Routing

1. Overview: layering, routed versus routing protocols, switching and routing, IP addressing.

MNEMONICS!!!

Layering > OSI MODEL >

All	People	Seem	To	Need	Domino's	Pizza
Application	Presentation	Session	Transport	Network	Data-Link	Physical

Encapsulation >

Don't	Don't	Don't	Stop	Pouring	Free	Beer
Data	Data	Data	Segments	Packets	Fragments	Bits

Routing protocols distribute routing information throughout all routers on a network. [OSPF](), [RIP](), [EIGRP]() or [BGP]().

A **routed** protocol is a protocol by which data can be routed. Routed protocols can be sent over a routed network. Today, with [IP]() (think of[TCP/IP]()) being the predominate protocol in use on the Internet and in most networks. IP is designed to be routed over and through different networks.

A **routed** protocol is a layer 3 protocol that applies logical addresses to devices and routes data between networks.

Routing protocol provides information. Routed protocol uses this information

Switching packets inside the subnet > MAC Addresses (Mac table) on Layer 2
Routing - routes packets between the subnets, inside the network > IP Addresses > using the network ID from the header then comparing it to the routing table. on Layer 3 (more intelligent switching)

Source and destination IP stay the same the entire life of a packet (in general), but MAC addresses are re-written every time they pass through a router.
To put it in its simplest terms, devices on one network will **NOT** learn the MAC address of devices on another network.

You really need to have the OSI model in mind and the encapsulation that's done and what's added in each Layer in the PDU.
PC1 fist will do **ANDING** using the PC2's IP address with it's Subnet Mask to see if they are on the same subnet, if they **aren't** the PC1 will send the data to gateway (R1) but if PC1 doesn't have in its ARP table the MAC address of R1 it will do ARP and then actually send the data.
R1 receives the data checks the packet to see if it destined for him. He looks that the packet isn't for him because the Dest IP isn't his so he checks his routing table to see if he has a route for that Dest IP. If he has a route he sends the data to R2 but if he hasn't R2's MAC address in his ARP table he does ARP and then forwards the data.
R2 receives the data and does the same thing and sends it to PC2.
As you can see the Source/Destination IP stay the same but Source/Destination MAC change.

IP Addressing:

Hardware address: HARDCODED ! used to uniquely identify a host within a local network. Function of the **Data-Link (Layer 2)**.

Ethernet uses Media Access Control (MAC) = 48 bits (hex), first SIX hex digits = **manufacturer**, the last SIX = **host ID**

>> L2 network just by switches = flat network, no hierarchy = ENORMOUS broadcast domain

The scalability limitations of Layer-2 hardware addresses are mitigated using logical addresses.
Logical Address: Network Layer 3 and provides hierarchy, not hardcoded, but CAN BE dynamically assigned and changed.

Two components: **Network ID + Host ID (+ SubnetMask** which identifies the network)
32 bits = 4 Octets!
Hosts on different networks require a router to communicate:

Host A: 158.**80**.164.100 255.255.0.0
Host B: 158.**85**.164.101 255.255.0.0

IP Address Classes:

Class A	1 - 127	255.0.0.0	127 Net	16.777.214 Hosts	0xxx
Class B	128 -191	255.255.0.0	16384 Net	6534 Hosts	10xx
Class C	192 - 223	255.255.255.0	2097152 Net	254 Hosts	110x
Class D	224 - 239	**Muticast**	**Traffic**	**-**	1111

Too much useful addresses are lost due to the lack of flexibility!!!

On each IP network, **two** host addresses are reserved for special use:
Network address (usually the first address of the spectrum). It's used in routing tables and contains all **0** in the host portion.
Broadcast address (usually the last address of the spectrum). == all hosts on the network. Contains all **1** in the host portion.
Router **never** forwards a multicast or broadcast packet from on interface to another
Switch **always** forwards multicast or broadcast packets to every port, except the original port the signal came from

Subnetting is the process of stealing bits from the host portion of the subnet mask. **More** hosts = **less** networks.

To get 10 new network we need 2^n >= 10 > 2^4 = 16 networks > we need to steal 4 bits from the hosts octet > 255.255.255.**240 or /28** (24+4)
2^4 = 16 - 2 = 14 available hosts per network
0.0.0.0/0 - reserved to identify all networks **= default route**.

Internet Protocol (IP) provides two fundamental Network layer sevices:
Logical Addressing - provides unique address (network id + host id)
Routing - determines the best path to destination and routes data accordingly

===

2. Overview: IP addressing, VLSM (Variable Length Subnet Mask) and CIDR (Classless Inter-Domain Routing).

CIDR (Classless Inter-Domain Routing) - simplified method of representing a **subnet mask. CIDR** identifies the number of bits set to **1 (on)** preceded by a slash. 192.168.1.1/24 or /30 ...

VLSM allows the use of different masks for each subnet. After a network address is subnetted, those subnets can be further subnetted. VLSM is simply subnetting a subnet. VLSM can be thought of as sub-subnetting.
Subnetting 10.0.0.0/8 to 10.0.0.0/16, /16 to /24...

===

3. Introduction to Routing and Packet Forwarding

Routing is the process of forwarding packets of information from one network to another. Therefore routes are based on the destination **network**, and NOT the destination host. The routing decisions are based on a routing table (Destination IP is checked against the routing table). If there is no route in the table, they will send the packet to the **default gateway**.

Routing tables contain:
The destination network and it's subnet mask
The "next hop" router to the destination network
Routing **metrics** and **administrative distance (AD)**

To determine the best route to a destination, elements are considered in the following order:

Prefix length - number of bits used to identify the network (the more, the better) > ***ALWAYS PREFERRED !***

Metric (within a routing protocol) - **Distance** vs **Cost**

Administrative distance (between separate routing protocols)

Only routes with the **best metric** are **added** to the routing table. If rotes have the same metric, most protocols will load-balance

If a router is running multiple routing protocols, **Administrative Distance** is used to determine which one is to be trusted.(Lower = better).

Connected	0
Static	1
Eigrp Summary	5
External BGP	20
Internal EIGRP	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
External EIGRP	170
Internal BGP	200
Unknown	**255**

Unknown Is never inserted into the routing table!

Router# *show ip route* --show the routing table

[120/1] = 120 = 120 AD; 1 = hop-count

Router# *clear ip route** --clear the table, force the protocols to repopulate it

==

4. Static Routing. Default Routes and On-Demand Routing.

A static routing table is created, maintained and updated by a network administrator MANUALLY.

A static route to every network must be configured on EVERY router for full connectivity. This provides superior control over routing, but quickly becomes impractical on large networks. Also DOES NOT adapt to network changes > more work

Routers will NOT share STATIC routes thus reducing CPU/RAM overhead and saving bandwidth.

Static routes have an **AD of 1** and are always preferred over dynamic routes, UNLESS the default **AD** is changed.

A static route with an adjusted **AD** is called a floating **static route** serving as a **backup** to a static route.

A dynamic routing table is created, maintained and update by a ROUTING protocol running on the router (RIP, EIGRP, OSPF)

Routers SHARE dynamic routing information with each other, thus increasing CPU and RAM and bandwidth usage. They are capable of dynamically choosing a different (better) path in case the network changes.

Static routing: advantages: minimal CPU/RAM overhead; no bandwidth overhead (updates are not shared); better control on routes

Disadvatages: changes have to be manually configured; no dynamic changes; impractical on large scale

Dynamic routing: advantages: simpler to configure on large scale; dynamically chooses a new/better path; **can load balance**

disadvantages: updates are shared - consume resources; calculations put load on CPU/RAM; no strict choice of best route

On-Demand Routing:

ODR (On Demand Routing) is designed to be used in a partially meshed environment (e.g Frame Relay networks) where a hub router maintains one link each to multiple stub routers (spokes routers). Therefore, for any spoke to communicate with another spoke, such traffic must pass through the hub.

On-Demand Routing (ODR) is an enhancement to Cisco Discovery Protocol (CDP), a protocol used to discover other Cisco devices on either broadcast or non-broadcast media. With the help of CDP, it is possible to find the device type, the IP address, the Cisco IOS®

version running on the neighbor Cisco device, the capabilities of the neighbor device, and so on. ODR is able to carry Variable Length Subnet Mask (VLSM) information.

	Dynamic Routing	Static Routing
Configuration Complexity	Generally independent of the network size	Increases with network size
Topology Changes	Automatically adapts to topology changes	Administrator intervention required
Scaling	Suitable for simple and complex topologies	Suitable for simple topologies
Security	Less secure	More secure
Resource Usage	Uses CPU, memory, link bandwith	No extra resources needed
Predictability	Route depends on the current topology	Route to destination is always the same

==

5. A Routing Table.

There are two common types of static routes in the routing table:
- Static route to a specific network
- Default static route = default gateway **ip route 0.0.0.0 0.0.0.0**{*exit-intf* | *next-hop-ip*}

S = static route; * = possible default route

```
Router(config)# ip route network-address subnet-mask
{ip-address | exit-intf}
```

Parameter	Description
network-address	Destination network address of the remote network to be added to the routing table.
subnet-mask	• Subnet mask of the remote network to be added to the routing table. • The subnet mask can be modified to summarize a group of networks.
ip-address	• Commonly referred to as the next-hop router's IP address. • Typically used when connecting to a broadcast media (i.e., Ethernet). • Commonly creates a recursive lookup.
exit-intf	• Use the outgoing interface to forward packets to the destination network. • Also referred to as a directly attached static route. • Typically used when connecting in a point-to-point configuration.

```
File Edit View Terminal Help

R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     172.31.0.0/24 is subnetted, 3 subnets
C       172.31.1.0 is directly connected, Loopback1
C       172.31.14.0 is directly connected, Serial0/2
C       172.31.123.0 is directly connected, FastEthernet1/0
R1#
```

Route Summarization:

It's meant to make the job of the router easier. Less processing needed to determine a route for a packet. But with modern CPUs in routers, it really isn't an issue.

The only benefit besides that is possibly easier to read routing tables, but that's arguable. With auto summary you can also have two groups of similar classless subnets on each side of a network, and when summarized to their classful boundaries, end up overlapping each other and causing a routing conflict/loop.

==

6. Introduction to Dynamic Routing Protocols. ***Route Filtering ?***. Interior Gateway Protocols (IGP) and Exterior Gateway Protocols (EGP). Distance Vector versus Link State Routing Protocols.

Dynamic Routing Protocols:

- Discovery of new networks
- Auto router update
- Best path determination
- Failover ? - load balancing
- **NO HUMAN ERROR**

Two categories:

Distance-vector protocols - RIP, IGRP - use **distance**

Link-state protocols - OSPF, IS-IS - use **cost**

EIGRP = HYBRID!

Interior Protocols: within **AUTONOMOUS SYSTEM** (subnets under the control of a single administrative entity)

Exterior Protocols: are used for exchanging routing information between **AUTONOMOUS SYSTEMS**, such as Border Gateway Protocol.

They rely on IGPs to resolve routes within an AS

==

7. Distance-Vector Routing Protocols.

Use **distance/hop count** as their metric: how routers away is the network

RIP, IGRP

Periodic updates of the **full** routing table are sent to routing neighbors

Distance-vector protocols suffer from slow convergence and are prone to loops

Some form of 'distance' is used to calculate a route's metric.

Bellman-Ford algorithm is used to determine the shortest path

A distance-vector routing protocol begins by advertising directly-connected networks to its neighbors.

RIP - every 30 sec; IGRP - every 90 sec. Each neighbor adds the routes from the updates to their routing table, they trust the update

completely, and will forward their routing table (connected and learned routes) to every other neighbor. **Routing by rumor.**

Because of the **periodic** updates, **distance-vector** protocols converge **slowly.** Also, trusting **blindly** their neighbors, they are highly **susceptible** to **routing loops**.

Distance-vector protocols utilize some form **of distance** to calculate a route's metric.

RIP uses **hop-count. IGRP** uses a composite of **bandwidth and delay.**

===

8. Link-State Routing Protocols.

Use **cost** as their metric: **OSPF, IS-IS**

SHARES only link info individually!

Link-state routing protocols were developed to alleviate the **convergence** and **loop issues** of **distance**-**vector** protocols.

They maintain three separate tables:

- **Neighbor table** - contains a list of all neighbors, and the interface each neighbor is connected to.

  Neighbors are formed by sending **Hello** packets.
- **Topology table** - otherwise known as the **link-state** table - contains a map of all links within an **area,** including each link's status.
- **Shortest-path table** - contains the best routes to each particular destination (a.k.a. **routing table**)

Link-state protocols do NOT 'route by rumor' - they send updates advertising the **state** of their links. (**link** being a directly connected network). All routers know the state of all existing links in their **area** and store this information in a **TOPOLOGY** table.

All routers within an area have *identical* topology tables.

The best route to each link (network) is stored in the **shortest-path (routing) table**. If the state of a link changes (interface failing), and advertisement containing **ONLY this link-state change** will be sent to all routers in the **area.** Each router will adjust its **topology** table accordingly and will calculate a new **best** route if required.

Consistent **topology tableS** means **link-state** protocols can CONVERGE very quickly and are IMMUNE to routing loops.

Also, because updates are sent only after link-state changes, and contain only the change, they are LESS bandwidth-intensive, than distance-vector protocols.

However, the three **tables** utilize more resources on the router itself.

For the metric, the **Dijkstra formula** is used to determine the shortest path.

===

9. Routing Information Protocol

RIP (Routing Information Protocol) is a standardized Distance Vector protocol, designed for use on smaller networks, and is widely supported.

Timers: (ALSO VALID FOR RIPv2) !!!

Update Timer - **30** seconds (default) - sending updates to **BROADCAST (v1) or MULTICAST (v2)**

Invalid Timer - **180** seconds - how long a route remains in the routing table before marked as ***invalid*** if no updates are heard about this route. If an update is received, the timer is restarted. **INVALID = 16 METRIC = UNREACHABLE.**

Hold-down Timer - **180** second - routes in **hold-down** state:

- **Invalid** timer has expired
- Marked with **metric 16** from another update
- Marked with a **higher metric** from another update to prevent loops

Flush Timer - **240** seconds - runs with **Invalid timer** - flushed **60 secs** after marked invalid.

Timers have to be identical on all routers!!!

SLOW CONVERGENCE, because the update has to **HOP** through every connected router!

PASSIVE INTERFACES: disable MULTICAST updates FROM specific interface, but listen to INCOMING updates (if there are just end devices beyond that)

passive-interface s0

passive-interface default //all are passive

no passive-interface s0   //just s0 is not passive

RIP Neighbors: RIP sends updates AS **BROADCAST; RIPv2 AS MULTICAST 224.0.0.9 !**

We can configure RIP ***neighbors***, which allow **unicast** routing to those neighbors.

> # neighbor 10.3.5.1

BUT, the router will still **broad/multicast** in addition to the **unicast,** therefore we need passive interfaces!

># passive-interface s0

>#neighbor 10.3.5.1

Interoperating RIP and RIPv2 (**PER INTERFACE BASIS)**

># interface s0

># ip rip send version 2; #ip rip receive version 1; #ip rip receive version 1 2; #ip rip v2-broadcast

Only **on p2p** send update only on change:

># ip rip triggered

Debugging:

># show ip route (opt. 172.18.0.0); # show ip protocols (rip timers, versions..); #

Router 0# default information originate //shares the static routes (0.0.0.0!!!)

==

9.1 .RIP version 1.

>> Does NOT support VLSM! >> networks must be contiguous. (same major network & subnetmask

Characteristics: periodic router updates - **30 seconds -** as broadcasts to 255.255.255.255

send out the **full routing table** every update

it's metric is the **hopcount, max hopcount = 15 hops**

Bellman-Ford algorithm for determining the best 'path'

AD = 120

can load balance up to 4 paths with **equal metric (hopcount)** > slower links can congest

==

10. RIPv2.

RIPv2 is classless! **>> includes SM into updates > networks can be discontiguous.**

Enhancements in v2: Updates are sent via **multicast 224.0.0.9**

Encrypted authentication can be configured between routers

Route tagging is supported

RIPv2 can work with RIPv1:

- **V1 sends** only **V1** packets
- **V1 receives** both **v1** and **v2** updates
- **V2 sends both** v1 and v2 updates , but receives **ONLY V2 updates**

Unless RIPv2 is manually specified, Cisco will **default** to **RIPv1** when configuring RIP.

Configure RIPv1:

Router(config)# router rip *//enables the RIP process*

Router(config)# network 172.16.0.0 *//which networks we wish to advertise to other routers*

Router(config)# network 172.17.0.0 *//they are the networks directly connected*

==

11. Hybrid Routing Protocols. **EIGRP (Enhanced Interior Gateway Routing Protocol).**

Works with AS number! # *router eigrp 10 //for AS 10*

Metric: Bandwidth and Delay of the Line are used! [10000000/bandwidth+delay]*256

CISCO Proprietary protocol. **HYBIRD!!! Fast Convergence!**

Support VLSM.

Can run multiple protocols >> Reliable Transport Protocol !!!

RTP > supports reliable and unreliable !

Load balancing - equal or not-equal paths.

Authentication - Router ID

Note: Only EIGRP supports **unequal** cost load balancing.

Diffusing Update Algorithm (DUAL) - determines the best path and ensures loop free routing, also **backup** routes!!!

EIGRP forms **neighbor** relations with adjacent routers in the same **AS. >> Neighbor Table**

Updates are sent either by **UNCIAST or MULTICAST(224.0.0.10)** by **Reliable Transport Protocol**

Updates are sent **ONLY** when a change occurs and **ONLY** the change (Bounded Triggered Updates)

Cost = Bandwidth + Delay of the line

HOLD TIMER: how long the router waits before marking the route inactive, if it stops receiving hello packets

By default: 3 x Hello Timer, can also be adjusted per interface, and must not be the same on All routers.

Neighbor Table - IP of the router, Interface from which the HELLO came, Hold timer, sequence number

Adjacencies will not form unless the primary IPs are on the same subnet

Topology Table - saves a backup path - list of **ALL** routes in the AS

Routing Table - the best route for each known network

Packets:

Hello - unreliable ; multicast - form neighbor relationships; 5 secs on fast, 60 secs on slow links

Update - reliable; unicast(for new neighbors) or multicast (if a metric is changed)

Query - reliable; unicast or multicast - when a **Successor** route fails, and there are not **FS**, the route is in **active** state and **queries** for an alternative route.

Reply - reliable ; unicast - response to **query** - only with an alternative route **(???)**

ACK - unreliable ; unicast - acknowledges delivery - **HELLO** packets with NO DATA, just and ACK number

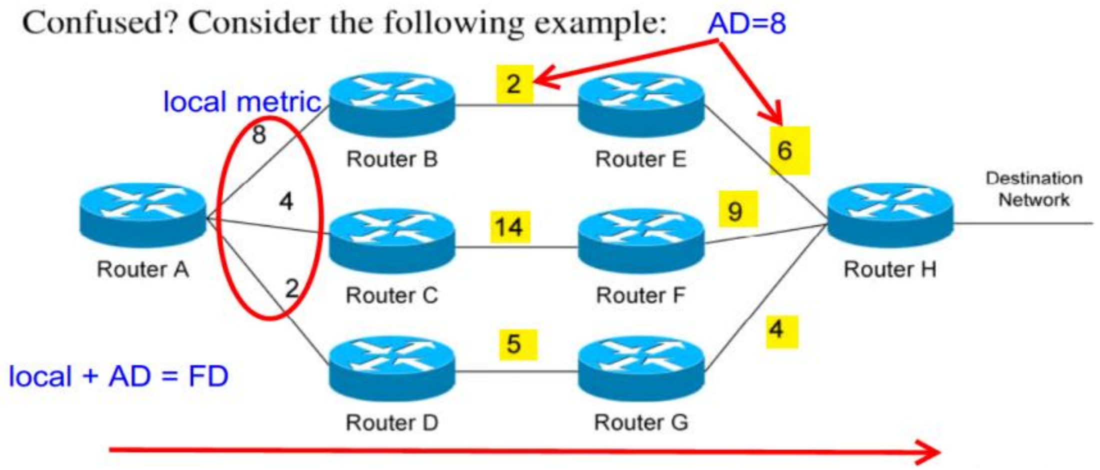
Router updates, containing all known routes and their metrics, populate the TOPOLOGY table. The router with the lowest metric will be the **feasible distance** and will be installed into the table.

Feasible distance = advertised distance (from the other router to the network) + the metric to the router itself

To converge quickly, the topology table contains also **feasible successors.** The successor's **advertised distance** Has to be less than the current **feasible distance.** This is known ad the **Feasible condition (FC).**

Routes that are not **feasible successors** become route **possibilities.**

Successors allow the routers to provide connectivity without reconverging.



We'd have 3 FDs:

8 + (2 + 6) = **16 << Feasible Successor**

4 + (14 + 9) = **27 << Possbilitiy**

2 + (5 + 4) = **11 << Smallest** goes into the table!

If no Feasible Successor exists and a link fails, a route will enter an **Active** (converging) state until an alternate route is found.

Router# *show ip eigrp topology (active) // only active*

(all-links) //all links

Passive Interfaces: but the interface will **neither SENT** nor **RECEIVE** updates or hello packets (>> no NEIGHBOR relations!!)

#passive-interface s0 //only s0 passive

#passive-interface default //every is passive

#no passive-interface s0 //only s0 will NOT be passive

Router 0# redistribute static //shares the static routes (0.0.0.0!!!)

Useful:

#show eigrp neighbors

#show eigrp interfaces // which interfaces have eigrp

==

12. OSPF.

Separates an **Autonomous System** into individual **areas** !!!

Routers build a **Topology Database** of all links **within** their area, all routers **within the same area** have identical topology DBs.

Updates contain only information about local links. >> **CONSERVES** bandwidth and CPU load.

Area 0 = BACKBONE - all others areas must connect to **Area 0,** or use **virtual links to it.**

Adjacent router's interfaces have to be in the **same area** to form a neighbor relationship and share updates!

Link-State protocol, scales efficiently to support larger networks

Loopback interfaces = ALWAYS UP!

== Router ID = ALWAYS THE SAME! Can be every X.X.X.X ID e.g. 200.200.200.200 (it's also high)

Process ID can be different on every router!

Link State Advertisement - update, explain something to someone

Support multiple areas! == group of routers

Authentication!

Shortest Path First Algorithm

Complex **cost** calculation! With many features!

Databases:

- Adjacency - Neighbors - say hello to them, share info = establish connection to the neighbors and keep it up
- Link-state - Topology table -
- Forwarding - SPF runs against the Link-State table and populates this one and then the routing table

Packets:

  - Hello, 10 secs, 30 on Frame Relay @ 224.0.0.5 < built in Dead Timer
- Database description - Link-State DB Check
- Link-State Request - request more info on an entry
- Link-State Update - reply to a request/update after change
- Link-State Ack. - acknowledges an update "thx I got it"

Learns first about the:

1. **Directly** connected ?
2. Hello Packets to adjacencies
3. Build Link-State Packets
4. Flood LSP Neighbors
5. Collects LSP's and builds topology map
6. Run SPF and populate the routing table

Designated Router and Backup DR:

Elect (**on boot) a DR**, everyone sends their updates to the **DR,** and then the **DR** updates everyone else with one BIG update (avoid traffic)

ID looks like an IP address. Highest wins.

Router 0# default information originate //shares the static routes (0.0.0.0!!!)

Passive Interfaces: Disable multicast updates, unicast works. The interface will **neither SENT** nor **RECEIVE** updates or hello packets (>> no NEIGHBOR relations!!)

#network 10.0.0.0 0.0.0.3 area 0 //area 0 = 0 backbone

OSPF Multi-Area

SPF - reduces traffic LSA - runs only on change

Routing table is smaller with multiple areas

- Backbone - Area 0 - High speed traffic
- Regular - end users - Area 1+

#Router ospf 1 //one OSPF process, different AREAS

#net xxxx area 0

#net xxxx area 1

#network xxxx 0.0.255.255 area 1

Wildcard Mask: the last two octets can match any number! = **RANGE OF HOSTS = FASTER to read**

OSPF Neighbor States:

(Darling) Down - no hello packets heard from neighbor

(I) Init - hello packet has been heard, but no 2-way communication

(Taste) Two-Way - 2-way communication established. **DR and BDR are elected. Hello Packets** carry a lists of the sender's known neighbors: | HELLO | R3 | R4 | << PACKET. The routers know see each other in the other's Hello packet!

(Some) ExStart - routers are preparing to exchange link-state info.

(Extremely) Exchange - exchanging of **Database Descriptors (DBDs)** - description of the router's **Topology Database.**Routers examine the neighbor's DBD to determine if it has info to share.

(Loud) Loading - routers are finally exchanging **LSAs**, containing information about all links connected. = Routers exchanging their topology tables

(Farts) Full - the routers are fully synchronized. The topology tables in the **AREA** are identical. **Full/(B)DR, /DROther** - **nor DR or BDR**

OSPF Network Types:

Broadcast Multi-Access - a topology where broadcast occurs.

Ethernet…. **OSPF** will elect **DRs and BDRs.** Traffic **TO DRs and BDRs: 224.0.0.6; traffic FROM DRs and BDRs: 224.0.0.5**

Neighbors must **not** be manually specified.

Point-to-Point - p2p - **NO DRs and BDRs** - all traffic is multicast 224.0.0.5 - Neighbors must **not** be manually specified.

Point-to-Multipoint - one interface can connect to multiple destinations - each connection is treated as **P2P. P2MP- Frame Relay**

Neighbors must **not** be manually specified.

Non-broadcast Multi-access Network (NBMA) - one interface <> multiple destination, NO broadcast whatsoever!

Frame Relay - **OSPF WILL elect DRs and BDRs** - Neighbors **MANUALLY** defined, thus all OSPF traffic is unicast, instead of multicast.

Internal Routers - all interfaces belong to the same **Area**

Backbone Routers - contains **at least** one interface in **Area 0**

Area Border Router (ABR) - belong to multiple areas, thus containing **multiple Topology DBs** for each.

Autonomous System Border Router (ASBR) - connects a separate **AS** or provides access to external networks (**Gateway?)**

> **Type 2 (E2) -** includes only the external cost **(default**); **Type 1(E1)** - includes both external and internal cost (before and after the **ASBR**) = total metric. Type 1 > Type 2

Link-State Advertisements:

Link = router interface. From them and their states the **Topology DB** is created.

Router LSA (Type 1) - contains a list if all local links, the status and the cost. Generated by **ALL** routers and **flooded** to **ALL** within the **area.**

Network LSA (Type 2) - generated by **DR** - contains a list of ALL routers attached to the **DR.**

Network Summary (Type 3) - generated by all **Area Border Routers.** Contains a list of all destination networks within an area to allow **inter-area** communication.

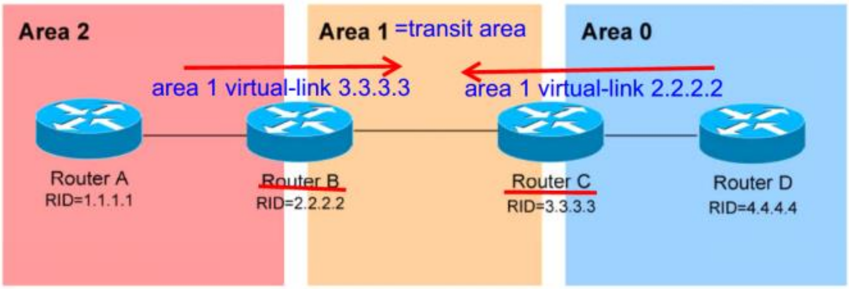
ASBR Summary (Type 4) - generated by **ABRs** and contains a route to **ANY ASBR** in the OSPF system (like a gateway)

External LSA (Type 5) - generated by **ASBR** and contains routes to destination networks **OUTSIDE** the **local AS.** Flooded to all areas of the OSPF system.

Virtual Links:

Area 2 must directly connect to **Area 0**, but doesn't have direct connection, so we use **Area 1** as **transit area.**

The two **Area Border Routers (ABRs)** 2 -> 1, and 1 -> 0 have to be configured = **TRANSIT AREA'S ABRS!!!**



It is also possible to have two **Area 0** (discontiguous) connected with a **virtual link.**

Area 0 <> Area 1 <> Area 0

RouterB# router-id 2.2.2.2 **RouterC# router-id 3.3.3.3**

RouterB# area 1 virtual-link 3.3.3.3 **RouterC# area 1 virtual-link 2.2.2.2**

==

13. EGP (Exterior Gateway Protocol) and BGP (Border Gateway Protocol).

==

14. Integrated IS-IS (Intermediate System to Intermediate System).

==

15. Advanced IP Routing Issues: Network Address Translation (NAT). Introduction to IP Multicast Routing, Protocol Independent Multicast (PIM), Large-Scale IP Multicast Routing.

A **public** address can be routed on the Internet. Thus, hosts that must be Internet-accessible must be configured with (or *reachable* by) public addresses. Allocation of public addresses is governed by the Internet Assigned Numbers Authority (IANA).

A **private** address is intended for internal use within a home or organization, and can be freely used by anyone. However, private addresses can *never be routed* on the Internet. In fact, Internet routers are configured to immediately drop traffic with private addresses.

NAT (Network Address Translation) - it is possible to translate between private and public addresses. It allows (one or more!) private address to be **stamped** with a public address, allowing it to communicate across the Internet.
NAT also hides the specific inside addresses and the structure of the network.
Also supports public-to-public and private-to-private translations.

==

==

MISCELANOUS:

Route Source	Administrative Distance
Connected	0
Static	1
EIGRP summary route	5
External BGP	20
Internal EIGRP	90
IGRP	100
OSPF	110
IS-IS	115
RIP	120
External EIGRP	170
Internal BGP	200

#show ip interface brief

A router connects multiple networks, which means that it has multiple interfaces that each belong to a different IP network. When a router receives an IP packet on one interface, it determines which interface to use to forward the packet to the destination. The interface that the router uses to forward the packet may be the final destination, or it may be a network connected to another router that is used to reach the destination network.

LANs are commonly Ethernet networks that contain devices, such as PCs, printers, and servers. WANs are used to connect networks over a large geographical area. For example, a WAN connection is commonly used to connect a LAN to the Internet service provider (ISP) network.

The primary functions of a router are to:

- Determine the best path to send packets -routing table
- Forward packets toward their destination - destination ip address > table > interface

- Process switching solves a problem by doing math long hand, even if it is the identical problem.
- Fast switching solves a problem by doing math long hand one time and remembering the answer for subsequent identical problems .
- Cisco Express Forwarding solves every possible problem ahead of time in a spreadsheet.

SERIAL = P2P = NO SOURCE REQUIRED !!
SERIAL DESTINATION = BROADCAST, because no MAC addresses

The following lists some dynamic protocols and the metrics they use:

- **Routing Information Protocol (RIP)** - Hop count
- **Open Shortest Path First (OSPF)** - Cisco's cost based on cumulative bandwidth from source to destination
- **Enhanced Interior Gateway Routing Protocol (EIGRP)** - Bandwidth, delay (,load, reliability)

When a router has two or more paths to a destination with equal cost metrics, then the router forwards the packets using both paths equally. This is called equal cost load balancing

Network discovery is the ability of a routing protocol to share information about the networks that it knows about with other routers that are also using the same routing protocol. Routers have converged after they have finished exchanging and updating their routing tables, they also determine a new best path if the initial path becomes unusable (or if the topology changes) without involving the network administrator.
D = EIGRP; * = possible default route;
EX = external, forwarded by EIGRP
Routers make their primary forwarding decision at Layer 3, the Network layer. However, router interfaces participate in Layers 1, 2, and 3. Layer 3 IP packets are encapsulated into a Layer 2 data link frame and encoded into bits at Layer 1. Router interfaces participate in Layer 2 processes associated with their encapsulation. For example, an Ethernet interface on a router participates in the ARP process like other hosts on that LAN.

CHAPTER 6 STATIC ROUTING

Static routes are very common and do not require the same amount of processing and **overhead** as dynamic routing protocols.
Static routes are not advertised over the network >> SECURITY!

Static route **AD = 1**, always preferred over dynamically learned.
Static route with EXIT INTERFACE **AD = 0**
Static routes remain in the routing table as long as it's INTERFACE stays UP > # ip route IP MASK NEXT-HOP ***permanent***
Static routes can be used to discard traffic to a **virtual null interface** > # ip route 10.0.0.0 255.0.0.0 **null0**
To reduce the number of routing table entries, multiple static routes can be summarized into a single static route if:
The destination networks are contiguous and can be summarized into a single network address.
The multiple static routes all use the same exit interface or next-hop IP address.
Another type of static route is a floating static route. Floating static routes are static routes that are used to provide a backup path to a primary static or dynamic route, in the event of a link failure. The floating static route is only used when the primary route is not available.
To accomplish this, the floating static route is configured with a higher administrative distance than the primary route. Recall that the administrative distance represents the trustworthiness of a route.
Global config:
Router(config)# **ip route***network-address subnet-mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [*distance*] [**name** *name*] [**permanent**] [**tag** *tag*]
The subnet mask can be modified to summarize a group of networks.
Static routes have : Next hop addr. or exit interface
Resolve: 192.168.2.0/24 via 172.16.2.2
Resolve: 172.16.2.2 from the routing table (Serial0/0/0) (RECURSIVE LOOKUP)
Note: For point-to-point interfaces, you can use static routes that point to the exit interface or to the next-hop address. For multipoint/broadcast interfaces, it is more suitable to use static routes that point to a next-hop address.
Fully specified static route = Next Hop + Exit Interface!!!
Default static routes are commonly used when connecting:
An edge router to a service provider network
A stub router (a router with only one upstream neighbor router)
S* 0.0.0.0/0 via …
S = static; * = candidate; /0 none of the bits must match
The **ipv6 unicast-routing** global configuration command must be configured to enable the router to forward IPv6 packets.
CLASSFUL ADDRESSING
As shown in Figure 1, class A networks used the first octet to identify the network portion of the address. This is translated to a 255.0.0.0 classful subnet mask. Because only 7 bits were left in the first octet (remember, the first bit is always 0), this made 2 to the 7th power, or 128 networks. The actual number is 126 networks, because there are two reserved class A addresses (i.e., 0.0.0.0/8 and 127.0.0.0/8). With 24 bits in the host portion, each class A address had the potential for over 16 million individual host addresses.
As shown in Figure 2, class B networks used the first two octets to identify the network portion of the network address. With the first two bits already established as 1 and 0, 14 bits remained in the first two octets for assigning networks, which resulted in 16,384 class B network addresses. Because each class B network address contained 16 bits in the host portion, it controlled 65,534 addresses. (Recall that two addresses were reserved for the network and broadcast addresses.)
As shown in Figure 3, class C networks used the first three octets to identify the network portion of the network address. With the first three bits established as 1 and 1 and 0, 21 bits remained for assigning networks for over 2 million class C networks. But, each class C network only had 8 bits in the host portion, or 254 possible host addresses.
Summary and Floating Static routes
Another type of static route is a floating static route. Floating static routes are static routes that are used to provide a backup path to a primary static or dynamic route, in the event of a link failure. The floating static route is only used when the primary route is not available.
To accomplish this, the floating static route is configured with a higher administrative distance than the primary route. Recall that the administrative distance represents the trustworthiness of a route.