Computer Communication & Networks

Lecture 10, 11

 **Dynamic Routing**

Advantages of Dynamic over static:

• There is no need to know the destination networks.

• Need to advertise the directly connected networks.

• Updates the topology changes dynamically.

• Administrative work is reduced

• Used for large organizations.

• Neighbor routers exchange routing information and build the routing table

 automatically.

**Types of Dynamic Routing Protocols**

• Distance Vector Protocol

• Link State Protocol

• Hybrid Protocol



**Routing Information Protocol v1**

 Open Standard Protocol

• Classful routing protocol

• Updates are broadcasted via 255.255.255.255

• Administrative distance is 120

• Metric : Hop count

 Max Hop counts: 15 Max routers: 16(which is unreachable)

• Load Balancing up to 4 equal paths

• Used for small organizations

• Exchange entire routing table for every 30 seconds

**Rip Timers**

**Update timer** : 30 sec

– Time between consecutive updates

**Invalid timer** : 180 sec

– Time a router waits to hear updates

– The route is marked unreachable if there is no update during this interval.

**Flush timer** : 240 sec

– Time before the invalid route is purged from the routing table

**RIP Version 2**

• Classless routing protocol

• Supports VLSM

• Auto summary can be done on every router

• Supports authentication

• Trigger updates

• Uses multicast address 224.0.0.9.

**Advantages of RIP**

– Easy to configure

– No design constraints

– No complexity

– Less overhead

**Disadvantage of RIP**

– Bandwidth utilization is very high as broadcast for every 30 second

– Works only on hop count

– Not scalable as hop count is only 15

– Slow convergence

**Configuring RIP 1**

Router(config)# router rip

Router(config-router)# network <Network ID>

**Configuring RIP 2**

Router(config)# router rip

Router(config-router)# network <Network ID>

Router(config-router)# version 2



**On Hyderabad Router**

HYDERABAD # config t

HYDERABAD(config) # router rip

HYDERABAD(config-router) # network 10.0.0.0

HYDERABAD(config-router) # network 1.0.0.0

HYDERABAD(config-router) # exit

HYDERABAD(config) # exit

**On KSA Router**

KSA # config t

KSA(config) # router rip

KSA(config-router) # network 20.0.0.0

KSA(config-router) # network 1.0.0.0

KSA(config-router) # exit

KSA(config) # exit

**CONFIGURATION OF RIP V2**

**On Hyderabad Router**

HYDERABAD # config t

HYDERABAD(config) # router rip

HYDERABAD(config-router) # network 10.0.0.0

HYDERABAD(config-router) # network 1.0.0.0

HYDERABAD(config-router) # *Version 2*

HYDERABAD(config-router) # exit

HYDERABAD(config) # exit

**On KSA Router**

KSA # config t

KSA(config) # router rip

KSA(config-router) # network 20.0.0.0

KSA(config-router) # network 1.0.0.0

KSA(config-router) # *Version 2*

KSA(config-router) # exit

KSA(config) # exit

**Administrative Distance:**

It is the trustworthiness of the information received by the router.

The number is between 0 and 255.

Least value is more preferred.

Default administrative distances are as follows:

Directly connected=0

Static Route=1

IGRP=100

OSPF=110

RIP=120

EIGRP=90/170

IS-IS=115

**ACCESS CONTROL LIST**

• ACL is a set of rules which will allow or deny the specific traffic moving through the router

• It is a Layer 3 security which controls the flow of traffic from one router to another.

• It is also called as Packet Filtering Firewall.

****

**Rules of Access List**

•All deny statements have to be given First

• There should be at least one Permit statement

• An implicit deny blocks all traffic by default when there is no match (an invisible statement).

• Can have one access-list per interface per direction. (i.e.) Two access-lists per interface, one in inbound direction and one in outbound direction.

• Works in Sequential order

• Editing of access-lists is not possible (i.e) selectively adding or removing access list statements is not possible.

**Wild Card Mask**

Tells the router which addressing bits must match in the address of the ACL statement.

• It’s the inverse of the subnet mask, hence is also called as Inverse mask.

• A bit value of 0 indicates MUST MATCH (Check Bits)

• A bit value of 1 indicates IGNORE (Ignore Bits)

Wild Card Mask for a Host will be always 0.0.0.0

• A wild card mask can be calculated using the formula :

 **Global Subnet Mask**

**– Customized Subnet Mask**

**-------------------------------**

**Wild Card Mask**

E.g.

 255.255.255.255

– 255.255.255.240

**---------------------**

0. 0. 0. 15

**Creation of Standard Access List**

Router(config)# access-list <acl no> <permit/deny> <source address>

<source WCM>

**Implementation of Standard Access List**

Router(config)# interface <interface type> <interface no>

Router(config-if)# ip access-group <number> <out/in>

**To Verify :**

Router# show access-list

Router# show access-list <no>

**Creation of Extended Access List**

Router(config)# access-list <acl no> <permit/deny> <protocol>

<source address> <source wildcard mask>

<destination address> < destination wildcard mask> <operator>

<service>

**Implementation of Extended Access List**

Router(config)#interface <interface type> <interface no>

Router(config-if)#ip access-group <number> <out/in>

****

**Operators :**

eq (equal to)

neq (not equal to)

lt (less than)

gt (greater than)

**Named Access List**

• Access-lists are identified using Names rather than Numbers.

• Names are Case-Sensitive

• No limitation of Numbers here.

• One Main Advantage is Editing of ACL is Possible (i.e) Removing a specific statement

from the ACL is possible.

(IOS version 11.2 or later allows Named ACL)

**Creation of Standard Named Access List**

Router(config)# ip access-list standard <name>

Router(config-std-nacl)# <permit/deny> <source address> <source wildcard mask>

**Implementation of Standard Named Access List**

Router(config)#interface <interface type><interface no>

Router(config-if)#ip access-group <name> <out/in>

**Creation of Extended Named Access List**

Router(config)# ip access-list extended <name>

Router(config-ext-nacl)# <permit/deny> <protocol> <source address>

<source wildcard mask> <destination address>

< destination wildcard mask> <operator> <service>

**Implementation of Extended Named Access List**

Router(config)#interface <interface type><interface no>

Router(config-if)#ip access-group <name> <out/in>

****

**STANDARD ACCESS-LIST**

**Lab-1 Requirement**

1. Entire 20.0.0.0 network should be denied

2. Host 30.1.1.2 and 30.1.1.3 from UAE lan should be denied

**Creation of ACL :**

HYD (config) # access-list 5 deny 20.0.0.0 0.255.255.255

HYD (config) # access-list 5 deny 30.1.1.2 0.0.0.0

HYD (config) # access-list 5 deny host 30.1.1.3

HYD (config) # access-list 5 permit any

**Implementation**

HYD (config) # interface fa0/0

HYD (config-if ) # ip access-group 5 out

**Lab 2 :**

**Restricting telnet access to the router to specified networks or hosts**

**Creation of ACL :**

HYD (config) # access-list 10 pemit host 10.1.1.2

HYD (config) # access-list 10 pemit host 10.1.1.3

**Implementation**

CHE (config) # line vty 0 4

CHE (config-line) # access-class 1 in

CHE (config-line) # exit

CHE (config) # exit

**Extended ACL**

**Lab Requirment**

1. 10.1.1.2 should not ping 20.1.1.2

2. 30.0.0.0 NETWORK should not access 10.1.1.2 web service

**On HYD:**

HYD # config terminal

HYD (config) # access-list 100 deny icmp 10.1.1.2. 0.0.0.0 20.1.1.2 0.0.0.0 echo

HYD (config) # access-list 100 deny tcp host 10.1.1.2 30.0.0.0 0.255.255.255 eq 80

HYD (config) # access-list 100 permit ip any any

**Implementation**

HYD (config) #interface fa0/0

HYD (config-if) # ip access- group 100 in

HYD (config-if) #exit

HYD (config) # exit

HYD # sh ip access-list

**LAB Restricting Telnet Access**