

DHCP in Action: Streamlining Network Client Address Management

Dynamic Host Configuration Protocol (DHCP) is a network protocol that lets network administrators manage and automate the assignment of IP addresses. Without DHCP for IPv4, the administrator must manually assign and configure IP addresses, preferred DNS servers, and default gateways. As the network grows in size, this becomes an administrative problem when devices are added or moved from one internal network to another.

Related CCNA v1.1 exam topic:

- 4.3 Explain the role of DHCP and DNS within the network
- 4.6 Configure and verify DHCP client and relay

In this lab, we will explore how to:

- Configure a router as an IPv4 DHCP Client
- Configure a Router as an IPv4 DHCP Server
- Configure an IPv4 DHCP Relay

Setup and Scenario

In this set of lab-based demonstrations, you are the network engineer for a growing organization tasked with updating the network to support new network needs. The network was originally deployed using static IPv4 addresses on all hosts and servers since the network was small, but now the network has grown and requires that IPv4 address assignment be automated and centralized on a DHCP server.

You've been asked to:

- Configure RTR2 E0/3 to request and receive a dynamic IPv4 address from the ISP.
- Configure RTR1 as an IPv4 DHCP server for the RTR1 LAN and RTR2 LAN.
- Configure RTR2 as an IPv4 DHCP Relay Agent for the RTR2 LAN.

*Be sure to **START** the lab before continuing to the demo labs but do not power on PC1 or PC2 for now.*

Part 1: Reviewing the Current State of the Network

Before we jump into configuring DHCP across the network, let's check the current status of the network and how it is operating.

Step 1

Open a console connection to RTR1 and RTR2 and verify the OSPF configuration.

```
RTR1# show run | section ospf
router ospf 1
router-id 0.0.0.1
network 10.1.1.0 0.0.0.3 area 0
network 192.168.1.0 0.0.0.255 area 0
```

```
RTR2# show run | section ospf
router ospf 1
router-id 0.0.0.2
network 10.1.1.0 0.0.0.3 area 0
network 192.168.2.0 0.0.0.255 area 0
default-information originate
```

Notice that both RTR1 and RTR2 are configured with OSPF process ID 1 and that all network statements are assigned to area 0. RTR2 is also configured with the 'default-information originate' command that allows RTR2 to advertise a default route to other OSPF routers as long as RTR2 has a default route in its own routing table. RTR2 does not currently have a default route but this will be corrected in the next steps. ### Step 2 Verify the routing tables on RTR1 and RTR2

```
RTR1# show ip route
<... output omitted ...>
```

Gateway of last resort is not set

```
    10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C    10.1.1.0/30 is directly connected, Ethernet0/0
L    10.1.1.1/32 is directly connected, Ethernet0/0
    192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C    192.168.1.0/24 is directly connected, Ethernet0/1
L    192.168.1.1/32 is directly connected, Ethernet0/1
O    192.168.2.0/24 [110/20] via 10.1.1.2, 2w3d, Ethernet0/0
```

```
RTR2# show ip route
<... output omitted ...>
```

Gateway of last resort is not set

```

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C      10.1.1.0/30 is directly connected, Ethernet0/0
L      10.1.1.2/32 is directly connected, Ethernet0/0
O      192.168.1.0/24 [110/20] via 10.1.1.1, 2w3d, Ethernet0/0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C      192.168.2.0/24 is directly connected, Ethernet0/1
L      192.168.2.1/32 is directly connected, Ethernet0/1

```

RTR1 is learning about the RTR2 LAN (192.168.2.0/24) and RTR2 is learning about the RTR1 LAN (192.168.1.0/24).

Also notice that RTR2 does not currently have a default route to advertise to RTR1.

Step 3

Verify the interfaces on RTR2

```

RTR2# show ip interface brief
Interface      IP-Address      OK? Method Status      Protocol
Ethernet0/0    10.1.1.2        YES TFTP    up          up
Ethernet0/1    192.168.2.1    YES TFTP    up          up
Ethernet0/2    unassigned      YES TFTP    administratively down down
Ethernet0/3    unassigned      YES unset  administratively down down

```

The internet-facing Ethernet0/3 interface is currently administratively down and has not IP address configured. You will correct this in the next part of the lab.

Part 2: Configure a router as an IPv4 DHCP Client

The RTR2 Ethernet 0/3 interface is connected to the "Internet" external connector node in CML. This node acts as an ISP that provides dynamic IPv4 addresses to its clients. In this part of the lab, you will enable RTR2 to request and receive an IPv4 DHCP address on interface Ethernet 0/3. RTR2 will also automatically create a default route pointing to the ISP. RTR2 should then advertise this default route to RTR1 via OSPF.

Step 1

Configure RTR2 E0/3 as an IPv4 DHCP client and enable the interface. After the interface comes up, it should take about 10 seconds for the E0/3 to received its IPv4 address.

```

RTR2# conf t
Enter configuration commands, one per line. End with CNTL/Z.
RTR2(config)# interface E0/3
RTR2(config-if)# ip address dhcp
RTR2(config-if)# no shutdown
RTR2(config-if)#
*Jan  3 16:48:43.228: %LINK-3-UPDOWN: Interface Ethernet0/3, changed state to up
RTR2(config-if)#
*Jan  3 16:48:44.228: %LINEPROTO-5-UPDOWN: Line protocol on Interface Ethernet0/3, changed state to up
RTR2(config-if)#
*Jan  3 16:48:50.386: %DHCP-6-ADDRESS_ASSIGN: Interface Ethernet0/3 assigned DHCP address 192.168.255.141, mask 255.255.255.0, hostname RTR2

```

In the highlighted text above, notice that the interface is now configured with the 192.168.255.141/24 address. Use the `show ip interface brief` command to confirm the new DHCP address.

Step 2

Verify the routing table on RTR1 and RTR2. Both routers should now have a default route in its routing table.

```

RTR1# show ip route
<... output omitted ...>

```

Gateway of last resort is 10.1.1.2 to network 0.0.0.0

```

O*E2  0.0.0.0/0 [110/1] via 10.1.1.2, 00:06:47, Ethernet0/0
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C      10.1.1.0/30 is directly connected, Ethernet0/0
L      10.1.1.1/32 is directly connected, Ethernet0/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C      192.168.1.0/24 is directly connected, Ethernet0/1
L      192.168.1.1/32 is directly connected, Ethernet0/1
O      192.168.2.0/24 [110/20] via 10.1.1.2, 2w3d, Ethernet0/0

```

```

RTR2# show ip route
<... output omitted ...>

```

Gateway of last resort is 192.168.255.1 to network 0.0.0.0

```

S*    0.0.0.0/0 [254/0] via 192.168.255.1
      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C      10.1.1.0/30 is directly connected, Ethernet0/0
L      10.1.1.2/32 is directly connected, Ethernet0/0
O      192.168.1.0/24 [110/20] via 10.1.1.1, 2w3d, Ethernet0/0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks

```

```

C      192.168.2.0/24 is directly connected, Ethernet0/1
L      192.168.2.1/32 is directly connected, Ethernet0/1
      192.168.255.0/24 is variably subnetted, 2 subnets, 2 masks
C      192.168.255.0/24 is directly connected, Ethernet0/3
L      192.168.255.141/32 is directly connected, Ethernet0/3

```

RTR2 is automatically configured with a static default route thanks to DHCP. RTR1 is learning the default route via OSPF thanks to an External Type 5 LSA being advertised by RTR2 because of the default-information originate command. ### Step 3 Verify connectivity from RTR1 and RTR2 to a public Internet address.

```

RTR1# ping 8.8.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms

```

```

RTR2# ping 8.8.8.8
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.8.8, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms

```

Since RTR2 is configured with NAT (which will be covered in Season 2, Episode 3), both RTR1 and RTR2 are able to reach a public Internet address. ## Part 3: Configure a Router as an IPv4 DHCP Server The Cisco IOS DHCP server is a full DHCP server implementation that assigns and manages IPv4 addresses from specified address pools within the device to DHCP clients. The DHCP server can be configured to assign additional parameters such as the IPv4 address of the DNS server and the default gateway. In this part of the lab, you will configure RTR1 as an IPv4 DHCP server for the RTR1 LAN and the RTR2 LAN. ### Step 1 Configure the IPv4 DHCP pool for the RTR1 LAN (192.168.1.0/24) and exclude the first five addresses of that network. Use the pool name RTR1_LAN.

```

RTR1(config)# ip dhcp pool RTR1_LAN
RTR1(dhcp-config)# network 192.168.1.0 /24
RTR1(dhcp-config)# default-router 192.168.1.1
RTR1(dhcp-config)# dns-server 8.8.8.8
RTR1(dhcp-config)# domain-name lab.example
RTR1(dhcp-config)# lease 0 12
RTR1(dhcp-config)#exit
RTR1(config)# ip dhcp excluded-address 192.168.1.1 192.168.1.5

```

In the configuration example above the, IPv4 addresses are assigned from the address pool 192.168.1.0/24 with a lease time of 12 hours. Additional parameters are the default gateway, domain name, and DNS server. Also, IPv4 addresses from 192.168.1.1 to 192.168.1.5 are not assigned to the end devices. Use the ``?`` option under the DHCP pool to investigate the different configuration options available. ### Step 2 Use the `show ip dhcp pool RTR1_LAN` command to investigate the DHCP pool configuration.

```

RTR1# show ip dhcp pool RTR1_LAN

Pool RTR1_LAN :
  Utilization mark (high/low)      : 100 / 0
  Subnet size (first/next)          : 0 / 0
  Total addresses                    : 254
  Leased addresses                   : 0
  Excluded addresses                 : 5
  Pending event                     : none
  1 subnet is currently in the pool :
  Current index      IP address range      Leased/Excluded/Total
  192.168.1.1        192.168.1.1 - 192.168.1.254    0 / 5 / 254

```

The output confirms that 5 addresses are excluded from the total range and that there are currently no leased addresses. ### Step 3 Enable DHCP debugging on RTR1 and then start the PC1 device.

```

RTR1# debug ip dhcp server events
DHCP server event debugging is on.
RTR1# debug ip dhcp server packet
DHCP server packet debugging is on.

```

Right-click on PC1 and select **Start**.

Within a few seconds you should see some DHCP debugging messages. If you scroll through them, you should find the four DHCP messages exchanged between PC1 and RTR1:

```

*Jan  6 15:34:45.855: DHCPD: DHCPDISCOVER received from client 0152.5400.1529.f9 on interface Ethernet0/1.
*Jan  6 15:34:47.856: DHCPD: Sending DHCPOFFER to client 0152.5400.1529.f9 (192.168.1.6).
*Jan  6 15:34:47.859: DHCPD: DHCPREQUEST received from client 0152.5400.1529.f9 on interface Ethernet0/1
*Jan  6 15:34:47.859: DHCPD: Sending DHCPACK to client 0152.5400.1529.f9 (192.168.1.6).

```

In the output above, PC1 was assigned the 192.168.1.6 address by the DHCP server on RTR1. ### Step 4 Verify the IPv4 configuration on PC1 and test connectivity to the Internet. Open the console on PC1. Use the `ip addr show dev eth0` command to verify its IPv4 configuration. PC1 will have an IP address from the 192.168.1.0/24 network.

```

PC1:~$ ip addr show dev eth0
2: eth0: mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 52:54:00:15:29:f9 brd ff:ff:ff:ff:ff:ff

```

```
inet 192.168.1.6/24 scope global eth0
    valid_lft forever preferred_lft forever
inet6 fe80::5054:ff:fe15:29f9/64 scope link
    valid_lft forever preferred_lft forever
```

Use the `ip route list` command to verify the PC's routing table. You should see the default gateway set to 192.168.1.1 which is the IP address of RTR1 E0/1.

```
PC1:~$ ip route list
default via 192.168.1.1 dev eth0 metric 202
172.17.0.0/16 dev docker0 scope link src 172.17.0.1
192.168.1.0/24 dev eth0 scope link src 192.168.1.6
```

From the command prompt, ping the public DNS server at **8.8.8.8**. This ping should be successful.

Open a VNC connection to PC1 and use the browser to test connectivity to **www.cisco.com**. This test should also be successful since the DHCP server is configured to provide clients with the DNS server's IP address (8.8.8.8)

Step 5

Configure the IPv4 DHCP pool for the RTR2 LAN (192.168.2.0/24) and exclude the first five addresses of that network. Use the pool name RTR2_LAN.

```
RTR1(config)# ip dhcp pool RTR2_LAN
RTR1(dhcp-config)# network 192.168.2.0 /24
RTR1(dhcp-config)# default-router 192.168.2.1
RTR1(dhcp-config)# dns-server 8.8.8.8
RTR1(dhcp-config)# domain-name lab.example
RTR1(dhcp-config)# lease 0 12
RTR1(dhcp-config)#exit
RTR1(config)# ip dhcp excluded-address 192.168.2.1 192.168.2.5
```

Step 6

Start a packet capture on the link between RTR1 and RTR2 to capture the DHCP messages sent from PC2.

Right-click the link between RTR1 and RTR2 and click **Packet Capture**. In the packet capture tab, select Settings and add a BPF filter for DHCP packets. Click **Apply** and then click **Start**.

Right-click on PC2 and select **Start** to boot up the device.

Return to the packet capture tab. You should not see any DHCP messages crossing the link between RTR1 and RTR2. Recall that DHCP messages from PC2 (DHCP Discover and DHCP Request) are broadcast messages and don't get forwarded off the local LAN by default. You will correct this issue in the next part of the lab.

Part 4: Configure an IPv4 DHCP Relay

A DHCP relay agent is any host that forwards DHCP packets between clients and servers. Relay agents are used for forwarding requests and replies between clients and servers when they are not on the same subnet. DHCP requests are sent as broadcasts, and because routers don't forward broadcasts, you need a relay functionality to reach the DHCP server. In this part of the lab, you will configure RTR2 E0/1 as a DHCP relay agent. This will allow PC2 DHCP messages to be forwarded to RTR1.

To configure the DHCP relay agent to forward packets to a DHCP server, you need to enter the interface configuration mode using the `interface` command. Then, use the `ip helper-address` command to specify that the interface will forward UDP broadcasts, including BOOTP and DHCP, to the specified server address.

The `ip helper-address` command should be issued on the interface where the DHCP broadcasts are received.

These steps show how DHCP requests are processed when DHCP relay is used:

1. A DHCP client broadcasts a DHCP request.
2. The DHCP relay device includes option 82 and sends the DHCP request as a unicast packet to the DHCP server. Option 82 includes remote ID and circuit ID information.
3. The DHCP server responds to the DHCP relay.
4. The DHCP relay strips off option 82 and sends the response to the DHCP client.

Step 1

Configure the DHCP relay functionality on RTR2 E0/1.

```
RTR2(config)# interface E0/1
RTR2(config-if)# ip helper-address 10.1.1.1
```

The `ip helper-address` command tells RTR2 that any broadcasts received on E0/1 will be sent to the 10.1.1.1 address which is configured on RTR1 E0/0.

Stop and re-start PC2 to trigger the client's DHCP process again.

Step 2

Return to the packet capture tab to observe the DHCP messages that are now crossing the link between RTR1 and RTR2. Your output should be similar to what is shown below:

Time	Source	Destination	Protocol	Info
0.000000	192.168.2.1	10.1.1.1	DHCP	DHCP Discover - Transaction ID 0xba02f774
0.001355	10.1.1.1	192.168.2.1	DHCP	DHCP Offer - Transaction ID 0xba02f774
0.002901	192.168.2.1	10.1.1.1	DHCP	DHCP Request - Transaction ID 0xba02f774
0.003654	10.1.1.1	192.168.2.1	DHCP	DHCP ACK - Transaction ID 0xba02f774

The packet capture confirms that DHCP Discover and DHCP Request received on RTR2 E0/1 are being relayed to 10.1.1.1 on RTR1. Click through the captured packets to investigate the option fields in the DHCP header.

Step 3

Verify the IPv4 configuration on PC2 and test connectivity to the Internet.

Open the console on PC2 and access the command prompt. Use the `ip addr show dev eth0` command to verify its IPv4 configuration. PC1 will have an IP address from the 192.168.2.0/24 network.

Use the `ip route list` command to verify the PC's routing table. You should see the default gateway set to 192.168.2.1 which is the IP address of RTR2 E0/1.

From the command prompt, ping the public DNS server at **8.8.8.8**. This ping should be successful.

Open a VNC connection to PC2 and use the browser to test connectivity to **www.cisco.com**. This test should be successful.

Step 4

Return to RTR1 and inspect the DHCP debugging output. You should see the following messages:

```
*Jan  6 16:21:01.881: DHCPD: DHCPDISCOVER received from client 0152.5400.1f04.4d through relay 192.168.2.1.
```

```
*Jan  6 16:21:01.881: DHCPD: Sending DHCPOFFER to client 0152.5400.1f04.4d (192.168.2.7).
```

```
*Jan  6 16:21:01.883: DHCPD: DHCPREQUEST received from client 0152.5400.1f04.4d on interface Ethernet0/0
```

```
*Jan  6 16:21:01.883: DHCPD: Client is Selecting (DHCP Request with Requested IP = 192.168.2.7, Server ID = 10.1.1.1)
```

```
*Jan  6 16:21:01.883: DHCPD: Sending DHCPACK to client 0152.5400.1f04.4d (192.168.2.7).
```

Notice that RTR1 is aware that the DHCP Discover message was relayed from 192.168.2.1 and the DHCP client (PC2) is being serviced by the DHCP server located at 10.1.1.1.