# Network Device and Configuration

**Chapter Three**

Routing Protocols Configuration

Feb 2024

# Introduction to Routing

- A **routing protocol** is a set of rules used by routers to determine most appropriate paths into which they should forward packets towards their intended destinations.

# Introduction to Routing

- **Routing is a process accomplished by router and defined as follows:**
  - The process during which data packets are forwarded from one machine or device (node) to another on a network until they reach their destinations.
  - Selecting the minimum cost, distance, and/or time path from several alternatives for a good or message to reach its destination.
  - IP routing uses IP addresses to forward IP packets from their sources to their destinations.

# Introduction to Routing

- The term **routing** encapsulates two tasks: deciding the paths for data transferred, and sending the packets on these paths.
- Routing is a function carried out at the $3^{rd}$ layer of the OSI reference model.
- A routing algorithm decides the output line to transfer the incoming packets:
  - Algorithms are based on the routing protocol that uses metrics; bandwidth, delay, and reliability-to assess whether a particular path is the optimal path available for transfer of the data packets.

# Routing Table

- The routing table contains information about various routes between devices in order to present the most efficient paths for data packets.
- This table is usually stored inside the **Random Access Memory** of forwarding devices, such as routers.
- The routing table is commonly referred to as a resource for finding the next hop, or subsequent route for a data packet.

# Routing Table

- Routing tables contains the following entries:
  - **Destination:** This is the IP address of the packet's final destination.
  - **Subnet mask:** Also known as the netmask, this is a 32-bit network address that identifies whether a host belongs to the local or remote network.
  - **Gateway:** This is the next hop, or the neighboring device's IP address to which the packet is forwarded.
  - **Interface:** the port at which a router connects to a given network.
  - **Metric:** this entry assigns a value to each available route to a specific network. The value ensures that the router can choose the most effective path. In some cases, the metric is the number of routers that a data packet must cross before it gets to the destination address.
  - **Routes:** This includes directly attached subnets, indirect subnets that aren't attached to the device but can be accessed through one or more hops, and default routes to use for certain types of traffic or when information is lacking.

# Routing Table

```
===========================================================================

IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0       10.0.0.1        10.0.0.32     35
         10.0.0.0    255.255.255.0       On-link         10.0.0.32    291
        10.0.0.32  255.255.255.255       On-link         10.0.0.32    291
       10.0.0.255  255.255.255.255       On-link         10.0.0.32    291
        127.0.0.0        255.0.0.0       On-link         127.0.0.1    331
        127.0.0.1  255.255.255.255       On-link         127.0.0.1    331
  127.255.255.255  255.255.255.255       On-link         127.0.0.1    331
        224.0.0.0        240.0.0.0       On-link         127.0.0.1    331
        224.0.0.0        240.0.0.0       On-link         10.0.0.32    291
  255.255.255.255  255.255.255.255       On-link         127.0.0.1    331
  255.255.255.255  255.255.255.255       On-link         10.0.0.32    291
===========================================================================
Persistent Routes:
  None

IPv6 Route Table
===========================================================================
Active Routes:
 If Metric Network Destination      Gateway
  1    331 ::1/128                  On-link
 21    291 [redacted]               On-link
 21    291 [redacted]
                                    On-link
  1    331 [redacted]               On-link
 21    291 [redacted]               On-link
```

# Routing Table

## challenge of designing a routing table

- Recording information on many devices with a fixed memory or storage space.
- Incorrect definition of topology of a network.
- Other routing problems, such as **black holes**.

# Types of Routing

- The two classifications of routing are **static routing** and **dynamic routing**.
- These classifications are based on the way in which routing tables are created and updated every time they are used.
- Routing in which the data in routing table is stored and updated manually called **static routing**.
- Routing in which the information in routing table is changed dynamically, by the router itself, are referred to as **dynamic routing**.

# Static Routing

- It is simply the process of manually entering routes into a device's routing table via a configuration file that is loaded when the routing device starts up.
- Routers will not share static routes with each other, thus **reducing CPU/RAM overhead** and **saving bandwidth**.
- However, static routing is **not fault-tolerant**, as any change to the routing infrastructure (such as a link going down, or a new network added) requires manual intervention.
- Static routes have an **Administrative Distance (AD)** of 1, and thus 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**.
- Use static routing when you have very few devices to configure and when you know the routes will probably never change.

# Advantages and Disadvantages of Static Routing

## Advantages of Static Routing

- Minimal CPU/Memory overhead
- No bandwidth overhead (updates are not shared between routers)
- Granular control on how traffic is routed – adds security

## Disadvantages of Static Routing

- Infrastructure changes must be manually adjusted
- No dynamic fault tolerance if a link goes down -the administrator is responsible
- Impractical on large network

# Static Routing

## Command syntax

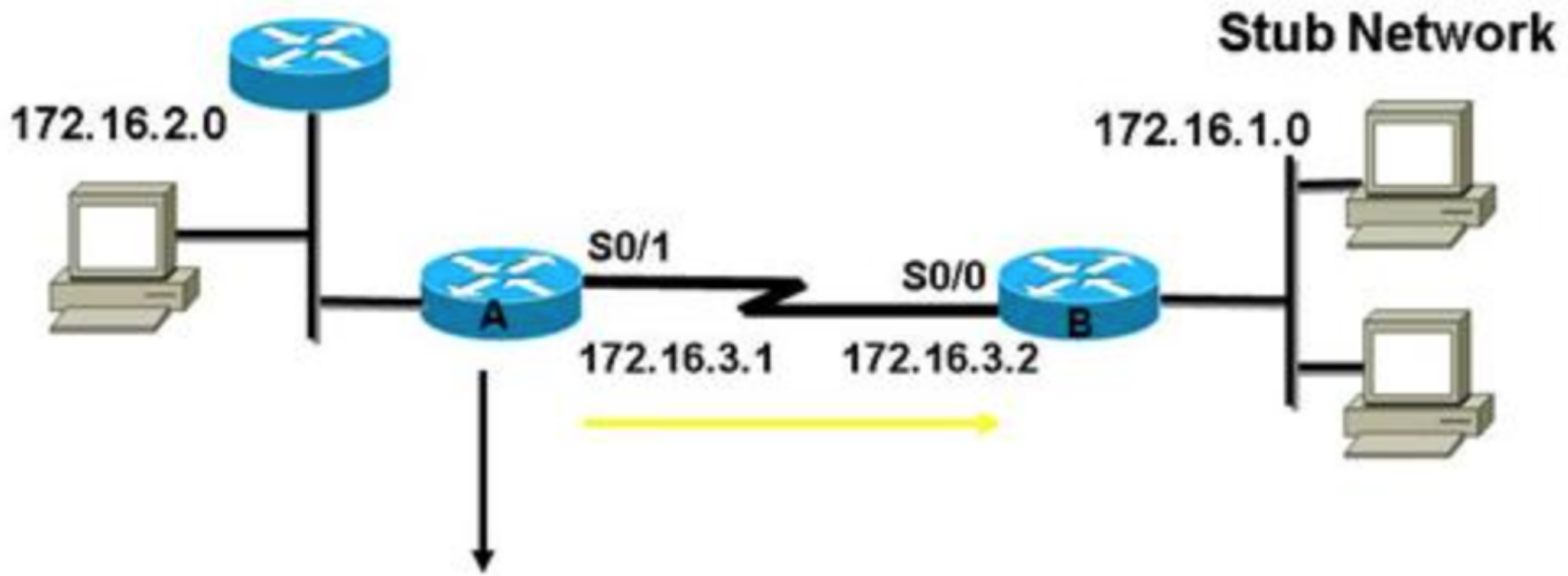
*ip route [destination_network] [mask] [next-hop_address or exit interface] [administrative_distance] [permanent]*

- **ip route:** The command used to create the static route.
- **Destination_network:** The network you're placing in the routing table.
- **Mask:** The subnet mask being used on the network.
- **Next-hop_address:** The address of the next-hop router that will receive the packet and forward it to the remote network.
- **Exit interface:** Used in place of the next-hop address if you want, and shows up as directly connected route.

# Static Routing

- **Administrative_distance:** By default, static routes have an administrative distance of 1 (or even 0 if you use an exit interface instead of a next-hop address).
- **Permanent:** If the interface is shut down or the router can't communicate to the next-hop router, the route will automatically be discarded from the routing table.

# Static Routing




ip route 172.16.1.0 255.255.255.0 172.16.3.2

or

ip route 172.16.1.0 255.255.255.0 s0/1

# Static Routing

## Floating static route:

- A floating static route is a type of static route used as a backup or failover option in a network.
- It is given a higher administrative distance than the dynamic routes typically used in the network, which means that the router will prefer to use the dynamic routes but will fall back to using the floating static route if the dynamic routes are unavailable.

# Static Routing

## Default Route

- A default route defines where packets will be sent if no specific route for the destination network is listed in the routing table.
- If no default route is set, the router will discard all packets with destination addresses not found in its routing table.
- **Syntax:**
  - *ip route 0.0.0.0 0.0.0.0 [next-hop_address or exit interface]*

# Dynamic Routing

- It uses software and routing algorithms to determine optimal network data transfer and communication paths.
- Routers do share dynamic routing information with each other, which **increases CPU, RAM**, and **bandwidth** usage.
- However, routing protocols are capable of dynamically choosing a different (or better) path when there is a change to the routing infrastructure.
- A router using dynamic routing will 'learn' the routes to all networks that are directly connected to the device.
- Next, the router will learn routes from other routers that run the same routing protocol (**RIP, RIP2, EIGRP, OSPF, IS-IS, BGP** etc.).

# Advantages and Disadvantages of Dynamic Routing

## Advantages of Dynamic Routing

- Simpler to configure on larger networks
- Will dynamically choose a different (or better) route if a link goes down
- Ability to load balance between multiple links

## Disadvantages of Dynamic Routing

- Updates are shared between routers, thus consuming bandwidth
- Routing protocols put additional load on router CPU/RAM
- The choice of the best route is in the hands of the routing protocol, and not the network Administrator

# Categories of Dynamic Routing Protocols

- **Distance Vector Routing Protocols (DVRP)**
  - DVRP find the best path to a remote network by judging **distance**.
  - The route with the least number of hops to the network is determined to be the best route.
  - Both **RIP** and **IGRP** are examples of distance-vector routing protocols.
  - They send the entire routing table to directly connected neighbors.

# Categories of Dynamic Routing Protocols

## Key characteristics of DVRP

- Periodic updates of the full routing table are sent to routing neighbors.
- Distance-vector protocols suffer from slow convergence, and are highly susceptible to loops.
- Some form of distance is used to calculate a route's metric.
- The **Bellman-Ford algorithm** is used to determine the shortest path.

# Categories of Dynamic Routing Protocols

- **Distance Vector Routing Protocols (DVRP)**
  - A DVRP begins by advertising directly-connected networks to its neighbors. These updates are sent regularly (RIP – every 30 seconds; IGRP – every 90 seconds).
  - Neighbors will add the routes from these updates to their own routing tables.
  - Each neighbor trusts this information completely, and will forward their full routing table (connected and learned routes) to every other neighbor.
  - Thus, routers fully (and blindly) rely on neighbors for route information, a concept known as **routing by rumor**.
  - Distance-vector protocols utilize some form of distance to calculate a route's metric.
    - **RIP** uses hop count as its distance metric, and **IGRP** uses a composite of bandwidth and delay.

# Categories of Dynamic Routing Protocols

- **Link State Routing Protocol (LSRP)**
  - Link-state protocols are also called **shortest-path-first protocols**, each router create three separate tables.
  - Link state routers know more about the internetwork than any distance-vector routing protocol.
  - Link-state protocols send updates containing the state of their own links to all other routers on the network.
  - Examples of link state routing protocols are **IS-IS** and **OSPF**.
  - Link-state routing protocols were developed to alleviate the **convergence** and **loop issues** of distance-vector protocols.

# Categories of Dynamic Routing Protocols

**Link-state protocols maintain three separate tables:**

## Neighbor table

- contains a list of all neighbors, and the interface each neighbor is connected.
- Neighbors are formed by sending Hello packets.

## Topology table

- also 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 (also known as the routing table)

# Categories of Dynamic Routing Protocols

- **Link State Routing Protocol (LSRP)**
  - Link-state protocols do not **route by rumor**.
  - Instead, routers send updates advertising the state of their links (a link is a directly-connected network).
  - If the state of a link changes, such as a router interface failing, an advertisement containing only this link-state change will be sent to all routers within that area.
  - Each router will adjust its topology table accordingly, and will calculate a new best route if required.

# Categories of Dynamic Routing Protocols

- **Link State Routing Protocol (LSRP)**
  - By maintaining a consistent topology table among all routers within an area, link-state protocols can **converge very quickly** and are **immune to routing loops**.
  - Additionally, because updates are sent only during a link-state change, and contain only the change (and not the full table), link-state protocols are **less bandwidth intensive** than distance-vector protocols.
  - However, the three link-state tables utilize more RAM and CPU on the router itself.
  - Link-state protocols utilize some form of cost, usually based on bandwidth, to calculate a route's metric.
  - The **Dijkstra formula** is used to determine the shortest path.

# Categories of Dynamic Routing Protocols

Do not confuse **routing protocols** with **routed protocols**:

## Routed protocols

- A routed protocol is a Layer 3 protocol that applies logical addresses to devices and routes data between networks (such as IP).

## Routing protocol

- A routing protocol dynamically builds the network, topology, and next hop information in routing tables (such as RIP, EIGRP, etc.)

# Basic Concepts on AD, Metrics and Wildcard Mask

- **Administrative Distance (AD)**
  - An AD is the value used by routers to choose the best path when there are two or more routes to the same destination from two different routing protocols.
  - Static routes have a default administrative distance of 1.
  - A router prefers a static route to a dynamic route because the router considers a route with a low number to be the shortest.
  - If you want a dynamic route to override a static route, you can specify an administrative distance for the static route.
  - It is used to rate the trustworthiness of routing information received on a router from a neighbor router.
  - Is an integer from 0 to 255, where **0** is the **most trusted** and **255** means **no traffic will be passed via this route**.

# Basic Concepts on AD, Metrics and Wildcard Mask

- **Administrative Distance (AD)**
  - If a router receives two updates listing the same remote network, the first thing the router checks is the AD.
  - If one of the advertised routes has a lower AD than the other, then the route with the lowest AD will be placed in the routing table.
  - If both advertised routes to the same network have the same AD, then routing protocol **metrics** (such as **hop count** or **bandwidth of the lines**) will be used to find the best path to the remote network.
  - The advertised route with the lowest metric will be placed in the routing table.
  - But if both advertised routes have the same **AD** as well as the same **metrics**, then the routing protocol will load-balance to the remote network (which means that it sends packets down each link).

# Basic Concepts on AD, Metrics and Wildcard Mask

| IP route Type | Administrative Distance (Default) |
|---|---|
| Connected interface | 0 |
| Static route | 1 |
| External BGP | 20 |
| EIGRP | 90 |
| OSPF | 110 |
| RIP | 120 |
| External EIGRP | 170 |
| Internal BGP | 200 |
| Unknown | 255 |

# Basic Concepts on AD, Metrics and Wildcard Mask

- **Metric**
  - There are cases when a routing protocol learns of more than one route to the same destination.
  - To select the best path, the routing protocol must be able to evaluate and differentiate between the available paths. For this purpose a metric is used.
  - A **metric** is a value used by routing protocols to assign costs to reach remote networks.
  - The metric is used to determine which path is most preferable when there are multiple paths to the same remote network.

# Basic Concepts on AD, Metrics and Wildcard Mask

- **Metric**
  - Each routing protocol uses its own metric.
    - For example, **RIP** uses hop count, **EIGRP** uses a combination of bandwidth and delay, and Cisco's implementation of **OSPF** uses bandwidth.
    - Hop count is the easiest metric to envision. The hop count refers to the number of routers a packet must cross to reach the destination network.

# Basic Concepts on AD, Metrics and Wildcard Mask

- **Wildcard Mask**
  - A wildcard mask is a mask of bits that indicates which parts of an IP address can assume any value.
  - In the Cisco IOS, they are used in several places, for example:
    - To indicate the size of a network or subnet for some routing protocols, such as OSPF.
    - To indicate what IP addresses should be permitted or denied in access control lists (ACLs).

# Basic Concepts on AD, Metrics and Wildcard Mask

- **Wildcard Mask**
  - A wildcard mask can be thought of as a subnet mask, with ones and zeros inverted;
    - for example, a wildcard mask of 0.0.0.255 corresponds to a subnet mask of 255.255.255.0.
  - To calculate wildcard mask Simply subtract your mask from 255.255.255.255 to get your wildcard mask.
  - **Example:-**
  - The wildcard mask of /26 is:
    - 255.255.255.255 - 255.255.255.192 = 0.0.0.63
  - The wildcard mask of /19 is:
    - 255.255.255.255 - 255.255.224.0 = 0.0.31.255
  - The wildcard mask of /12 is:
    - 255.255.255.255 - 255.240.0.0 = 0.15.255.255

# Routing Information Protocol (RIP)

- **RIP** is a standards-based, distance-vector, interior gateway protocol (IGP) used by routers **to exchange routing information**.
- RIP only uses **hop count to determine the best path** between two locations.
- **Hop count** is the number of routers the packet must go through till it reaches the destination network.
- The maximum allowable number of hops a packet can traverse in an IP network implementing RIP is 15 hops.
  - It has a maximum allowable hop count of 15 by default, meaning that 16 is deemed unreachable.

# Routing Information Protocol (RIP)

- RIP works well in small networks, but it's inefficient on large networks with slow WAN links or on networks with a large number of routers installed.
- Each router broadcasts its entire RIP table to its neighboring routers every **30 seconds**.
- If RIP finds more than one link to the same remote network with the same hop count, it will automatically perform a **round-robin load balancing**.
- Problem with this type of routing metric arises w/n two links to a remote network are **different bandwidths but same hop count**.

# The differences between RIPv1 and RIPv2

## RIPv1

- It can supports class full network only i.e. all devices in the network must use the same subnet mask.
- RIPV1 does not send a subnet mask to the routing table.
- It does not support for VLSM and discontinuous networks.
- The routing update address used for Broadcast is 255.255.255.255.
- RIPV1 does not provide trigger updates.
- It is less secure.

# The differences between RIPv1 and RIPv2

## RIPv2

- It can support class full and classless networks.
- RIPv2 sends subnet mask to the routing table.
- It supports for VLSM and discontinuous networks.
- The routing update address used for Multicast is 224.0.0.9.
- RIPv2 provides trigger updates.
- It is more secure because it supports authentication.

# RIP Timers

**RIP uses four different kinds of timers to regulate its performance:**

## Route update timer

- Sets the interval (typically 30 seconds) between periodic routing updates in which the router sends a complete copy of its routing table out to all neighbors.

## Hold-down timer

- Routes will enter into the holddown state when an update packet is received that indicated the route is unreachable.
- The default is 180 seconds.

# RIP Timers

## Route invalid timer

- It determines the length of time that must elapse (180 seconds) before a router determines that a route has become invalid.
- It will come to this conclusion if it hasn't heard any updates about a particular route for that period.

## Route flush timer

- This sets the time between a route becoming invalid and its removal from the routing table (240 seconds).

# RIP configuration

## RIPv1 Syntax

- *router rip (on Global configuration mode)*
- *network Network_Address*

## RIPv2 Syntax

- *router rip (on Global configuration mode)*
- *version 2*
- *network Network_Address*

# Interior Gateway Routing Protocol (IGRP)

- Is a dynamic class routing protocol used by autonomous system (AS) routers running on TCP/IP hosts.
- An AS is a collection of networks under a common administrative domain, which basically means that all routers sharing the same routing table information are in the same AS.
- IGRP overcomes Routing Information Protocol (RIP) network limitations and supports multiple routing metrics, including delay, bandwidth, load and reliability.
- Routing updates are broadcast every 90 seconds (by default).

# Interior Gateway Routing Protocol (IGRP)

## Characteristics of IGRP

- Developed by Cisco
- Uses composite metrics
- Uses multipath routing
- Supports unequal-cost load balancing
- Supports hold-downs and split horizon
- Deprecated

# Enhanced Interior Gateway Routing Protocol (EIGRP)

- EIGRP is an advanced distance vector routing protocol based on the principles of the Interior Gateway Routing Protocol (IGRP).
- A protocol with faster converging abilities, route selection and calculation and ability to record information from neighboring devices.
- It uses bandwidth, delay, load and reliability to calculate the metric for its routing table (not hop count used by legacy protocols).
- For this reason, EIGRP always selects and calculates the most optimal route for efficiency.
- It uses a DUAL algorithm to avoid loops and send occasional hello packets to check the status of neighbor routers.

# Enhanced Interior Gateway Routing Protocol (EIGRP)

## Characteristics of EIGRP

- Advanced operational efficiency
- Capabilities of both link state and distance vector
- A classless routing protocol
- Faster converging because it pre-calculates routes and does not broadcast hold-down timer packets before converging
- Unique features including use of Reliable Transport Protocol (RTP-capable of transmitting both multicast and unicast), a diffusing update algorithm (DUAL), updates and updated information about neighbors

# EIGRP Configuration

## EIGRP Syntax

- *router eigrp AS (On global Configuration mode)*
- *network Network_Address | network Network _Address Wildcard mask*

- Where *AS* can be any number in the range from 1 to 65535 both inclusive.

# Open Shortest Path First (OSPF)

- OSPF is a link state routing protocol (LSRP) that uses the Shortest Path First (SPF) network communication algorithm (Dijkstra's algorithm) to calculate the shortest connection path between known devices.
- OSPF is an Interior Routing Protocol (IGP) that routes Internet Protocol (IP) packets within a single routing network domain only.
- OSPF finds the best network layout (topology) by calculating shortest device connection paths using the Shortest Path First (SPF) algorithm.

# Open Shortest Path First (OSPF)

## OSPF two layer hierarchy

- Backbone area (area 0)
- Off backbone area ( area 1 -65, 535)

# Open Shortest Path First (OSPF)

## Characteristics of OSPF

- AD value is 110
- Supports classless network
- Supports VLSM/CIDR and has unlimited hop counts
- Supports hierarchical network
- Route propagation using multicasting

# OSFP Configuration

## OSPF Syntax

- *router ospf Process_ID (On global Configuration mode)*
- *network Network_Address Wildcard_mask* ***area 0***

- Where *Process_ID* can be any number in the range from 1 to 65535 both inclusive.

# Routing Protocol Comparison

| Name | Class | Type | AD | Metric | Classful/ Less | Algorithm | Transport type |
|---|---|---|---|---|---|---|---|
| RIPv1 | Distance Vector | IGP | 120 | Hop count | Classful | Bellman-Fold | UDP/520 |
| RIPv2 | Distance Vector | IGP | 120 | Hop count | Classless | Bellman-Fold | UDP/520 |
| IGRP | Distance Vector | IGP | 100 | Composite (BW+DLY) | Classful | Dijkstra(SPF) | IP Protocol 9 |
| EIGRP | Advanced Distance Vector | IGP | 90 (internal) 170(external) | Composite (BW+DLY) | Classless | DUAL | EIGRP Protocol 88 |
| OSPF | Link State | IGP | 110 | Cost(BW) | Classless | Dijkstra(SPF) | OSPF Protocol 89 |

# ACL (Access Control List)

- Access lists are a set of rules, organized in a rule table.
- Each rule or line in an access-list provides a condition, either **permit** or **deny**.
- **ACLs can be used for two purposes on Cisco devices:**
  - To filter traffic
  - To identify traffic
- When using an access-list to filter traffic, a permit statement is used to **"allow"** traffic, while a deny statement is used to **"block"** traffic.
- Similarly, when using an access list to identify traffic, a permit statement is used to **"include"** traffic, while a deny statement states that the traffic should **"not" be included**.
- Access control lists enable you to permit or deny packets based on **source** and **destination IP address**, **IP protocol information**, or **TCP** or **UDP protocol information**.

# Types of Access Lists

- There are two categories of access lists: **numbered** and **named**.
- **Numbered**
  - Numbered ACLs use a numeric code to refer to each ACL.
- **Types of numbered ACLs:**

## Standard

- Permits or denies packets based on **source IP address**.
- Valid standard ACL IDs are **1 – 99** or a string.

## Extended

- Permits or denies packets based on **source** and **destination IP address** and also based on **IP protocol information**.
- Valid extended ACL IDs are a number from **100 – 199** or a string.

# Types of Access Lists

- **Named**
  - Named access lists provide a bit more flexibility.
  - Named ACLs use a descriptive word or phrase, such as "INBOUND" or "ALLOW_HTTP", to identify each ACL.
  - **Types of named access**
    - IP standard named access lists
    - IP extended named access lists

# Types of Access Lists

- **IP access-lists** use **wildcard masks** to determine two things:
  - Which part of an address must match exactly?
  - Which part of an address can match any number?
- Consider the following address and wildcard mask:
  - Address: 172.16.0.0
  - Wild Card Mask: 0.0.255.255
- The above would match any address that begins “172.16.”
- The last two octets could be anything.
- **How do I know this?**

# Types of Access Lists

- **Two Golden Rules of Access Lists:**
  1. If a bit is set to 0 in a wild-card mask, the corresponding bit in the address must be matched exactly.
  2. If a bit is set to 1 in a wild-card mask, the corresponding bit in the address can match any number. In other words, we “don’t care” what number it matches.

# Types of Access Lists

- To see this more clearly, we'll convert both the address and the wild card mask into binary:
  - **Address:** 10101100.00010000.00000000.00000000
  - **Wild Card Mask:** 00000000.00000000.11111111.11111111
- Any 0 bits in the wildcard mask, indicates that the corresponding bits in the address must be matched exactly.
  - Thus, looking at the above example, we must exactly match the following in the first two octets: 10101100.00010000 = 172.16
- Any 1 bits in the wildcard mask indicates that the corresponding bits can be anything.
  - Thus, the last two octets can be any number, and it will still match this access-list entry.

# Types of Access Lists

- If wanted to match a **specific address** with a wildcard mask (we'll use an example of 172.16.1.1), how would we do it?
  - **Address:** 172.16.1.1
  - **Wild Card Mask:** 0.0.0.0
- Written out in binary, that looks like:
  - **Address:** 10101100.00010000.00000001.00000001
  - **Wild Card Mask:** 00000000.00000000.00000000.00000000
- Remember what a wildcard mask is doing. A 0 indicates it must match exactly, a 1 indicates it can match anything. The above wildcard mask has all bits set to 0, which means we must match all four octets exactly.
- **There are two ways to match a host:**
  - Using a wildcard mask with all bits set to 0 – 172.16.1.1 0.0.0.0
  - Using the keyword "**host**" – host 172.16.1.1

# Types of Access Lists

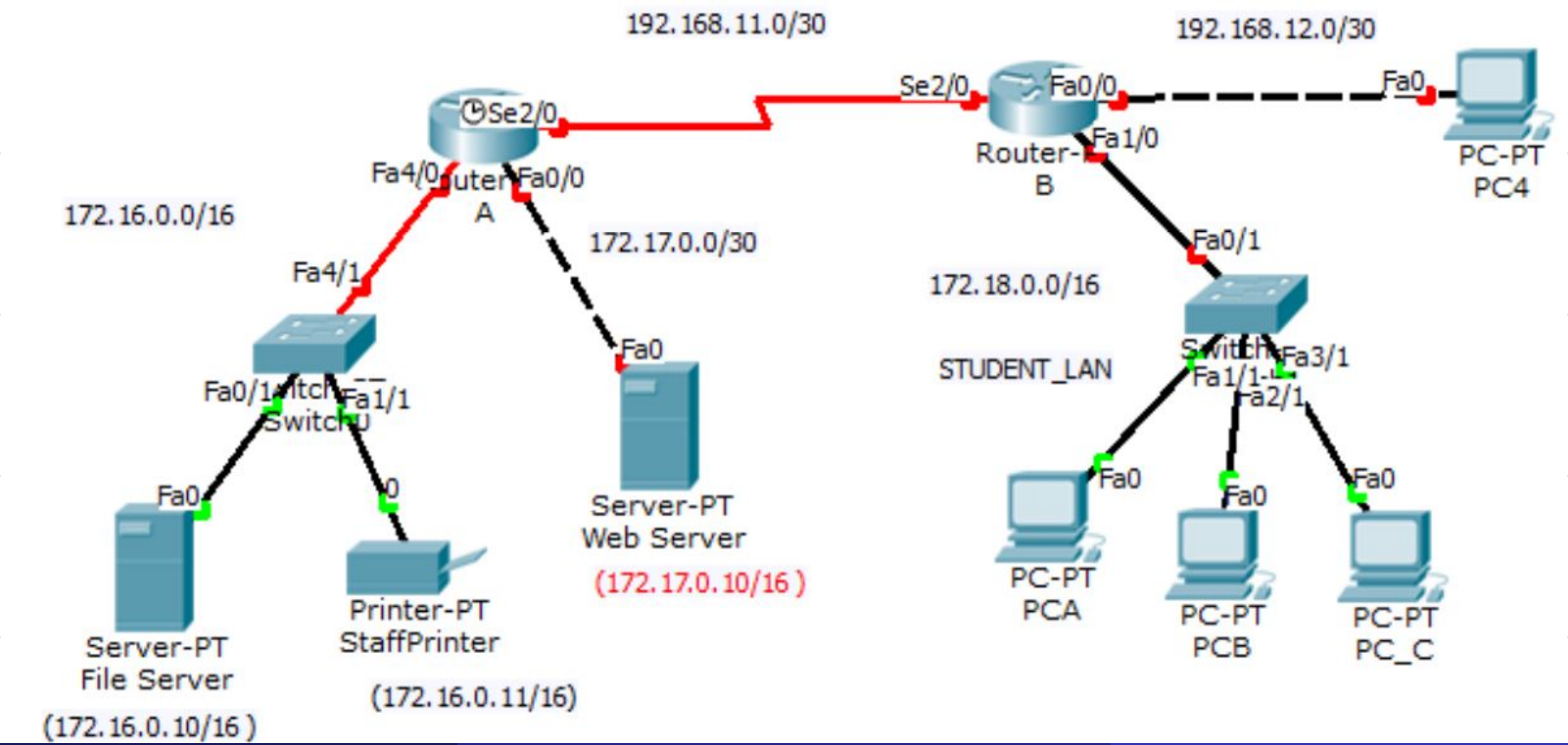
- **How would we match all addresses with a wildcard mask?**
  - **Address:** 0.0.0.0
  - **Wild Card Mask:** 255.255.255.255
- Written out in binary, that looks like:
  - **Address:** 00000000.00000000.00000000.00000000
  - **Wild Card Mask:** 11111111.11111111.11111111.11111111
- Notice that the above wildcard mask has all bits set to 1. Thus, each bit can match anything – resulting in the above address and wildcard mask matching all possible addresses.
- **There are two ways to match all addresses:**
  - Using a wildcard mask with all bits set to 1 – 0.0.0.0 255.255.255.255
  - Using the keyword **"any"** – any

# Standard IP Access List

## Syntax

*access-list [1-99] [permit | deny] [source address] [wildcard mask] [log]*

- **Standard IP access-lists** are based upon the **source host** or **network IP address**, and should be placed closest to the destination network.



# Standard IP Access List

- In order to block network 172.18.0.0 from accessing the 172.16.0.0 network, we would create the following access-list on Router A:
  - Router(config)# access-list 10 deny 172.18.0.0 0.0.255.255
  - Router(config)# access-list 10 permit any
- Notice the wildcard mask of 0.0.255.255 on the first line. This will match (deny) all hosts on the 172.18.x.x network.
- The second line uses a keyword of any, which will match (permit) any other address. Remember that you must have at least one permit statement in your access list.

# Standard IP Access List

- **To apply this access list, we would configure the following on Router A**
  - Router(config)# int fa4/0
  - Router(config-if)# ip access-group 10 out
    - To view all IP access lists configured on the router:
  - Router# show ip access-list
    - To view what interface an access-list is configured on:
  - Router# show ip interface
  - Router# show running-config

# Extended IP Access List

## Syntax

*access-list [100-199] [permit | deny] [protocol] [source address] [wildcard mask] [destination address] [wildcard mask] [operator [port]] [log]*

- **Extended IP access-lists** block based upon the **source IP address**, **destination IP address**, and **TCP** or **UDP port number**.
- Extended access-lists should be placed closest to the source network.

# Extended IP Access List

- Consider the following example: (use the above network topology)
- Assume there is a web server on the 172.17.x.x network with an IP address of 172.17.0.10.
- In order to block network 172.18.0.0 from accessing anything on the 172.17.0.0 network, EXCEPT for the HTTP port on the web server, we would create the following access-list on Router B:
  - Router(config)# access-list 101 permit tcp 172.18.0.0 0.0.255.255 host 172.17.0.10 eq 80
  - Router(config)# access-list 101 deny ip 172.18.0.0 0.0.255.255 172.17.0.0 0.0.0.3
  - Router(config)# access-list 101 permit ip any any
    - The first line allows the 172.18.x.x network access only to port 80 on the web server.
    - The second line blocks 172.18.x.x from accessing anything else on the 172.17.x.x network.
    - The third line allows 172.18.x.x access to anything else.

# Extended IP Access List

- We could have identified the web server in one of two ways:
- Router(config)# access-list 101 permit tcp 172.18.0.0 0.0.255.255 host 172.17.0.10 eq 80
- Router(config)# access-list 101 permit tcp 172.18.0.0 0.0.255.255 172.17.0.10 0.0.0.0 eq 80
- To apply this access list, we would configure the following on Router B:
- Router(config)# int fa1/0
- Router(config-if)# ip access-group 101 in

# Network Address Translation (NAT)

- It enables private IP networks that use unregistered IP addresses to connect to the Internet.
- **NAT** operates on a router, usually connecting two networks together, and translates the private (not globally unique) addresses in the internal network into legal (globally known) addresses, before packets are forwarded to another network.
- As part of this capability, NAT can be configured to advertise only one address for the entire network to the outside world.
- This provides additional **security** by effectively hiding the entire internal network behind that address.
- **NAT** offers the **dual functions of security** and **address conservation** and is typically implemented in remote-access environments.

# How NAT Works

- A workstation inside a network makes a request to a computer on the Internet.
- Routers within the network recognize that the request is not for a resource inside the network, so they send the request to the firewall.
- The firewall sees the request from the computer with the internal IP.
- It then makes the same request to the Internet using its own public address, and returns the response from the Internet resource to the computer inside the private network.
- From the perspective of the resource on the Internet, it is sending information to the address of the firewall.
- From the perspective of the workstation, it appears that communication is directly with the site on the Internet.

# How NAT Works

- NAT is only a temporarily solution to the address shortage problem.
- IPv4 will eventually be replaced with IPv6, which supports a vast address space.

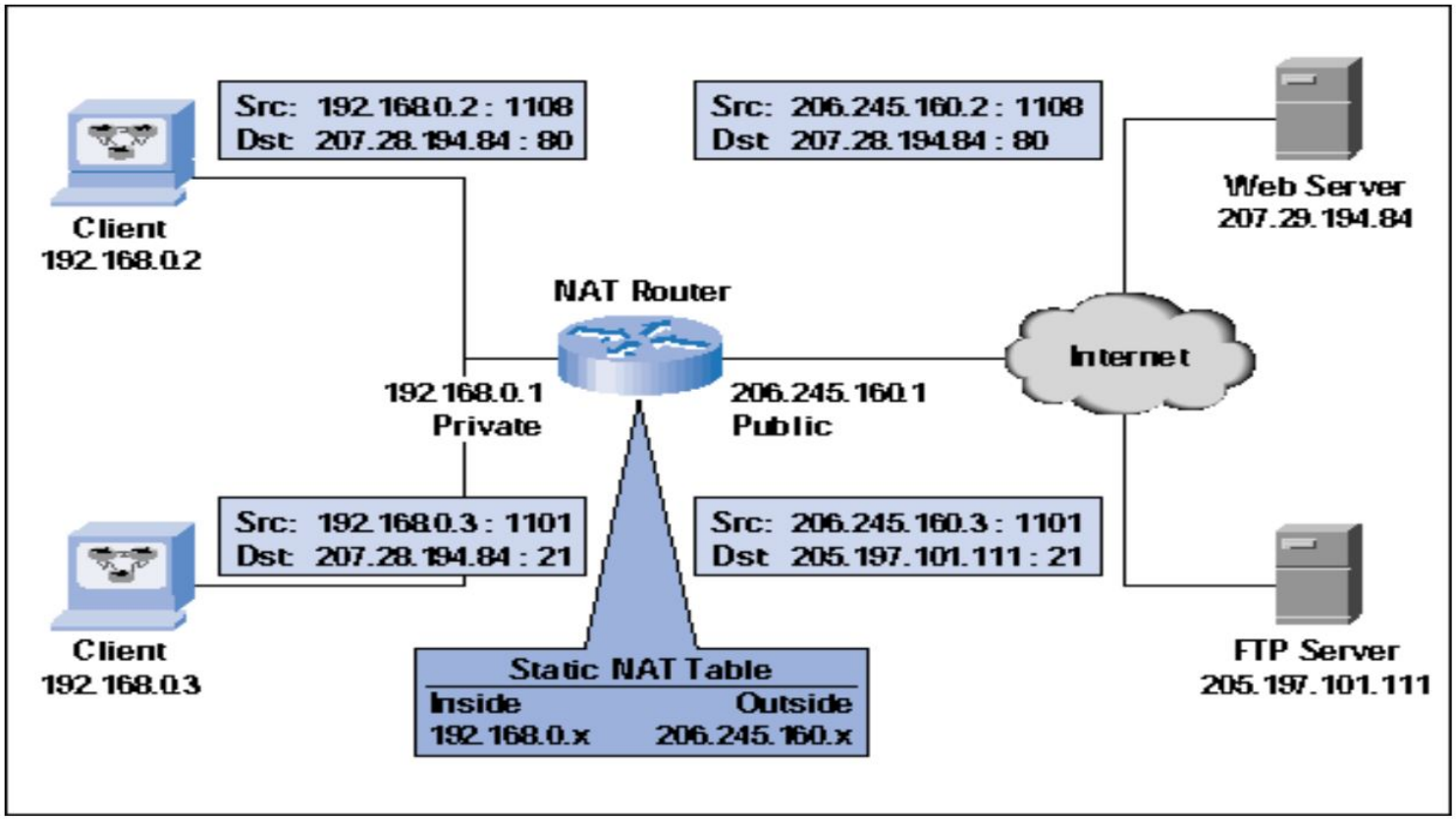
## Situation to use NAT

- ISP did not provide sufficient public IP address
- Company is going to merge in a company w/c use same address space
- Want to hide internal IP address space from outside
- Want to assign same IP address to multiple machines

# Types of NAT

## Static NAT

- Provides **one-to-one mapping** between local and global addresses, consequently, every computer on the network must be allocated a single dedicated routable IP address.



# Types of NAT

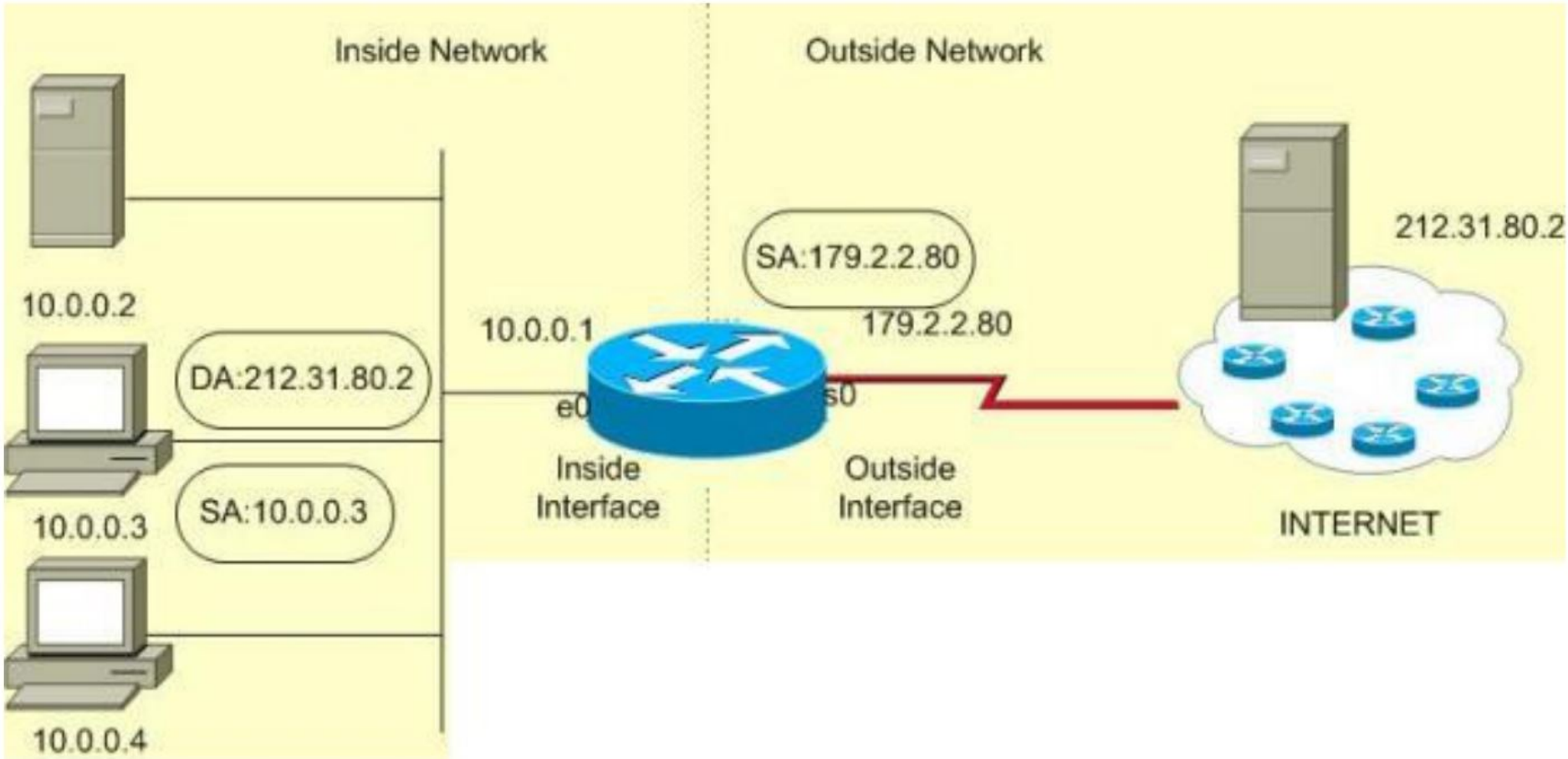
- To configure static inside source address translation for the example shown in the Figure above, the following need to be performed on the router:
- **Specify the inside interface:**
  - Router(config)#interface fast ethernet0/0 (private side interface)
  - Router(config)#ip nat inside
- **Specify the outside interface:**
  - Router(config)#interface fast ethernet0/1 (Public side interface)
  - Router(config)#ip nat outside
- **Enter static translation entry:**
  - Router(config)# ip nat inside source static 192.168.0.1 206.245.160.1

# Types of NAT

## Dynamic NAT

- A pool of routable IP addresses is configured on the router and dynamically the router assigns addresses from this pool to every machine that requires sending traffic to the “outside world”.
- Like Static NAT, this creates a one-to-one mapping between internal and external IP addresses; however, these mappings are not permanent.

# Types of NAT




- To configure dynamic inside source address translation for the example shown in the figure above, the following need to be performed:

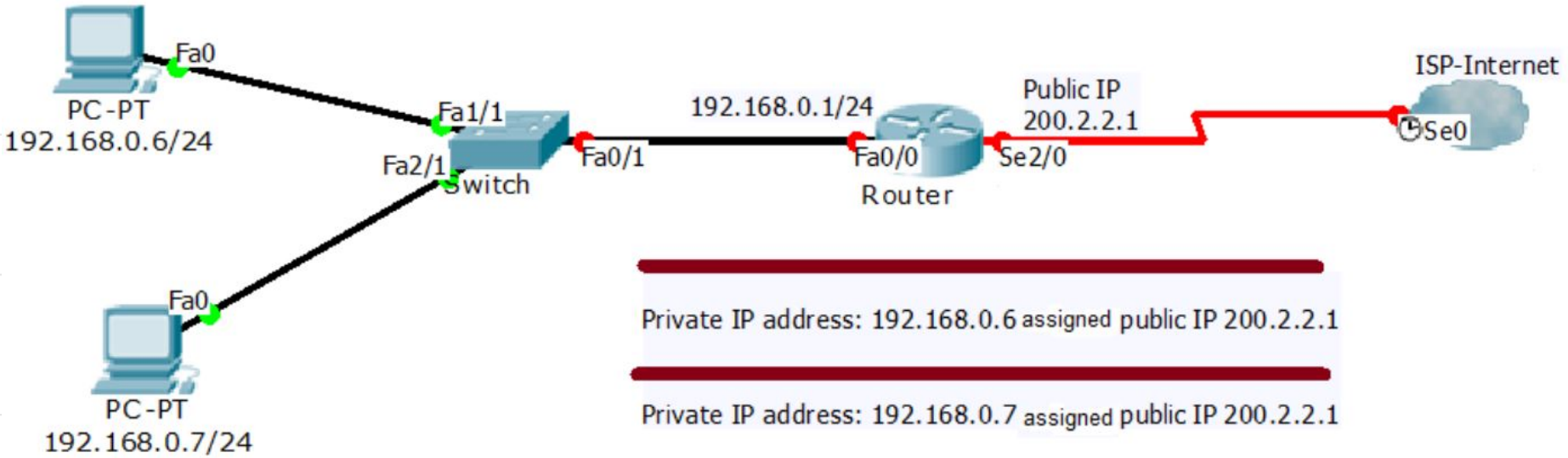
# Types of NAT

- **Dynamic NAT Configuration**
- **Specify the inside interface:**
  - Router(config)#interface fast ethernet0/0
  - Router(config-if)# ip nat inside
- **Specify the outside interface:**
  - Router(config)#interface serial0/0
  - Router(config-if) ip nat outside
- **Define an Access List to permit the inside local addresses to be translated:**
  - Router(config)#access-list 1 permit 10.0.0.0 0.0.0.255
- **Define a pool of global addresses:**
  - Router(config)# ip nat pool DNAT1 179.2.2.65 179.2.2.90 netmask 255.255.255.224
- **Enter dynamic translation entry:**
  - Router(config)# ip nat inside source list 1 pool DNAT1

# Types of NAT

## NAT overload (PAT)

- Port address translation is another variation of NAT and the most popular one.
- It is also called **NAT Overloading** because it is designed to **map many private IP addresses to just a single registered IP address** (overloaded address) by applying different port addresses in the TCP or UDP header.



# Types of NAT

- The first step in any NAT configuration is to define the inside and outside interfaces.
- **Set the fast ethernet 0/0 interface as the inside interface:**
  - R1# configure terminal
  - R1(config)# interface fastethernet0/0
  - R1(config-if)# ip nat inside
- **Next step is to set the serial interface S2/0 as the outside interface**
  - R1(config-if)# interface serial2/0
  - R1(config-if)# ip nat outside
- We now need to create an Access Control List (ACL) that will include local (private) hosts or network(s).
  - R1(config)# access-list 100 permit ip 192.168.0.0 0.0.0.255 any
- The above command instructs the router to allow the 192.168.0.0/24 network to reach any destination.
- All that's left now is to enable NAT overload and bind it to the outside interface previously selected:
  - R1(config)# ip nat inside source list 100 interface serial 2/0 overload

# Types of NAT

## Disadvantages of Address Translation

- Each connection has an added delay.
- Troubleshooting is more difficult.
- Not all applications work with address translation.

# Address Translation Terms

| Term | Explanation |
|---|---|
| Inside local IP address | The IPv4 address that is assigned to a host on the inside network |
| Inside global IP address | A legitimate IPv4 address assigned by the ISP that represents one or more inside local IPv4 addresses to the outside world |
| Outside global IP address | An outside device with a registered public IP address |
| Outside local IP address | An outside device with an assigned private IP address |


Thank You!