Contents

github.com/rkpust/Computer-Networks

github.com/rkpust/Computer-Networks

Networking Fundamentals

Here's a brief overview of each topic:

Network Device: A network device is any hardware component that connects computers and other devices in a network to allow communication. Examples include routers, switches, hubs, and bridges.

Network: A network is a collection of interconnected devices (computers, servers, routers, etc.) that can communicate with each other to share resources and information. It can be as small as a local area network (LAN) or as large as the internet.

Host: A host is any device connected to a network that can send, receive, or store data. Typically, this refers to computers, servers, or even IoT devices that communicate on the network.

IP (Internet Protocol): IP is a set of rules used to identify devices on a network and route data packets between them. It assigns unique IP addresses to hosts so that they can be found and communicated with. Two versions of IP are commonly used: IPv4 and IPv6.

Hub: A hub is a simple network device that broadcasts data to all devices connected to it. It doesn't have the intelligence to direct data traffic efficiently, often leading to network congestion. Hubs are now largely replaced by more efficient switches.

Bridge: A bridge is a network device that connects two separate networks, allowing them to communicate and function as one. It filters and forwards data based on MAC addresses, improving network traffic management and reducing collisions.

Switch: A switch is similar to a hub but more intelligent. It directs data traffic only to the device that needs it, using MAC addresses to efficiently manage data transmission and reduce network congestion. Switches are widely used in modern networks.

Router: A router is a device that connects different networks (such as a local network to the internet). It routes data packets between devices on different networks based on their IP addresses, ensuring data reaches its correct destination.

github.com/Keust/Computer-Networks

OSI Model

The OSI (Open Systems Interconnection) model is a conceptual framework that standardizes the functions of a communication system into seven distinct layers. Each layer serves a specific role in facilitating network communication.

Here's a quick rundown of the seven layers from the top (Layer 7) to the bottom (Layer 1):

Application Layer (Layer 7): The layer closest to the end user. It provides network services to applications. Examples: HTTP, FTP, DNS.

Presentation Layer (Layer 6): Responsible for data translation, encryption, and compression. It ensures that data is in a usable format (like converting between different character encodings).

Session Layer (Layer 5): Manages and controls the dialog between two devices. It establishes, maintains, and terminates connections.

Transport Layer (Layer 4): Provides end-to-end communication and data flow control. It ensures complete data transfer, including error correction and retransmission. Examples: TCP, UDP.

Network Layer (Layer 3): Determines how data is routed across networks and manages addressing. Responsible for logical addressing (like IP addresses). Example: IP.

Data Link Layer (Layer 2): Provides node-to-node data transfer and handles error detection and correction. This layer deals with MAC addresses and frames. Example: Ethernet, Wi-Fi.

Physical Layer (Layer 1): The lowest layer, which deals with the physical aspects of transmitting data over a medium (e.g., cables, radio waves). It defines electrical, mechanical, and procedural specifications.

Each layer serves a distinct function, and they work together to enable data communication across networks.

github.com/rkpust/Computer-Networks

IP Classification

IP addresses are classified into different categories based on their range and intended use. Here's a brief note on each classification:

IPv4 Address Classes

IPv4 addresses are divided into five primary classes (A, B, C, D, E), but the first three are most commonly used for public and private networks:

Class A:

- **Range**: 1.0.0.0 to 127.255.255.255
- **Default Subnet Mask**: 255.0.0.0

- **Usage**: Designed for large networks (up to 16 million hosts).
    - **Public IP**: Yes, part of the public IP address space.
    - **Example**: 10.0.0.1

Class B:

- **Range**: 128.0.0.0 to 191.255.255.255
- **Default Subnet Mask**: 255.255.0.0
- **Usage**: Medium-sized networks (up to 65,000 hosts).
- **Public IP**: Yes, part of the public IP address space.
- **Example**: 172.16.0.1

Class C:

- **Range**: 192.0.0.0 to 223.255.255.255
- **Default Subnet Mask**: 255.255.255.0
- **Usage**: Small networks (up to 254 hosts).
- **Public IP**: Yes, part of the public IP address space.
- **Example**: 192.168.1.1

Class D (Multicast):

- **Range**: 224.0.0.0 to 239.255.255.255
- **Usage**: Used for multicast communication (group communication to multiple receivers).
- **Public IP**: Yes, but reserved for multicast applications.
- **Example**: 233.0.0.1

Class E (Experimental):

- **Range**: 240.0.0.0 to 255.255.255.255
- **Usage**: Reserved for experimental purposes and future use.
- **Public IP**: No, not used for public networks.
- **Example**: 250.0.0.1

Private IP Addresses

Some IP address ranges are reserved for use within private networks and are not routable on the public internet. These are used for internal communication and require NAT (Network Address Translation) to connect to the internet.

- **Class A (Private)**: 10.0.0.0 to 10.255.255.255
- **Class B (Private)**: 172.16.0.0 to 172.31.255.255
- **Class C (Private)**: 192.168.0.0 to 192.168.255.255

github.com/rkpust/Computer-Networks

Loopback Address

- **Address**: 127.0.0.0 to 127.255.255.255
- **Usage**: Reserved for testing and diagnostics on the local device (loopback interface).
- **Example**: 127.0.0.1 (often referred to as "localhost").

Summary of Common Classes:

- **Class A**: Very large networks (16 million hosts).
- **Class B**: Medium-sized networks (~65,000 hosts).
- **Class C**: Small networks (~254 hosts).
- **Class D**: Multicast communication.
- **Class E**: Reserved for experimental use.

The division of IP addresses helps ensure efficient use of address space and routing in both local and wide-area networks.

Subnetting

The default subnet mask for a Class A IP address like **10.251.215.42** is **255.0.0.0** (or **/8** in CIDR notation).

Let's calculate each of the required elements:

Network Address

The network address is obtained by performing a bitwise AND operation between the IP address and the subnet mask.

IP Address (10.251.215.42)
In binary: `00001010.11111011.11010111.00101010`

Subnet Mask (255.0.0.0)
In binary: `11111111.00000000.00000000.00000000`

Performing the AND operation:

```
00001010.11111011.11010111.00101010
AND
11111111.00000000.00000000.00000000
-----------------------------------
00001010.00000000.00000000.00000000
```

The result is: **10.0.0.0**

github.com/rkpust/Computer-Networks

So, the **Network Address** is **10.0.0.0**.

Gateway

The default gateway is typically the first usable IP address in the network, which is usually **10.0.0.1** (or any other address the network administrator has chosen, but this is common).

Broadcast Address

The broadcast address is the last address in the subnet, which is found by setting all the host bits (the bits after the network portion) to 1.

For **10.0.0.0/8**, the broadcast address is:

Broadcast Address: **10.255.255.255**

Usable Hosts

The usable hosts are the addresses between the network address and the broadcast address, excluding both of those.

For a **/8** subnet, the number of usable host addresses is:

- Total addresses: **2^24** (since 24 bits are used for hosts)
- Usable addresses: **2^24 - 2** (subtracting the network and broadcast addresses)

This equals **16,777,214** usable host addresses.

Summary

- **Network Address**: 10.0.0.0
- **Gateway**: 10.0.0.1 (typically)
- **Broadcast Address**: 10.255.255.255
- **Usable Hosts**: 16,777,214 hosts in total

github.com/rkpust/Computer-Networks

Routing Types

Static Routing

- **Definition**: In static routing, routes are manually configured by a network administrator and do not change unless manually updated.
- **Pros**: Simple, secure (no automatic updates), and predictable.
- **Cons**: Doesn't adapt to network changes or failures automatically, requiring manual intervention to update routes when the network changes.
- **Use Case**: Small, stable networks where the topology doesn't change often.

Dynamic Routing

- **Definition**: Dynamic routing protocols automatically discover and adjust routes based on network changes or failures.
- **Pros**: Adaptive to changes, no manual configuration required for routes.
- **Cons**: More complex to configure and manage, and may introduce security risks if not properly secured.
- **Use Case**: Larger, more complex networks where the topology may change or evolve over time.

Default Routing

- **Definition**: A default route is used when there is no specific route to the destination. Routers will forward traffic to a pre-configured "default" route when they don't know how to reach the destination.
- **Pros**: Simplifies routing when the destination is not directly connected.
- **Cons**: It could lead to traffic being forwarded to the wrong destination if not configured correctly.
- **Use Case**: Often used in situations like connecting a network to the internet.

Static vs. Dynamic Routing

- **Static**: Routes are manually defined, suitable for small or simple networks.
- **Dynamic**: Routes are learned and updated automatically through protocols like RIP, OSPF, and BGP, ideal for large or changing networks.

github.com/lkpdst/Computer-Networks

Routing Protocol

Here's a brief overview of the dynamic routing protocols **OSPF** and **BGP**:

OSPF (Open Shortest Path First)

- **Type**: Link-State Protocol

- **Function**: OSPF is used for routing within a single Autonomous System (AS). It allows routers to share information about the state of their links (such as bandwidth, cost, and delays), which helps them build a complete network map to calculate the shortest path.
- **Key Features**:
    - **Uses Dijkstra's Algorithm** to compute the shortest path tree.
    - **Supports hierarchical network design** with areas, which improves scalability.
    - **Converges quickly** when the network topology changes.
    - **Classless** protocol (supports CIDR and VLSM).
    - OSPF is more efficient in larger, complex networks.
- **Usage**: Primarily used in enterprise networks and large organizations for internal routing.

BGP (Border Gateway Protocol)

- **Type**: Path-Vector Protocol
- **Function**: BGP is the primary protocol used for routing between different Autonomous Systems (ASes), such as on the internet. It exchanges routing information between different networks and selects the best path based on policy rather than just distance.
- **Key Features**:
    - **Uses Path Selection**: BGP selects routes based on policies (AS path, prefix length, next-hop, etc.).
    - **Scalable**: BGP can handle large routing tables (thousands of routes) and is fundamental for internet-wide routing.
    - **Slow convergence**: BGP converges slower than OSPF but is more robust for managing routing on a global scale.
    - **Supports Classless Inter-Domain Routing (CIDR)**.
    - **Can be used for routing control** by applying filters and policies, making it highly flexible.
- **Usage**: Used between ISPs, large enterprise networks, and the global internet to exchange routing information.

Key Differences:

- **OSPF** is **link-state**, meaning it calculates the shortest path based on the network topology and link conditions within a single AS.
- **BGP** is **path-vector**, focusing on policy-based routing between multiple ASes, making it the core protocol for internet routing.

Both OSPF and BGP are crucial for dynamic, scalable routing in modern networks. OSPF is ideal for internal networks, while BGP handles inter-network (internet) routing.

github.com/rkpost/Computer-Networks

Some Commands

Link Down

0. Login to the device and run the following cmd for checking uptime

*** Cisco device

[show version]

*** Huawei device

[display version]

1. Show two side interface Rx Power

i. Go to device and run the following cmd

[show interface description | include portnumber]

- show port status and remote device name from interface description

ii. run the following cmd

[show controllers portnumber phy]

[show controller optics 0/0/0/39 summary] or [sh controllers tenGigE 0/0/0/16 phy] for xe N55A1 series

[sh int transceiver] or [show interface gigabitEthernet 0/0/10 transceiver]

for cisco ASR-920-12CZ-D model

iii. [show logging | include 0/1/0/48]

iv. [show running-config interface Gi0/0/0/10] check 'negotiation auto' for 1G port, if not then configure

[show running-config interface vlan 900]

v. [sh arp vrf vrf_name] show mac

[ping vrf RGL_DSE 172.31.0.9]

[ping vrf RGL_DSE 172.31.2.53 source 172.31.0.10]

[show ip route vrf DHAKA_COM]

vi. check l2 circuit is up or down

github.com/rkpust/Computer-Networks

[show l2vpn xconnect interface GigabitEthernet0/0/0/9] or [show l2vpn xconnect pw-id 3981 detail]

or [show l2vpn xconnect] or [show l2vpn xconnect group] or [show l2vpn xconnect group BDCOM]

or [show l2vpn xconnect group BDCOM detail]

[sh mpls l2transport vc vcid 3382]

vii. [show mac address-table dynamic vlan 1544] show mac for vlan, only for vrf

2. Go go second device from interface description, let interface (0/0/13)

*** For Cisco XE

i. Go to device and run the following cmd

[show interface description | include portnumber]

- show port status from interface description

ii. [show hw-module subslot 0/0 transceiver 13 status]

- show Transceiver Rx optical power

iii. run the following cmd for time (0/0/13)

[show log | in portnumber]

[sh bridge-domain 3998] show mac address in aggregation

*** For Huawei

let interface GigabitEthernet 0/1/3

i. [display interface]

ii. or [display interface GigabitEthernet 0/1/3] or [display interface 25GE1/0/12 transceiver brief]

iii. [display mac-address interface 25GE1/0/12] [display current-configuration interface 25GE1/0/12]

github.com/rkpust/Computer-Networks

[display mac-address dynamic vlan 90] for switch

[display arp vpn-instance RGL_DSE slot 3] let interface GE3/1/8, and it has vpn-instance in configuration

[ping -vpn-instance RGL_DSE 172.31.2.54] ping to client device

[dis ip routing-table vpn-instance RGL_DSE]

[display current-configuration configuration vsi INFOLINK] for vsi on router, to see mac

[display mac-address dynamic vsi INFOLINK] for vsi on router, to see mac

[dis transceiver diagnosis interface gig0/0/18] for switch

[display transceiver interface GigabitEthernet 0/0/5] or

[display interface 25GE 1/0/9 transceiver verbose] to check Transcevier info like SFP type, length,etc

[display logbuffer | include 3/1/4] or [dis log | i 0/2/14 | i OperStatus]

[dis mpls l2vc 2128] to show link uptime

[display ospf peer] show ospf neighbour

show Rx Power

Who and What changed

[display configuration commit list] and [display configuration commit changes at 1000000199]

SFP info

[display elabel] or [display elabel brief] or [display elabel optical-module brief]

10BaseT means copper port, so no tranceiver power

Packet Loss

show all respectively and client's max utilization in cacti for specific time

[show version] ; uptime

i. [show log | in 0/0/7] ; log

[show running-config interface gigabitEthernet 0/0/7] ; configuration, find encapsulation dot1q number

github.com/rkpust/Computer-Networks

[show policy-map interface gig0/0/7] ; policy map

[show policy-map interface gig0/0/7 output] ; policy map for output P1, means CS, find out drop

[display qos-profile statistics interface gi0/2/28 vlan 3840 outbound] ; policy map for output be,af2,cs6,

means CS, find out drop

[show running-config interface bdi 2502] ; dot1q and bdi are same number, find vrf

[show ip route vrf GP_NRG_LTE_PayLoad_VRF] ; which ip has maximun times in lastmile this is aggregation

[show running-config interface loopback0] ; loopback0, loopback1, loopback2 for seeing own device ip, means

lastmile

[tracert in naas tool lastmile to agg and vice versa] ; find route is symentric or asymentric

[ping 10.255.255.185 source 10.253.203.1 size 8000 df-bit repeat 10000] ; ping lastmile to agg; if version xe

size 8000, others 9000

[show ip route 10.255.255.185] ; find out route learning time for lastmile and each hop, login to each hop, find anomaly,

if any anomaly then login and check the uptime and log

i. [show interfaces description | i 121990]

ii. [show interfaces gigabitEthernet 0/0/0/11]

iii. [show controllers optics 0/0/0/11 summary]

iv. [show log start today | i 0/0/0/11]

v.

vi. [show ip ospf neighbor]

vii. [show ip bgp vpnv4 unicast summary]

viii. [show running-config interface gigabitEthernet 0/0/0/11]

ix. [show l2vpn xconnect interface gigabitEthernet 0/0/0/11]

x. [show l2vpn xconnect interface gigabitEthernet 0/0/0/11 detail]

github.com/rkpust/Computer-Networks

Find Ring

2Sw:10.251.215.42

2SwP:GE1/0/46

SCR: 117331

[display current-configuration]

or

[display current-configuration interface 25GE1/0/46]

find vlan like 2370

[display vlan 2370] For Switch

[display mpls l2vc 1544] for Router

[show l2vpn xconnect pw-id 1544] for Cisco Router

find port like Trunk41

[display interface Eth-Trunk41]

find description like 'Description: 106904:FHL:L2_L3-BKB:GZ-BHTC-CO-02-N5504-PE-05:10.255.254.199:BE41:FIBER-PC'

now go to remote IP and port find next (continue)

Loss Calculation

Let a device Tx -19 dBm, another device Rx -11 dBm and Fiber 20km between among them.

Loss = (Fiber * 0.4) + 2 = 10 dBm

Calculated loss is tolerable, 2 is added for a device has 2 connectors and another device has 2 connectors, each connector loss is 0.5

github.com/rkpust/Computer-Networks