Computer Networks - Top Down approach - Network Layer

Saurabh Barjatiya

2012-03-20 Tue

Contents

1 Network Layer

Third layer in TCP/IP network model is network layer. We had looked at Transport Layer (TCP / UDP) and Application Layer (HTTP, SMTP, DNS) in last lecture. In this lecture we will look at network layer (IP, specifically IPv4). We will study both working of IPv4 (broad overview) and various features provided (or not provided) by IPv4 to upper transport layers.

1.1 IPv4 addresses

IPv4 addresses are 32-bit or 4-byte long. They are usually written in dotted-decimal notation as set of four integers between 0 and 255 (both inclusive) separated by dots. Example IP addresses are: 1.1.1.1, 0.0.0.0, 10.5.1.222, 255.255.255.255, 255.255.255.0, etc.

1.2 Netmask

Netmasks are used to group several set of IP addresses together. When converted to binary form netmasks would have all leading bits 1 followed by all 0s. Hence there are 33 possible netmasks (not all of them are practically useful). Example netmasks are 255.0.0.0, 255.254.0.0, 0.0.0.0, 255.255.255.255, etc.

Netmasks are also written using slash*() notation. In slash()* notation we write slash(/) followed by number of 1-bits in netmask. For example 255.0.0.0 can also be written as /8. Similarly 255.255.255.255 can also be written as /32.

When bits in IPv4 are ANDed with bits in netmask then we get network number or network address of current host. All hosts which have same network number are in same network and do not need a intermediate device (router / L3 switch / Gateway) for them to communicate with each other.

For example consider two addresses 10.5.1.222/24 and 10.5.1.137/24. When we calculate network numbers for both of these addresses we will get 10.5.1.0 (or to be more precise 10.5.1.0/24). Since both addresses have same network number, they belong to same network or to be more precise, they belong to same subnet and hence can communicate with each other directly without having any intermediate L3 device.

Another example can be 10.3.3.187/24 and 10.5.1.222/24. In this case we get two different network numbers 10.3.3.0/24 and 10.5.1.0/24. Hence machine with given set of IP addresses and corresponding netmasks cannot talk with each other directly and need an intermediate L3 device usually referred to as gateway or router.

It should be noted that netmasks play crucial role in determining which machines are in same network and which are in different network. For example if we change netmask from /24 to /8 in earlier example then both IP addresses would belong to network 10.0.0.0/8 and hence would be able to communicate directly without requiring any intermediate device.

1.3 Address Classes and CIDR (Class-less Inter-Domain Routing)

Historically only /8, /16 and /24, etc. were allowed to be used as netmasks and intermediate values like /23 and /12 were not allowed / supported. That form of addressing was known as classful addressing. Following different classes existed at time of classful addressing -

IP address starting bits	Class	IP range	Netmask
0	A	0.* to 127.*	/8
10	B	128.* to 191.*	/16
110	C	192.* to 223.*	/24
1110 (multicast)	D	224.* to 239.*	
1111 (reserved)	E	240.* to 255.*	

But this caused very rapid depletion of IP addresses as there was huge wastage of IP addresses. Hence CIDR or Class-less Inter-Domain Routing (pronounced sider) was introduced which allowed greater flexibility in netmasks.

1.4 Gateway

Now given that two machines with IP addresses like 10.3.3.187/24 and 10.5.1.222/24 cannot talk with each other directly without going through a intermediate L3 device, how do we make machines in different networks communicate? Also what is special about this gateway device that it can communicate with both machines although they are in different networks?

In order for gateway to be able to communicate with both machines gateway device has multiple interfaces (or multiple virtual interfaces) each having a different IP address. Now one of the interfaces of gateway device (say 10.3.3.1/24) can be in 10.3.3.0/24 network while other interface (say 10.5.1.1/24) can be in 10.5.1.0/24 network. In this setup gateway device would be able to communicate with both 10.3.3.187/24 and 10.5.1.222/24 directly.

Even more now 10.5.1.222/24 and 10.3.3.187/24 can send packets to each other via 10.5.1.1/24 and 10.3.3.1/24 respectively. Gateway device an internally forward packets from 10.3.3.1/24 to 10.5.1.1/24 and vice-versa as it is one single physical device with common RAM, processor etc. and does not requires any intermediate device for this transfer.

In practical scenarios we have L3 switches (or routers) which have many interfaces and not just two or three. For example Main IIIT core L3 switches are capable of having more than 1000 virtual interfaces and hence can be used to connect more than 1000 separate networks with each other. This is just theoretical number. Due to practical limitations we do not cross limit of 64 separate networks within campus. Even today at time of this writing there are 3000+ network ports in IIIT Hyderabad divided among approx 40 different networks.

We could always have just one big network with all devices in one single

large subnet. But due to broadcasts such network would always be heavily congested. Having so many devices in single network would also cause serious security problems as many networks attacks are possible or highly facilitated when both attacker and the target are in same subnet. Thus having one large network would cause serious security problems for critical infrastructure like servers, network devices, etc.

1.4.1 Gateway demo

In class login on 10.3.3.169 and use Cisco Packet tracer to demo L3 switch as gateway between two machines.

1.5 Routing

Connecting many different networks using one L3 switch / router should now be clear. But how do we connect large number of machines (over million machines over Internet) with each other. There is no single device that is powerful enough to route traffic between a million nodes. To solve this problem we have a large number of L3 switches and routers connected to each other which help in sending packet from one node to another.

Explain static routing used in IIIT Hyderabad in terms of Nilgiri building, Vindhya building and OBH (Palash nivas).

1.5.1 Traceroute and ping

`ping` can be used to check whether some remote machine is on or not. `ping` also displays TTL value which can help in estimating how far is given machine from our machine. Different OS (Windows, Linux) and devices (PC, switches) reply with different TTL values. Fortunately these different TTL values are far apart so that we can easily tell which OS / device is sending reply and how far it is from our machine (hop count).

`traceroute` can be used to find most inter-mediate L3 devices between two machines. It should be noted that some devices can choose not to reply to traceroute requests and may show up as * in traceroute output. Some devices may be hidden / passive and will not be detectable using traceroute. Even worse some firewalls may block traceroute packets from passing. Even after all this `traceroute` is great resource in finding network topology and number of hops between our and some other node.

1.5.2 Routing demo

- Traceroute to some OBH ground floor machine and explain output.
- Traceroute to other machines in network and explain output.
- Traceroute to google and explain output

2 Configuring various network parameters

2.1 Check IP address

One can check current IP address(s) assigned on machine using:

`/sbin/ifconfig`

command.

2.2 Configure IP address

We can use `ifconfig` command to configure IP address and netmask of `eth0` etc. interfaces. Syntax of `ifconfig` command is:

`ifconfig <interface> <ipaddress> netmask <netmask>`

Example, `ifconfig eth0 10.3.4.2 netmask 255.255.255.0`

2.3 Check current route / gateway

To check current gateway (in fact entire routing table) we can use

`route -n`

2.4 Configuring gateway

To configure gateway or default route, when there is no gateway configuration at present, we can use:

`route add default gw <gateway_ip>`

Example, `route add default gw 10.3.4.1`

2.5 Adding routes

To add specific routes for a given destination / mask combination via a gateway device we can use:

```
route add -net <destination_network> netmask <mask> gw <gateway_address>
```

Example, `route add -net 10.6.0.0 netmask 255.255.0.0 gw 10.4.2.208`

3 Configuration files

All the configuration that we do using commands as mentioned in previous section is not persistent across reboots. Hence, if for some reason system is rebooted all the address and routing configuration will get deleted. To make configuration persist across reboots, configuration changes have to be done to configuration files. In this section configuration files and what changes need to be done to be them is explained.

Note - All settings explained in this section are with respect to Cent-OS / Fedora / RHEL. Location and syntax of configuration of other OS like Ubuntu / Suse would most probably be different.

3.1 IP address and gateway configuration

IP address and default gateway are configured using files in `/etc/sysconfig/network-scripts/` folder with name `ifcfg-<interface>`. For example configuration file for `eth0` would have name `ifcfg-eth0`.

3.1.1 DHCP based addressing

Configuration file that uses DHCP for obtaining address may look like:

```
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
HWADDR=00:21:91:79:d0:e3
TYPE=Ethernet
PEERDNS=yes
```

3.1.2 Static addressing

Configuration file that uses static address may look like:

```
DEVICE=eth0
BOOTPROTO=static
IPADDR=10.3.4.18
NETMASK=255.255.255.0
GATEWAY=10.3.4.1
ONBOOT=yes
HWADDR=00:21:91:79:d0:e3
TYPE=Ethernet
PEERDNS=yes
DNS1=192.168.36.222
```

3.2 Route configuration

Permanent route configuration is done in file with name `route-<interface>` for example permanent route configuration related to `eth0` would be done in file `route-eth0` located in same folder (`/etc/sysconfig/network-scripts`). Sample `route-eth0` file may look like

```
GATEWAY0=10.3.3.1
NETMASK0=255.255.255.0
ADDRESS0=192.168.1.0
```

4 IPv4 vs IPv6

IPv4 protocol explained in this lecture has reached its age and must soon be replaced by newer versions like IPv6. IPv6 is much better protocol and device / OS support IPv6 is present in all major devices and Operating Systems. Migration to IPv6 is still pending due to:

- Lack of IPv6 knowledge among users
- Lack of IPv6 knowledge among administrators
- Lack of support / porting of large number of IPv4 dependent applications to IPv6
- DHCP, NAT, etc. which helped in reducing demand of IPv4 address space for many years

- Lack of ISP support / government support in developing countries like India for IPv6 addressing / networking

Students are advised to go through and understand IPv6 protocol on their own using various resources on Internet. The best and easiest resource to start from is 6deploy video tutorials hosted at http://www.6deploy.eu/e-learning/english/index.php