello packets also act as keep-alives to let routers know that other routers are still functional.

5.4.2 DR/BDR Election

On multi-access networks (networks supporting more than two routers), the hello protocol is used to process the election of a DR and a BDR.

The DR/BDR election process is as follows:

- All routers create list of eligible routers:
    - Priority greater than 0.
    - OSPF State of 2 way.
    - DR or BDR IP Address in same network as interface.
- The BDR is chosen first which is the router with the highest priority.
- The DR is chosen from the remaining routers again the one with the highest priority.
- If there were not enough routers to have a BDR and a DR then the BDR becomes the DR.
- If the priorities are equal the Router ID is used as a tie-breaker.

5.4.3 Adjacencies

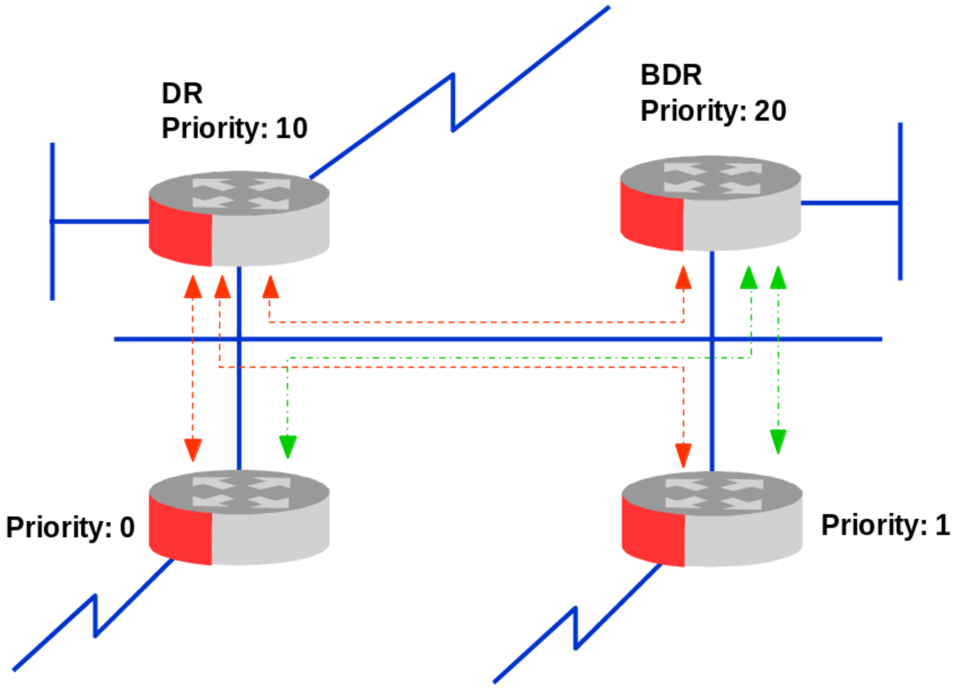


Illustration 7: OSPF Adjacencies

Among other things, the DR is responsible for generating LSA for the entire multi-access network. Designated routers allow a reduction in network traffic and in the size of the topological database.

When the link-state databases of two neighbouring routers are synchronised, the routers are said to be adjacent.

On multi-access networks, the designated router determines which routers should become adjacent. Topological databases are synchronised between pairs of adjacent routers. Adjacencies control the distribution of routing-protocol packets, which are sent and received only on adjacencies.

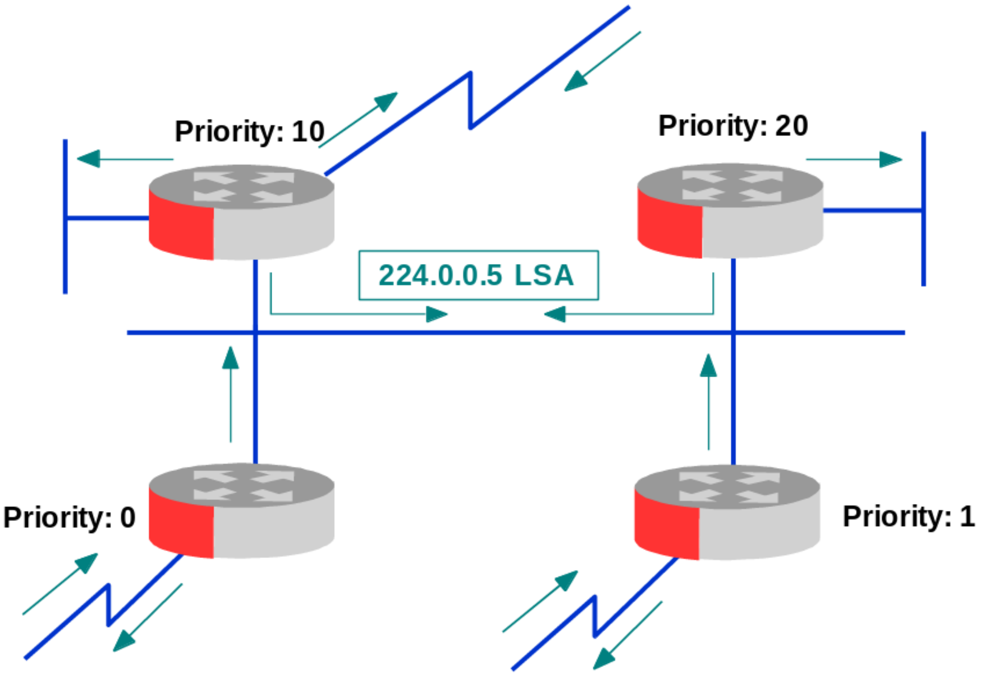


Illustration 8: OSPF Link State Advertisements (LSA) Type 1

5.4.4 Router Link State Advertisements (Type 1)

Each router periodically sends an LSA to provide information on a router's adjacencies or to inform others when a router's state changes. By comparing established adjacencies to link states, failed routers can be detected quickly, and the network's topology can be altered appropriately. From the topological database generated from LSAs, each router calculates a Shortest Path Tree (SPT), with itself as root. The shortest-path tree, in turn, yields a routing table.

Once the routing table has been established from the shortest path tree only hello packets are exchanged as a form of heart beat. Every OSPF speaker sends small hello packets out each of its interfaces every ten seconds. Hello packets are not forwarded or recorded in the OSPF database, but if none are received from a particular neighbour for forty seconds, that neighbour is marked down. LSAs are then generated marking links through a down router as down. The hello timer values can be configured, though they must be consistent across all routers on a network segment.

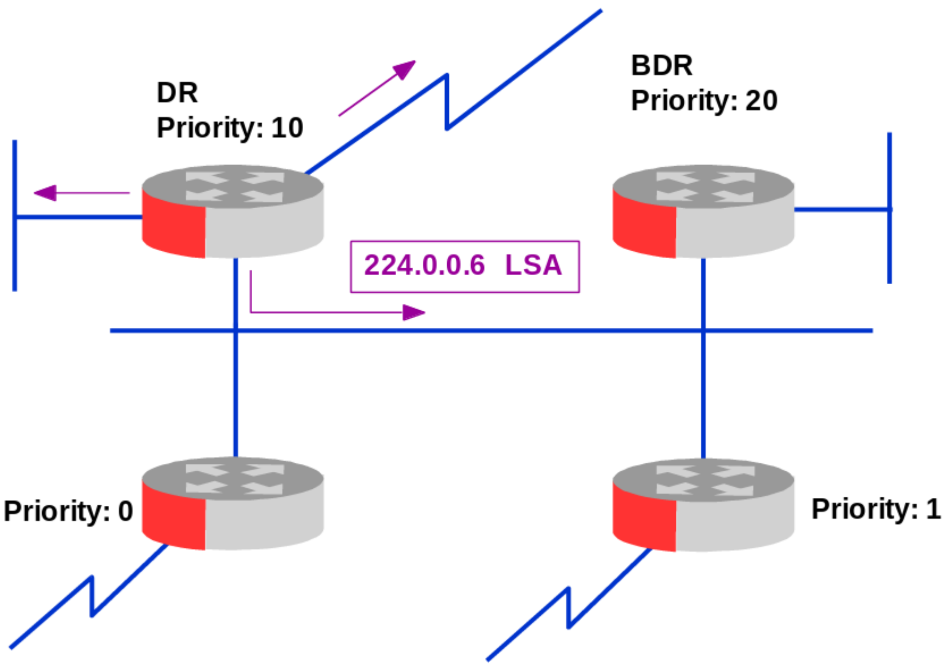


Illustration 9: OSPF Link State Advertisements (LSA) Type 2

5.4.5 Network Link State Advertisements (Type 2)

To reduce the effect of flooding DRs send information about the state of routers it is designated for to other DRs within the same area. These Network LSAs are sent to the multicast address *224.0.0.6*.

5.4.6 Other Link State Advertisements

OSPF has other LSA types associated with Area Border Routers (ABR) and Autonomous System Border Routers (ASBR) which are outside the scope of the course.

- Summary Link Advertisements (Type 3 and 4).
- External Link Advertisements (Type 5).
- Not So Stubby Area (NSSA) External Link Advertisements (Type 7).

5.4.7 OSPF Timers

Link state advertisements also age. The originating router re-advertises an LSA after it has remained unchanged for thirty minutes. If an LSA ages to more than an hour, it is flushed from the databases. These timer values are called architectural constants by RFC 2328 and 5340.

OSPFs various timers interact as follows:

- If a link goes down for twenty seconds, then comes back up, OSPF doesn't notice.
- If a link flaps constantly, but at least one of every four Hello packets make it across, OSPF doesn't notice.
- If a link goes down for anywhere from a minute to half an hour, OSPF floods an LSA when it goes down, and another LSA when it comes back up.
- If a link stays down for more than half an hour, LSAs originated by remote routers (that have become unreachable) begin to age out. When the link comes back up, all these LSAs will be re-flooded.
- If a link is down for more than an hour, any LSAs originated by remote routers will have aged out and been flushed. When the link comes back up, it will be as if it were brand new.

5.4.8 OSPF Shortest Path Tree

Now that each router has an identical LSDB each router uses Dijkstra's SPF algorithm to calculate the SPT. During the computation of the SPT, the shortest path to each node is discovered. The routing table is populated from the topology tree.

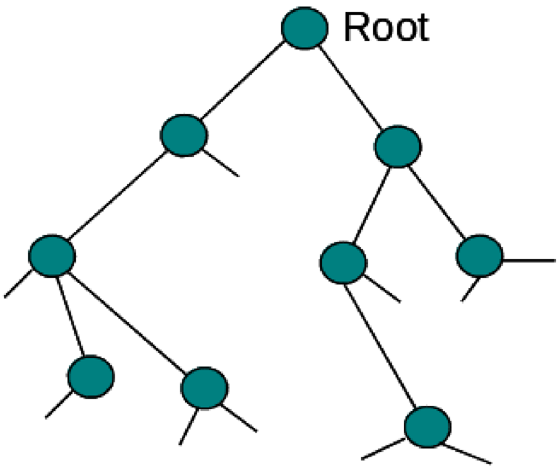


Illustration 10: Shortest Path Tree

5.5 Administrative Distances

As routes to the same location can be learned from numerous sources, i.e. Static, RIP and IGRP for example, a mechanism is required to determine the best sources. Here is the table of Administrative Distances used by Cisco Routers.

Route Source	Default Distance
Local Interface	0
Static Route	1
EIGRP	90
IGRP	100
OSPF	110
RIP	120
External EIGRP	170
Unknown	255

6. Configuring OSPFv2 for IPv4

OSPFv2 is the version of OSPF for the IPv4 protocol. Before proceeding to enable OSPF, remove the static routes.

Router n4

```
RTR_n4(config)# no ip route 192.168.2.0/24 10.10.10.2
```

Router n5

```
RTR_n5(config)# no ip route 192.168.1.0/24 10.10.10.1
```

6.1 OSPF options

At this stage a routing protocol to talk to the Router n5 is necessary. What commands are available. Well change into the router ospf configuration and use a ? symbol to find out.

```
RTR_n4# conf t
RTR_n4(config)# router ospf
RTR_n4(config-router)# ?
  area                     OSPF area parameters
  auto-cost                Calculate OSPF interface cost according to bandwidth
  capability               Enable specific OSPF feature
  compatible               OSPF compatibility list
  default-information      Control distribution of default information
  default-metric           Set metric of redistributed routes
  distance                 Define an administrative distance
  distribute-list          Filter networks in routing updates
  end                      End current mode and change to enable mode
  exit                     Exit current mode and down to previous mode
  list                     Print command list
  log-adjacency-changes    Log changes in adjacency state
  max-metric               OSPF maximum / infinite-distance metric
  mpls-te                  MPLS-TE specific commands
  neighbor                 Specify neighbor router
  network                  Enable routing on an IP network
  no                       Negate a command or set its defaults
  ospf                     OSPF specific commands
  passive-interface        Suppress routing updates on an interface
  quit                     Exit current mode and down to previous mode
 redistribute            Redistribute information from another routing protocol
  refresh                  Adjust refresh parameters
  router-id                router-id for the OSPF process
  timers                   Adjust routing timers
```

6.2 Router n4 configuration

First set the OSPF router-ID to the IP address of the local interface.

```
RTR_n4(config-router)# router-id 10.0.0.1
```

Enable routing for the networks that need to be routed. i.e. 192.168.1.0/24 and specify the backbone area.

```
RTR_n4(config-router)# network 192.168.1.0/24 area 0.0.0.0
RTR_n4(config-router)# network 10.10.10.0/30 area 0.0.0.0
```

6.3 Router n5 configuration

Configure Router n5 in a similar way to Router n4. Note the abbreviations in the command.

```
RTR_n5(config-router)# router-id 10.0.0.2

RTR_n5(config-router)# network 192.168.2.0/24 area 0.0.0.0
RTR_n5(config-router)# network 10.10.10.0/30 area 0.0.0.0
```

6.4 Reviewing Router configurations

6.4.1 Router n4

```
RTR_n4# show ip ospf neighbor

Neighbor ID Pri State    Dead Time Address    Interface       RXmtL RqstL DBsmL
10.0.0.2     1  Full/DR  32.163s 10.10.10.2  eth1:10.10.10.1   0     0     0
```

```
RTR_n4# show ip ospf database

       OSPF Router with ID (10.0.0.1)

                Router Link States (Area 0.0.0.0)

Link ID         ADV Router      Age  Seq#       CkSum  Link count
10.0.0.1        10.0.0.1         274 0x8000000c 0x174e 2
10.0.0.2        10.0.0.2         405 0x80000006 0x2443 2

                Net Link States (Area 0.0.0.0)

Link ID         ADV Router      Age  Seq#       CkSum
10.10.10.2      10.0.0.2         315 0x80000002 0x64b4
```

```
RTR_n4# show ip ospf route
============ OSPF network routing table ============
N    10.10.10.0/30         [10] area: 0.0.0.0
                           directly attached to eth1
N    192.168.1.0/24        [10] area: 0.0.0.0
                           directly attached to eth0
N    192.168.2.0/24        [20] area: 0.0.0.0
                           via 10.10.10.2, eth1

============ OSPF router routing table =============

============ OSPF external routing table ===========


RTR_n4# sh ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

C>* 10.0.0.1/32 is directly connected, lo
O   10.10.10.0/30 [110/10] is directly connected, eth1, 00:45:50
C>* 10.10.10.0/30 is directly connected, eth1
C>* 127.0.0.0/8 is directly connected, lo
O   192.168.1.0/24 [110/10] is directly connected, eth0, 00:45:50
C>* 192.168.1.0/24 is directly connected, eth0
O>* 192.168.2.0/24 [110/20] via 10.10.10.2, eth1, 00:45:40
```

6.4.2 Router n5

```
RTR_n5# sh ip o n

Neighbor ID Pri State       Dead Time Address     Interface        RXmtL RqstL
DBsmL
10.0.0.1     1  Full/Backup 32.315s   10.10.10.1  eth0:10.10.10.2    0     0
0


RTR_n5# show ip o d

       OSPF Router with ID (10.0.0.2)

                Router Link States (Area 0.0.0.0)

Link ID         ADV Router      Age  Seq#       CkSum  Link count
10.0.0.1        10.0.0.1         468 0x8000000c 0x174e 2
10.0.0.2        10.0.0.2         597 0x80000006 0x2443 2

                Net Link States (Area 0.0.0.0)

Link ID         ADV Router      Age  Seq#       CkSum
10.10.10.2      10.0.0.2         507 0x80000002 0x64b4
```

```
RTR_n5# show ip o r
============ OSPF network routing table ============
N    10.10.10.0/30         [10] area: 0.0.0.0
                           directly attached to eth0
N    192.168.1.0/24        [20] area: 0.0.0.0
                           via 10.10.10.1, eth0
N    192.168.2.0/24        [10] area: 0.0.0.0
                           directly attached to eth1

============ OSPF router routing table =============

============ OSPF external routing table ===========
```

```
RTR_n5# sh ip ro
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, A - Babel,
       > - selected route, * - FIB route

C>* 10.0.0.2/32 is directly connected, lo
O   10.10.10.0/30 [110/10] is directly connected, eth0, 00:51:33
C>* 10.10.10.0/30 is directly connected, eth0
C>* 127.0.0.0/8 is directly connected, lo
O>* 192.168.1.0/24 [110/20] via 10.10.10.1, eth0, 00:47:48
O   192.168.2.0/24 [110/10] is directly connected, eth1, 00:51:52
C>* 192.168.2.0/24 is directly connected, eth1
```

6.5 Reviewing traffic on the wire at Hub n6

OSPF Hello packet.

```
Frame: 82 bytes on wire (656 bits), on interface 0
    Interface id: 0 (veth3.0.50)
    Encapsulation type: Ethernet (1)
    Arrival Time: Feb 23, 2016 16:30:07.648400000 GMT
    Epoch Time: 1456245007.648400000 seconds
    Frame Number: 319
    Frame Length: 82 bytes (656 bits)
    Capture Length: 82 bytes (656 bits)
Ethernet II, Src: 00:00:00_aa:00:03, Dst: 01:00:5e:00:00:05
    Type: IP (0x0800)
Internet Protocol Version 4, Src: 10.10.10.2, Dst: 224.0.0.5
    Version: 4
    Header Length: 20 bytes
    Differentiated Services Field: 0xc0
    Total Length: 68
    Identification: 0x8490 (33936)
    Flags: 0x00
    Fragment offset: 0
    Time to live: 1
    Protocol: OSPF IGP (89)
    Header checksum: 0x4000 [validation disabled]
Open Shortest Path First
    OSPF Header
        Version: 2
        Message Type: Hello Packet (1)
        Packet Length: 48
        Source OSPF Router: 10.0.0.2 (10.0.0.2)
        Area ID: 0.0.0.0 (0.0.0.0) (Backbone)
        Checksum: 0xbf84 [correct]
        Auth Type: Null (0)
        Auth Data (none): 0000000000000000
    OSPF Hello Packet
        Network Mask: 255.255.255.252 (255.255.255.252)
        Hello Interval [sec]: 10
        Options: 0x02 (E)
            0... .... = DN: Not set
            .0.. .... = O: Not set
            ..0. .... = DC: Demand Circuits are NOT supported
            ...0 .... = L: The packet does NOT contain LLS data block
            .... 0... = NP: NSSA is NOT supported
            .... .0.. = MC: NOT Multicast Capable
            .... ..1. = E: External Routing Capability
            .... ...0 = MT: NO Multi-Topology Routing
        Router Priority: 1
        Router Dead Interval [sec]: 40
        Designated Router: 10.10.10.2
        Backup Designated Router: 10.10.10.1
        Active Neighbor: 10.0.0.1
```

OSPF Link State Update packet.

```
Frame: 110 bytes on wire (880 bits) on interface 0
Ethernet II, Src: 00:00:00_aa:00:02, Dst: 01:00:5e:00:00:05
    Type: IP (0x0800)
Internet Protocol Version 4, Src: 10.10.10.1, Dst: 224.0.0.5
    Protocol: OSPF IGP (89)
Open Shortest Path First
    OSPF Header
        Version: 2
        Message Type: LS Update (4)
        Packet Length: 76
        Source OSPF Router: 10.0.0.1 (10.0.0.1)
        Area ID: 0.0.0.0 (0.0.0.0) (Backbone)
        Checksum: 0x564a [correct]
        Auth Type: Null (0)
        Auth Data (none): 0000000000000000
    LS Update Packet
        Number of LSAs: 1
        Router-LSA
            .000 0000 0000 0001 = LS Age (seconds): 1
            0... .... .... .... = Do Not Age Flag: 0
            Options: 0x02 (E)
                0... .... = DN: Not set
                .0.. .... = O: Not set
                ..0. .... = DC: Demand Circuits are NOT supported
                ...0 .... = L: The packet does NOT contain LLS data block
                .... 0... = NP: NSSA is NOT supported
                .... .0.. = MC: NOT Multicast Capable
                .... ..1. = E: External Routing Capability
                .... ...0 = MT: NO Multi-Topology Routing
            LS Type: Router-LSA (1)
            Link State ID: 10.0.0.1 (10.0.0.1)
            Advertising Router: 10.0.0.1 (10.0.0.1)
            Sequence Number: 0x8000000b
            Checksum: 0x194d
            Length: 48
            Flags: 0x00
                .... .0.. = V: NO Virtual link endpoint
                .... ..0. = E: NO AS boundary router
                .... ...0 = B: NO Area border router
            Number of Links: 2
            Type: Transit  ID: 10.10.10.2      Data: 10.10.10.1      Metric: 10
                Link ID: 10.10.10.2 - IP address of Designated Router
                Link Data: 10.10.10.1 (10.10.10.1)
                Link Type: 2 - Connection to a transit network
                Number of Metrics: 0 - TOS
                0 Metric: 10
            Type: Stub     ID: 192.168.1.0     Data: 255.255.255.0   Metric: 10
                Link ID: 192.168.1.0 - IP network/subnet number
                Link Data: 255.255.255.0 (255.255.255.0)
                Link Type: 3 - Connection to a stub network
                Number of Metrics: 0 - TOS
                0 Metric: 10
```

OSPF Link State ACKnowledge packet.

```
Frame4: 78 bytes on wire (624 bits), on interface 0
Ethernet II, Src: 00:00:00_aa:00:03, Dst: 01:00:5e:00:00:05
    Type: IP (0x0800)
Internet Protocol Version 4, Src: 10.10.10.2, Dst: 224.0.0.5
    Protocol: OSPF IGP (89)
Open Shortest Path First
    OSPF Header
        Version: 2
        Message Type: LS Acknowledge (5)
        Packet Length: 44
        Source OSPF Router: 10.0.0.2 (10.0.0.2)
        Area ID: 0.0.0.0 (0.0.0.0) (Backbone)
        Checksum: 0x4440 [correct]
        Auth Type: Null (0)
        Auth Data (none): 0000000000000000
    LSA Header
        .000 0000 0000 0001 = LS Age (seconds): 1
        0... .... .... .... = Do Not Age Flag: 0
        Options: 0x02 (E)
            0... .... = DN: Not set
            .0.. .... = O: Not set
            ..0. .... = DC: Demand Circuits are NOT supported
            ...0 .... = L: The packet does NOT contain LLS data block
            .... 0... = NP: NSSA is NOT supported
            .... .0.. = MC: NOT Multicast Capable
            .... ..1. = E: External Routing Capability
            .... ...0 = MT: NO Multi-Topology Routing
        LS Type: Router-LSA (1)
        Link State ID: 10.0.0.1 (10.0.0.1)
        Advertising Router: 10.0.0.1 (10.0.0.1)
        Sequence Number: 0x8000000b
        Checksum: 0x194d
        Length: 48
```

6.6 Testing the link from Host n1 to Host n2

A ping and traceroute conform the link from Host n1 to Host n2.

```
root@n1:/tmp/pycore.41200/n1.conf# ping -c1 192.168.2.3
PING 192.168.2.3 (192.168.2.3) 56(84) bytes of data.
64 bytes from 192.168.2.3: icmp_seq=1 ttl=62 time=0.104 ms

--- 192.168.2.3 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.104/0.104/0.104/0.000 ms


root@n1:/tmp/pycore.41200/n1.conf# traceroute 192.168.2.3
traceroute to 192.168.2.3 (192.168.2.3), 30 hops max, 60 byte packets
 1  192.168.1.1 (192.168.1.1)  0.031 ms  0.006 ms  0.005 ms
 2  10.10.10.2 (10.10.10.2)  0.022 ms  0.010 ms  0.009 ms
 3  192.168.2.3 (192.168.2.3)  0.018 ms  0.013 ms  0.012 ms
```

6.7 Summary of the configuration in both routers

These lists can be copy and pasted into the respective vtysh terminals.

Router n4

```
conf t
hostname RTR_n4
int eth0
ip addr 10.10.10.1/30
no shut
int eth1
ip addr 192.168.1.1/24
no shut
int lo
ip addr 10.0.0.1/32
no shut
```

Static Route

```
ip route 192.168.2.0/24 10.10.10.2
```

Remove Static Route and OSPF Configuration

```
no ip route 192.168.2.0/24 10.10.10.2

router ospf
router-id 10.0.0.1
network 192.168.1.0/24 area 0.0.0.0
network 10.10.10.0/30 area 0.0.0.0
```

Router n5

```
conf t
hostname RTR_n5
int eth0
ip addr 10.10.10.2/30
no shut
int eth1
ip addr 192.168.2.1/24
no shut
int lo
ip addr 10.0.0.2/32
no shut
```

Static Route

```
ip route 192.168.1.0/24 10.10.10.1
```

Remove Static Route and OSPF Configuration

```
no ip route 192.168.2.0/24 10.10.10.2

router ospf
router-id 10.0.0.2
network 192.168.2.0/24 area 0.0.0.0
network 10.10.10.0/30 area 0.0.0.0
```

This page is intentionally blank