NETWORK PROTOCOLS & BASICS

QUALITY THOUGHT

INFOSYSTEMS (INDIA) PVT. LTD.

DHCP

DHCP, or Dynamic Host Configuration Protocol is a networking protocol that you most likely use every day on almost all of your devices.

If you don’t have to set a static IP address for your devices, odds are they are set with DHCP. DHCP is not just for IP address, subnet mask, and Gateway, however. DHCP provides information you typically don’t look at, for example: NTP servers, DNS servers, FTP and configuration servers for devices such as desk phones, and many other services that can be set using custom “option sets.” Proper configuration of DHCP is critical to maintaining a properly functioning and streamlined networking environment.

HOW IT WORKS

The process is as follows:

1.DHCPDiscover: A client device broadcasts a DHCP Discover message to all devices on the network asking for a DHCP server and address.

2.A DHCP server on the network receives this message and sees the request. It checks its current leases and finds an available IP address.

3.DHCPOffer: The server then provides a “DHCP offer” message for the client.

4.DHCPRequest: The client then sends a “DHCP request” packet back to the server letting it know it has chosen the IP offered to it.

5.DHCPACK: The Server responds with a DHCP ACK to acknowledge the request and send over any other DHCP options that have been set. This message is still a broadcast because the client doesn’t have an IP address officially assigned to it yet.

HOW IT WORKS (CONTD)

This process sounds complicated, but typically happens in seconds or even milliseconds. I have performed a packet capture to illustrate this process in action:

As you can see in the “Time” column, the entire process takes less than a second for everything to complete and my computer to be assigned an IP address!

No.	Time	Source	Destination	Protocol	Length	Sequence number	Differentiated Services Field	Info
23	14:45:08.907552	0.0.0.0	255.255.255.255	DHCP	342		0x00	DHCP Discover -
86	14:45:09.770401	192.168.1.1	192.168.1.3	DHCP	590		0x00	DHCP Offer -
87	14:45:09.770706	0.0.0.0	255.255.255.255	DHCP	355		0x00	DHCP Request -
88	14:45:09.840487	192.168.1.1	192.168.1.3	DHCP	590		0x00	DHCP ACK -

DHCP CONFIGURATION

CONFIGURATION

You may have noticed a few words I haven't talked about yet in the previous example, such as "lease" and "scope." I am going to go over many of the common DHCP terminology here.

POOL/SCOPE

The IP addresses available for clients to request. Many home networks today use a 192.168.x.0/24, like 192.168.1.0/24. This provides an address range of 192.168.1.1 – 192.168.1.254. Any address that is not taken in this range is available to be "leased" to clients on the network.

DHCP CONFIGURATION

LEASE

A lease is an IP address given to a client. This is called a lease because it expires after a certain period of time and can return to the address pool if necessary. Typically, a client will continue requesting the same leased IP address at half the configured lease time. For instance, if an address is leased to a device for 24 hours, it will request the address again at 12 hours to prevent the address from returning to the pool and being used by another device. If this request does not happen (perhaps if the device has left the network), the IP address will return to the pool and be reassigned to another device if one requests.

DHCP CONFIGURATION

IP RESERVATION

With an IP reservation, you can instruct the DHCP server to always assign the same address to a device using that device's MAC address. A MAC address is the hardcoded or virtual hardware address used by a device's network interface card (NIC).

IP CONFLICT

An IP conflict occurs when 2 or more devices on a network attempt to request the same IP address. This typically happens when one device has a static IP address set that is within the address pool and another device attempts to request that IP address. To avoid this, it is best to configure an IP reservation for the static client or modify the address scope such that they do not contain the IP address statically assigned to the client. For instance, if a client is given 192.168.1.10, set the scope to only provide IP addresses from 192.168.1.11 – 192.168.1.253. This will allow all of the addresses below 192.168.1.11 to be statically assigned without conflicting with a DHCP host.

DHCP OPTION SET

As previously stated, DHCP can provide other information aside from just the typical IP address, subnet mask, and DNS records. These option sets can help minimize the amount of manual work you have to do to maintain a network that contains many servers.

DHCP options are configured using an “Option Code” followed by the required information. Here is an example of a DHCP option set configured on an AWS VPC network:

This network is configured to use different DNS servers than what are automatically assigned. These are Google public DNS servers and are not typically configured in an environment like this. I have also configured a custom NTP server. These settings will propagate to any client configured for DHCP within this VPC.

DHCP OPTION SET

```
    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: (86400s) 1 day
  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: 192.168.1.1
```

DHCP OPTION SET

DHCP Option 54 is the DHCP Server of my router with IP address 192.168.1.1

DHCP Option 51 is the IP address lease time, which is set to 1 day.

DHCP Option 1 is the subnet mask, which is 255.255.255.0 or a /24.

DHCP Option 3 is the Router or Gateway, which is also 192.168.1.1

DHCP Option 6 is the DNS Server, also 192.168.1.1

TROUBLESHOOTING

Luckily, DHCP is a fairly straightforward protocol to troubleshoot. Many issues can be resolved by simply forcing the client to request a new IP address with a DHCP release command and DHCP renew command appropriate for the OS you are using.

The commands for Linux are:

`dhclient -r`

To release and

`dhclient`

To renew

“`dhclient -r -v && dhclient -v” TRY this command to check how dhcp works`

DNS

DNS, or “Domain Name System,” is one of the most important parts of The Internet today. DNS is essentially a server or hierarchical configuration of servers that acts as a “phone book” for a network. DNS translates a hostname (name of a computer) to an IP address (“phone number” of a computer).

If you type https://www.google.com in your browser, your browser doesn’t see it as words like we see it. Your browser sees the IP address, just as when you select a contact in your cellphone, your cellphone sees a phone number.

Every networked device, including webservers, phones, computers, modems, routers, even smart lightbulbs and stoves has an IP address. Since humans aren’t as good with numbers as computers are, we invented DNS to translate these numbers into domain names and hostnames that make much more sense to us.

DNS (Finding IP ADDRESS)

1. Open the terminal or command prompt of your operating system.
   a. For Windows: click on your start menu, type “Powershell” and hit “enter”.
   b. For OSX: Click on the “Spotlight” magnifying glass and type “Terminal” and hit “enter”.
2. Once you have the utility up, type:

   “ping google.com”

   (Linux and OSX users, press ctrl+c to stop the pings after you have seen a few.)

You should see output similar to this:

```
[root@localhost ~]# ping google.com -c 4
PING google.com (216.58.219.206) 56(84) bytes of data.
64 bytes from lga25s40-in-f14.1e100.net (216.58.219.206): icmp_seq=1 ttl=55 time=21.1 ms
64 bytes from lga25s40-in-f14.1e100.net (216.58.219.206): icmp_seq=2 ttl=55 time=17.1 ms
64 bytes from lga25s40-in-f14.1e100.net (216.58.219.206): icmp_seq=3 ttl=55 time=22.0 ms
```

3. As you can see, this translates an easy to remember domain name “google.com” into an easy for a computer IP address “216.58.219.206”.

Anatomy of DNS

A DNS Name Server is just a computer. It might be a very small computer, such as a Raspberry Pi, or it might be a giant pool of clustered server nodes. The only requirements for a server to be a DNS server is that it translates Domain Names to IP Addresses. DNS servers typically run a DNS server software such as Bind or Microsoft DNS. These servers can host simple “Static DNS” records that must be modified manually or they can dynamically update their DNS records based on queries they are asked to perform. If a DNS server does not have a particular record, it can reach out to another DNS server to ask it. If this DNS server has the information, the DNS server that initiated the query can then update its own database, or cache, in order to prevent it from having to ask again.

FQDN: The FQDN, or Fully Qualified Domain Name, is the name for an entire URL that resolves a resource. If you just tried to ping “.com” or “linuxacademy” or “www”, this probably wouldn’t work. It can be configured to work, but that’s a topic for another guide. To access resources on the web, the FQDN is required which includes at least the domain name and the TLD.

ANATOMY OF DNS

TLD: The TLD, or Top Level Domain, is the last part of an FQDN. This is typically “.com”, “.org”, or some other short abbreviation. More on this is in the following section.

Domain: The Domain is the most basic form of the record that can be accessed. In this case, google.com is the domain name. Sometimes, the website is hosted on the “naked domain” name like this and can be accessed. Many times, however, the naked domain name redirects to the subdomain such as www.google.com in order to get to a resource such as a webpage.

Subdomain or Host: The last part of the FQDN (Computers read the FQDN from right to left) is either a subdomain or a host. In this case, it is a subdomain as there are multiple hosts that respond to the www.linuxacademy.com subdomain. An example of a host would be webserver1.goole.com or webserver2.google.com. Each of those servers might host the www.google.com website, but when you access www.google.com, there’s a chance you could receive the site from either one. This will make a little more sense when we get to the CNAME records section further in this course.

Note: There is also a dot at the end of a URL that is typically hidden. This simulates the “root servers.” You can see this dot when creating new zone records, a process that will be explained later.

ANATOMY OF DNS

ZONES

A “zone” is a domain name for which the server answers queries. Zones can get very complicated, but essentially, there are Primary and Secondary zones. A Primary Zone is a zone for which the server is “authoritative.” This means the server has the final say in the DNS records it hosts. A Secondary Zone is a zone for which the server can respond, but does not have the final say. It must communicate with the Primary zone to ensure its records are correct. Having multiple DNS servers helps keep the DNS network resilient to attacks and failures.

Let’s say you have a Name Server for google.com. If google.com was a town, this server has the phone book for all of the townspeople. Mr. www.google.com, Ms. mail.google.com, and so on.