Final Project: DHCP & ACL Configurations

Ebony Cross-Williams

Capitol Technology University: Course CT 240

Instructor: Professor Mehri

Date Lab was Performed: April 26th, 2017

Team Members: Alex Chong & Elkin Rios

Due Date: April 30th, 2017

Abstract

The objective of the laboratory exercise was to explore the structure, mechanisms, and implementation of IPv4 routing configurations for Router on a Stick (ROAS), DHCP, and firewalls. This laboratory exercise was represented by the goal to gain further knowledge about router configuration, implementation, and interconnectivity tests for router dynamic network management specifically using the Router on a Stick configuration while increasing network security features by creating a firewall on the router. From these tasks, students gain additional knowledge behind router management and network application. Through troubleshooting, students validated accurate configurations and connectivity, students used essential concepts to properly dynamically configure a single router for multiple virtual subnetworks.

Introduction

Internetwork router connections are widely used to connect to the Internet. Routers, as Layer 3 hardware devices of the OSI model, work to route packets between networks (Odom, 2013). A firewall is a product used at Layer 3 and 4 of the OSI model, to protect and improve network security from lesser secure Internet networks (Davis, 2009). To gain better knowledge of various IPv4 mechanisms and principles of communication, this paper will examine inter-VLAN configuration, dynamic network management, various controlled access mechanisms, and principles of network security, including the functionality network firewalls, implementation of traffic control as well the various types of firewalls used on routers.

The objective of the laboratory exercise was to explore the structure, mechanisms, and implementation of IPv4 routing's configurations for Router on a Stick (ROAS), DHCP, and firewalls. This laboratory exercise was represented by the goal to gain further knowledge about router configuration, implementation, and interconnectivity tests for router dynamic network management specifically using the Router on a Stick configuration while increasing network security features by creating a firewall on the router. From these tasks, students gain additional knowledge behind router management and network application. Through troubleshooting, students validated accurate configurations and connectivity, students used essential concepts to properly dynamically configure a single router for multiple virtual subnetworks. As a result, we will see how specific IPv4 routing principles can efficiently implement network communication. In addition, students implemented essential concepts behind dynamic router management and network application. Through troubleshooting and system commands, students validated accurate configurations and connectivity for a single router interconnected to multiple virtual subnetworks.

Equipment and Materials

Devices:

R1 (R0) F0/0.10 IP Address 192.168.10.1/24 & F0/0.20 IP Address 192.168.20.1/24

Switch (SW0) VLAN10 IP address 192.168.10.101/24

VLAN20 IP address 192.168.20.101/24

PC1 Model #6637-AB0, Serial #5560738 IP Address 192.168.10.26/24 Gateway 192.168.10.1

PC2 IP Address 192.168.20.26/24 Gateway 192.168.20.1

Procedure

Part A: Setup Laboratory materials and initial startup settings

The tasks performed in part A of the laboratory exercise were conducted using two assembled desktop computers and three CISCO 2600 routers. To begin, obtain one computer monitor, a computer power cord and other connections cords, one keyboard, a computer mouse, a computer hard drive with serial compatibility, as well as an extension cord to plug-in all the computer and network devices. Accurately assemble and connect the parts of the computer as instructed.

Now, gather all materials needed for the router device. Obtain one console cable to utilize their terminal emulator known as “Putty”. Connect the from the assembled computer NIC interface to the router Ethernet interface. Open the Putty application and select the serial option so that you may have a COM1. Once Putty opens, hit enter several times until the router hostname appears. Erasing an existing configuration by first typing enable. Check to see it the router is password protected, if so, enter the password. Then type erase startup-config. When prompted type the command reload.

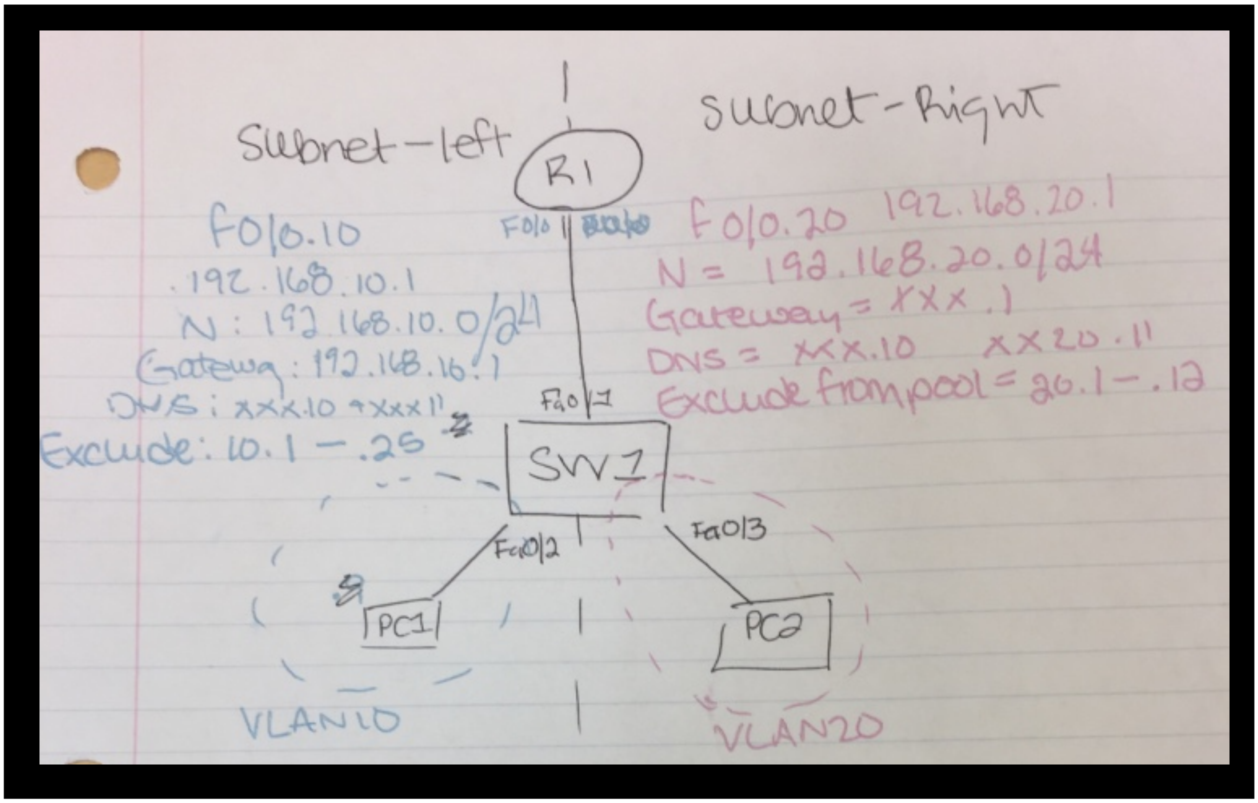
Figure 1: Erase existing configuration

```
Router#erase start
Erasing the nvram filesystem will remove all files! Continue? [confirm]
[OK]
Erase of nvram: complete
Router#
Router#sh start
%% Non-volatile configuration memory is not present
Router#
```

Note: factory reset was completed for router #1

Connect switch port fa0/1 to the interface FastEthernet0/0 on router R0. Use a straight-through cable for connection. Then, connect the personal computer labeled PC1 to switchport fa0/2 using a straight-through cable. Last, connect the personal computer labeled PC2 to switchport fa0/3 using a straight-through cable. For assigning IP addresses, the IP address scheme followed as seen in Figure 2: “IP Address Scheme”.

Figure 2: IP Address Scheme



Part B: Router on a Stick/Inter-VLAN Configuration

The first thing we must do is configure dynamic network system is to complete the Router on a Stick configuration (i.e. implementing inter-VLAN connection). The steps in this section will follow the procedures used in “Laboratory exercise #6: Router on a Stick”. To begin we will first create the vlans for the exercise (if they do not exist). While in global configuration mode, enter vlan id # (vlan 20). Then, add the vlan name by typing name [name chosen]. Next, exit. Repeat steps for both vlan 20 as well as vlan 10. configure the single switch IP and VLAN scheme.

Now, we can add ip addresses to the vlans. Go into global configuration mode. Enter interface vlan #id. For instance, enter int vlan 10.

Then, type ip address [ip address] [subnet mask]. Afterwards, type no shutdown and exit. Repeat step for both vlan 10 and vlan 20.

Figure 3: Creating VLANS and Assigning IP Address

Creating VLANs

```
GigabitEthernet0/1        unassigned      YES unset  down                  down

GigabitEthernet0/2        unassigned      YES unset  down                  down

sw0#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/4, Fa0/5, Fa0/6
                                                Fa0/7, Fa0/8, Fa0/9, Fa0/10
                                                Fa0/11, Fa0/12, Fa0/13, Fa0/14
                                                Fa0/15, Fa0/16, Fa0/17, Fa0/18
                                                Fa0/19, Fa0/20, Fa0/21, Fa0/22
                                                Fa0/23, Fa0/24, Gi0/1, Gi0/2
10   PC1                              active    Fa0/2
20   PC2                              active    Fa0/3
30   science                          active
40   art                              active
50   music                            active
99   native_vlan                      active
1002 fddi-default                     act/unsup
1003 token-ring-default               act/unsup
1004 fddinet-default                  act/unsup
1005 trnet-default                    act/unsup
sw0#
R1>
R1>
R1>
R1>en
```

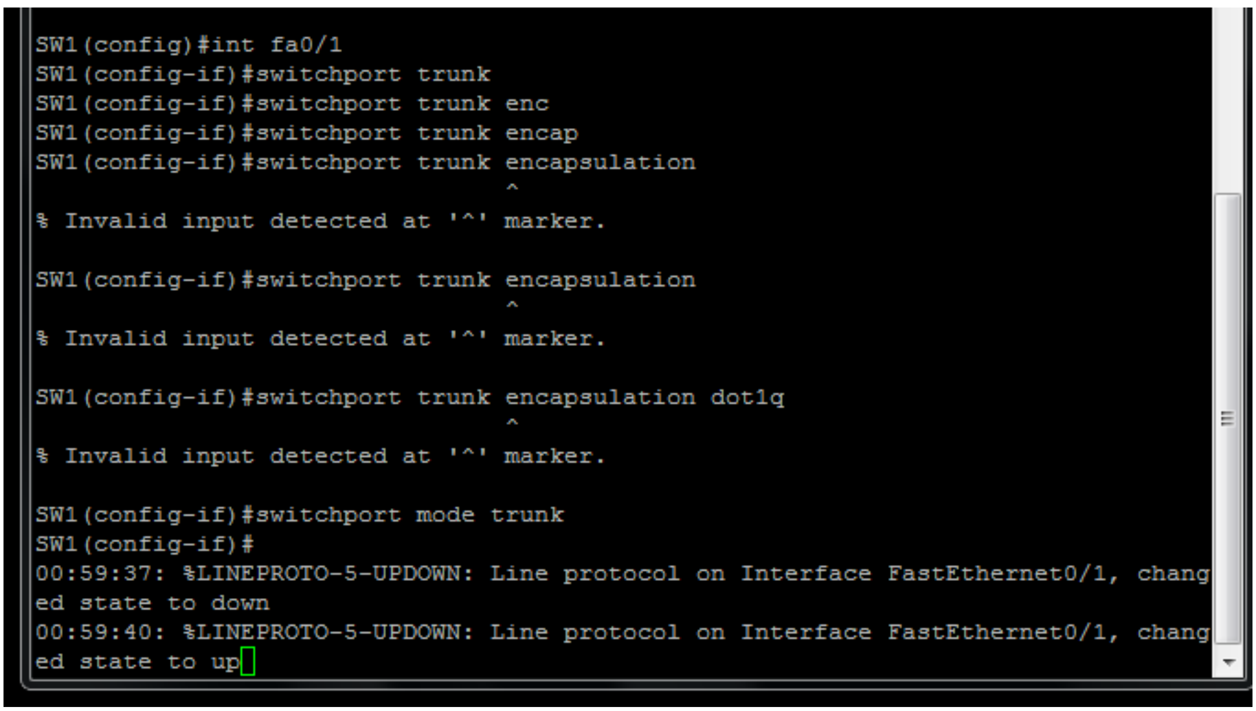
Assigning IP Address

```
sw0(config)#int vlan 10
sw0(config-if)#ip address 192.168.10.101 255.255.255.0
```

Note: VLAN10 IP Address was 192.168.10.101

Go to global configuration mode and change the name of the switch to SW0 using the command hostname SW0. Then exit out until you reach privilege mode. Next, we will add the specified vlans to the vlan database. Type the command, vlan database while in privilege mode. Then, enter the specific vlan id # such as entering vlan 10. A confirmation message will be immediately seen that stated the vlan has been added. Repeat step for VLAN 20. Note, ensure that the switch is in SERVER mode, otherwise the commands will not properly commit. On port fa0/0 of the switch, type switchport trunk encapsulation dot1q. Note, this last did not work on this switch model. Then, put the switchport into trunk mode by typing switchport mode trunk. Exit out of config-if mode.

Figure 4: Configure the Trunk port from switch SW0 to router R0



Note: switch SW0 command switchport trunk encapsulation dot1q was not recognized

Now that the VLANs are configured as well as the trunk link between the switch and the router, we are required to move some witch ports into the respective VLANs. For this exercise fa0/10 belongs to VLAN 10 and fa0/20 belongs to VLAN 20. These ports will become access ports for their corresponding VLANs. To begin, enter the specified port interface. Type switchport mode access. Then, enter switchport access vlan #id, to specify which VLAN you wish to configure. Then, type switchport mode access, exit and repeat for both VLAN 10 and 20. Once, the part is completed, enter wr in privilege mode to build configurations.

Figure 5: Apply switch interface to VLANs and set access

```
Enter configuration commands, one per line.  End with CNTL/Z.
sw0(config)#int  f0/2
sw0(config-if)#switchport acces vlan 10
sw0(config-if)#switchport mode access
sw0(config-if)#int f0/3
sw0(config-if)#switchport access vlan20
                                     ^
% Invalid input detected at '^' marker.

sw0(config-if)#switchport access vlan 20
sw0(config-if)#switchport mode access
sw0(config-if)#^Z
sw0#
```

Note: Access was set for VLAN 10 and VLAN 20

Part C: Router & DHCP Configuration

The VLANs have been configured in the switch database. We can now continue the ROAS setup on the router R0. Begin by changing the default name router to the hostname R0 (as referenced previously in this paper). Then, add VLANs 10 and 20 to the vlan database on the router. Enter the command vlan database and hit enter. Type the vlan #id such as vlan 10. Repeat steps for the other VLAN id number. Last, type apply and exit. You will see an output stating, “APPLY completed…” for confirmation.

Figure 6: Add VLAN 10 and VLAN 20 to vlan data on router R0

```
R0#vlan database
% Warning: It is recommended to configure VLAN from config mode,
  as VLAN database mode is being deprecated. Please consult user
  documentation for configuring VTP/VLAN in config mode.

R0(vlan)#vlan 10
VLAN 10 added:
    Name: VLAN0010
R0(vlan)#vlan 20
VLAN 20 added:
    Name: VLAN0020
R0(vlan)#apply
APPLY completed.
R0(vlan)#exit
APPLY completed.
Exiting....
R0#
```

Create sub-interfaces on the router for the VLANs recently added to the vlan database. The VLANs have a default gateway they will be added on the sub-interfaces to enable interconnecting communication between VLANs on the switch. Tell fa0/0 on router Ro to not have use of an IP address as well as turn on the physical interface. In global configuration mode, enter int fa0/0. Then, enter no ip address and no shutdown. Afterwards, exit. Configure the sub-interface for VLAN 10. Go to global configuration mode and type int fa0/0.10. Enter command encapsulation dot1q [vlan id number] (i.e. 10). Then enter the gateway ip address for vlan 10, stating ip address 192.168.10.1 255.255.255.0. Hit enter, type no shut.

Create a DHCP IP address pool for the valid IP addresses on VLAN10. In global configuration mode, type ip dhcp pool [name of pool]. The pool name for sub-interface fa0/0.10 was mypool. Next, give the network and subnet mask for the VLAN 10 that will be used to populate this pool. Specify the DNS domain name for practice as seen in the book *Cisco CCENT/CCNA ICND1 100-101 official Cert guide* (Odom, 2013). Type command domain-name [name of the DNS domain]. For this sub-interface, the DNS domain used is mydomain.com. Next, indicate the primary and secondary DNS servers using the command dns-server [primary ip address] [secondary ip address]. The ip address for the DNS servers were arbitrarily chosen. Afterwards, specify the default router (i.e. the default gateway) for the fa0/0.10 sub-interface. Enter default-router [default gateway IP address for VLAN 10]. Then, enter the lease duration for the address using the IP addresses that may be potentially used from the previously created pool. Once completed, exit pool configuration and return to the global configuration mode. Repeat steps for VLAN 20. Note, following steps in the book *Cisco CCENT/CCNA ICND1 100-101 official Cert guide* (Odom, 2013), a DNS domain name was not created.

Figure 7: DHCP Configuration on sub-interfaces of router R0

DHCP Configuration on sub-interface fa0/0.10

```
R1(config)#int f0/0.10
R1(config-subif)#encapsulation dot1q 10
R1(config-subif)#exit
R1(config)#ip address 192.168.10.1 255.255.255.0
                     ^
% Invalid input detected at '^' marker.

R1(config)#int f0/0.10
R1(config-subif)#192.168.10.1 255.255.255.0
                 ^
% Invalid input detected at '^' marker.

R1(config-subif)#ip address 192.168.10.1 255.255.255.0
R1(config-subif)#no shut
R1(config-subif)#exit
R1(config)#int f0/0.10
R1(config-subif)#ip dhcp pool mypool
R1(dhcp-config)#network 192.168.10.0 /24
R1(dhcp-config)#domain-name mydomain.com
R1(dhcp-config)#dns-server 192.168.10.10 192.168.10.11
R1(dhcp-config)#default-router 192.168.10.1
R1(dhcp-config)#lease 7
R1(dhcp-config)#exit
```

DHCP Configuration on sub-interface fa0/0.20

```
R1(config)#in f0/0.20
R1(config-subif)#encapsulation dot1q 20
R1(config-subif)#ip address 192.168.20.1 255.255.255.0
R1(config-subif)#ip dhcp pool subnet-vlan20
R1(dhcp-config)#network 192.168.20.0 /24
R1(dhcp-config)#dns-server 192.168.20.10 192.168.20.11
R1(dhcp-config)#default-router 192.168.20.1
R1(dhcp-config)#lease 1 2 3
R1(dhcp-config)#no shutdown
                    ^
% Invalid input detected at '^' marker.

R1(dhcp-config)#exit
R1(config)#int f0/0.20
R1(config-subif)#no shut
R1(config-subif)#exit
R1(config)#
```

Note: To configure interVLAN connection, the fastEthernet 0/0 on the router's physical interface must be turned on using the command "no shut" on that interface

Once basic setup properly for the router on a stick configuration is complete add additional DHCP features by excluded ip addresses from the pool ranges. Type the command ip dhcp excluded-address [first IP address] [last IP address]. Repeat command for both pools (i.e. IP addresses in both VLAN subnetworks) as seen in Figure 8: "IP Address Exclusion".

Figure 8: IP Address Exclusion

```
.g-subif)#exit
.g)#ip dhcp excluded-address 192.168.10.1 192.168.10.25
.g)#ip dhcp excluded-address 192.168.20.1 192.168.20.12
.g)#exit
```

Note: VLAN10 will exclude IP Address from 192.168.10.1 to 192.168.10.25, while VLAN20 will exclude IP Address from 192.168.20.1 to 192.168.20.12

Part D: Dynamically Configuring IP addresses for End Nodes

It is time to dynamic configure IP addresses for PC1 and PC2. Go to the start menu and open the control panel. Find the network connectivity directory and open the adapter settings. Right-click on the Local Connection button and click properties. Click the IPv4 option in the menu list, and then click properties. Unlike previous laboratory exercises, the exercise will use the option to dynamically assign the IP Address for the end node. To verify, the IP address for the end open the command prompt and step the command ipconfig /renew. Then, type ipconfig and view output for verification. The end nodes should now be able to ping each other's ip addresses in opposing VLANs subnetworks. Thus, inter-VLAN routing and DHCP router configuration has been successfully confirmed.

Part E: Implementing a firewall on the Router

Create a firewall by implementing the feature of access-control lists or ACLs. On router R0, enter global configuration mode. For this exercise, we will create a Standard ACL. Furthermore, for this exercise, create a ping filtration by using the command access-list [#1-99] {deny | permit} [ip address on filter list] [inverted subnet mask]. For instance, for our team, we specifically implemented the command access-list 2 deny 192.168.20.0 0.0.0.255. Next, implement this created ACL on vlan 10. While still on the router, enter the sub-interface for the VLAN 10 (i.e. fa0/0.10). Type the command ip access-group [ACL#] {in | out}. In our exercise, we will specify "in" for inbound communications, for this command. Then type end. To list some resulting output from router R0 that shows information about the implemented ACL(s), type show ip access-lists, while in privilege mode. To view the ACL(s), you can also try the command show ip int fa0/0.10 brief, while in privilege mode.

Figure 9: Creating Standard Number ACL

```
R1(config)#access-list 2 deny 192.168.20.0 0.0.0.255
R1(config)#int fa0/0.10
R1(config-subif)#ip access-group 2 in
R1(config-subif)#end
R1#
```

Once everything has been setup properly for the router on a stick configuration, DHCP, and ACL configurations, confirm interconnectivity (or the lack there off) between VLANs. To verify, end nodes should ping each other’s ip addresses in opposing VLANs subnetworks. Thus, successfully completing all supplemental tasks to verify inter-VLAN, DHCP, and ACL configuration.

Results

The laboratory exercise had serval steps of router configuration for inter-VLAN routing, DHCP, and ACL through exploration and implementation. To begin, we wiped any previous configuration from the startup-configuration. Then, we explored the different configuration needed for a single switch with multiple VLANs with a trunk link to a single router. We observed, that the router on a stick configuration allowed routing management between virtual local area networks. The next steps in the exercise followed the procedures covered in "Laboratory exercise #6: Router on a Stick" and therefore will not be focus on for this section of the paper. For additional details on results, for the submitted paper, "Laboratory exercise #6: Router on a Stick". After configuring ROAS, the next step on the router was to implement DHCP.

First, DHCP IP address pool was created for the valid IP addresses on VLAN10. The pool name for sub-interface fa0/0.10 was mypool. This step was resulted for VLAN 20 on sub-interface fa0/0.20. We confirmed the configured range of IP addresses and other corresponding statistics for each pool using the command show ip dhcp pool [poolname].

Figure 10: Confirming pools of IP addresses for VLAN 10 & VLAN 20 on sub-interfaces

```
R1#show ip dhcp pool mypool

Pool mypool :
 Utilization mark (high/low)    : 100 / 0
 Subnet size (first/next)       : 0 / 0
 Total addresses                : 254
 Leased addresses               : 0
 Pending event                  : none
 1 subnet is currently in the pool :
 Current index        IP address range                    Leased addresses
 192.168.10.1         192.168.10.1     - 192.168.10.254    0
R1#show ip dhcp pool subnet-vlan20

Pool subnet-vlan20 :
 Utilization mark (high/low)    : 100 / 0
 Subnet size (first/next)       : 0 / 0
 Total addresses                : 254
 Leased addresses               : 0
 Pending event                  : none
 1 subnet is currently in the pool :
 Current index        IP address range                    Leased addresses
 192.168.20.1         192.168.20.1     - 192.168.20.254    0
R1#
```

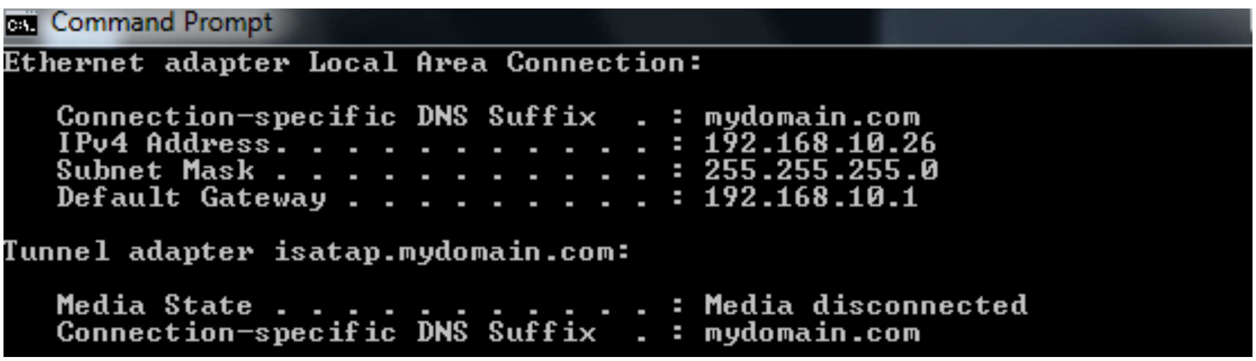
Note: Used command show ip dhcp pool [poolname]

Next, we gave the network and subnet mask for the VLAN 10 that would be used to populate this pool. We Specified the DNS domain name for practice as seen in the book *Cisco CCENT/CCNA ICND1 100-101 official Cert guide* (Odom, 2013). For this sub-interface, the DNS domain used is mydomain.com. Next, we indicated the primary and secondary DNS servers. The ip address for the DNS servers was arbitrarily chosen in which it followed examples in the book *Cisco CCENT/CCNA ICND1 100-101 official Cert guide* (Odom, 2013).

Afterwards, we specified the default router (i.e. the default gateway) for the fa0/0.10 sub-interface. Then, enter the lease duration for the address using the IP addresses that may be potentially used from the previously created pool. Once completed, exit pool configuration and return to the global configuration mode. Steps were repeated for VLAN 20. Note, following steps in the book *Cisco CCENT/CCNA ICND1 100-101 official Cert guide* (Odom, 2013), a DNS domain name was not created for VLAN 20. Once basic setup properly for the router on a stick configuration is complete add additional DHCP features by excluded ip addresses from the pool ranges. We could confirm the DHCP configurations including configuration of the domain, default gateway, and excluded IP address pool using the end nodes. On each end node, the command *ipconfig* was used to output the resulting configurations as seen in Figure 10: "Verifying DHCP configurations".

Figure 11: Verifying DHCP configurations

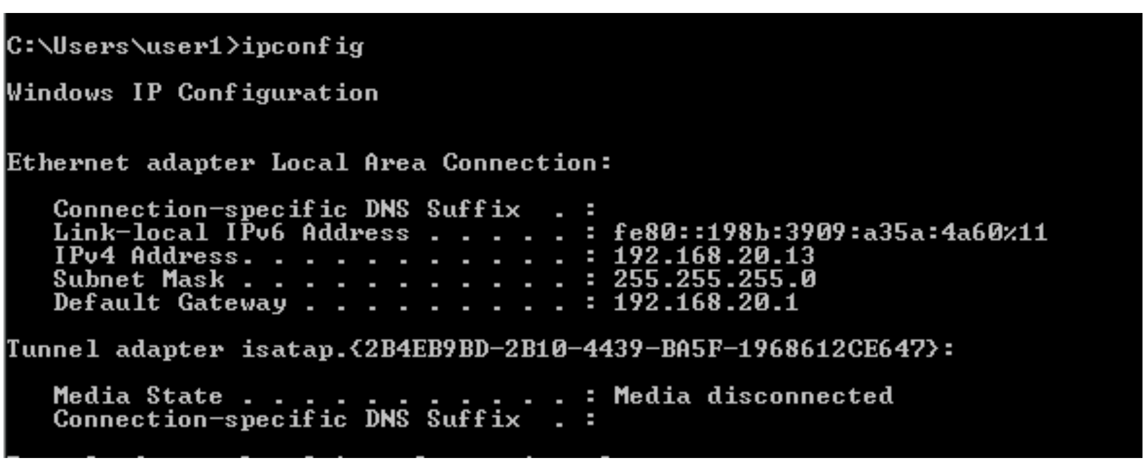
PC1 on VLAN10



```
Command Prompt
Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . : mydomain.com
   IPv4 Address. . . . . . . . . . . : 192.168.10.26
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.10.1

Tunnel adapter isatap.mydomain.com:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : mydomain.com
```

Note: PC1 on VLAN10 excluding IP Address from 192.168.10.1 to 192.168.10.25

PC2 on VLAN20



```
C:\Users\user1>ipconfig

Windows IP Configuration


Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::198b:3909:a35a:4a60%11
   IPv4 Address. . . . . . . . . . . : 192.168.20.13
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.20.1

Tunnel adapter isatap.{2B4EB9BD-2B10-4439-BA5F-1968612CE647}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :
```

Note: PC2 on VLAN20 excluding IP Address from 192.168.20.1 to 192.168.20.12

In addition, the end nodes could ping each other's ip addresses in opposing VLANs subnetworks. Thus, inter-VLAN routing and DHCP router configuration had been successfully confirmed.

Figure 12: Confirming DHCP and Inter-VLAN routing connectivity

PC1 pinging PC2

```
Ethernet adapter Local Area Connection:

   Connection-specific DNS Suffix  . : mydomain.com
   IPv4 Address. . . . . . . . . . . : 192.168.10.26
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.10.1

Tunnel adapter isatap.mydomain.com:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : mydomain.com

C:\Users\Student>ping 192.168.20.13

Pinging 192.168.20.13 with 32 bytes of data:
Reply from 192.168.20.13: bytes=32 time=3ms TTL=127
Reply from 192.168.20.13: bytes=32 time<1ms TTL=127
Reply from 192.168.20.13: bytes=32 time<1ms TTL=127
Reply from 192.168.20.13: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.20.13:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 3ms, Average = 0ms
```

Note: PC1 had an IP address of 192.168.10.26, active in VLAN 10.

PC2 pinging PC1

```
C:\Users\user1>ping 192.168.10.1

Pinging 192.168.10.1 with 32 bytes of data:
Reply from 192.168.10.1: bytes=32 time=1ms TTL=255
Reply from 192.168.10.1: bytes=32 time=1ms TTL=255
Reply from 192.168.10.1: bytes=32 time=1ms TTL=255
Reply from 192.168.10.1: bytes=32 time=1ms TTL=255

Ping statistics for 192.168.10.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms
```

Note: PC2 had an IP address of 192.168.20.13, active in VLAN 20.

The last supplemental task involved creating a firewall on the router by implementing the feature of access-control lists or ACLs. For this exercise, we created a Standard ACL for inbound communication between VLAN 10's subnet and VLAN 20. Furthermore, we implemented an access-control list that would internally block any pings (communication) from VLAN 10 to VLAN 20. We confirmed our ACL as seen in Figure 12: "Display ACL Information" that gave some resulting output from router R0 about the implemented ACL.

Figure 13: Display ACL Information

```
R1#show ip access-list
Standard IP access list 1
    10 permit 192.168.10.26
    20 deny   0.0.0.0, wildcard bits 255.255.255.0
    30 deny   0.0.0.30, wildcard bits 255.255.255.0
    40 deny   192.168.20.0, wildcard bits 0.0.0.255
    50 deny   192.168.10.0, wildcard bits 0.0.0.255
Standard IP access list 2
    10 deny   192.168.20.0, wildcard bits 0.0.0.255
Standard IP access list 3
    10 permit 192.168.10.1
Extended IP access list 101
    10 deny icmp any any
```

Note: Only IP access list #2 was implemented on a sub-interface of fa0/0 on router R0

The ACL created and implemented on sub-interface fa0/0.10 was also verified using the command show ip int fa0/0.10 brief, while in privilege mode.

Figure 14: Verify ACL Information for sub-interface fa0/0.10

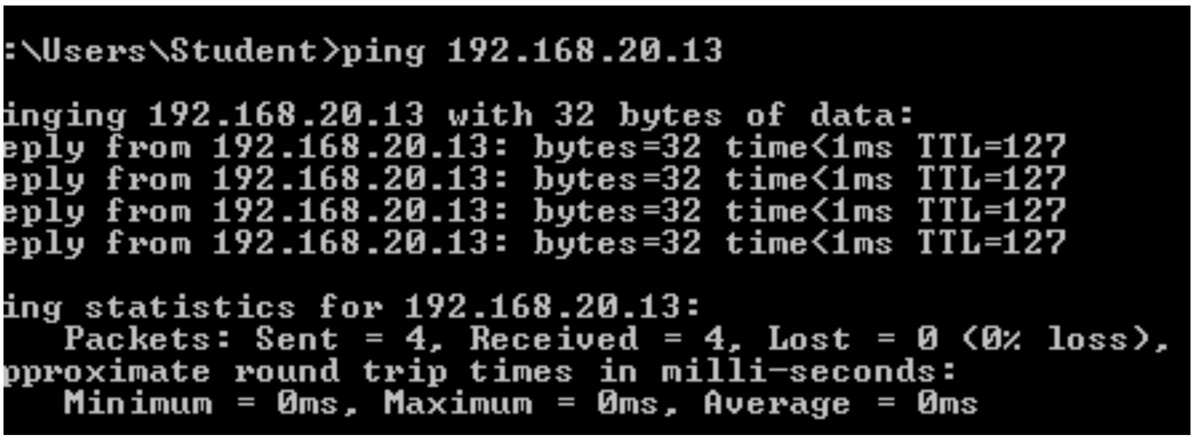
```
R1#show ip int fa0/0.10
FastEthernet0/0.10 is up, line protocol is up
  Internet address is 192.168.10.1/24
  Broadcast address is 255.255.255.255
  Address determined by setup command
  MTU is 1500 bytes
  Helper address is not set
  Directed broadcast forwarding is disabled
  Outgoing access list is 3
  Inbound  access list is 2
  Proxy ARP is enabled
  Local Proxy ARP is disabled
  Security level is default
  Split horizon is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
  ICMP mask replies are never sent
  IP fast switching is enabled
  IP fast switching on the same interface is enabled
  IP Flow switching is disabled
  IP CEF switching is enabled
  IP CEF Feature Fast switching turbo vector
  IP multicast fast switching is enabled
  IP multicast distributed fast switching is disabled
  IP route-cache flags are Fast, CEF
  Router Discovery is disabled
  IP output packet accounting is disabled
  IP access violation accounting is disabled
  TCP/IP header compression is disabled
  RTP/IP header compression is disabled
  Policy routing is disabled
  Network address translation is disabled
  BGP Policy Mapping is disabled
  WCCP Redirect outbound is disabled
  WCCP Redirect inbound is disabled
  WCCP Redirect exclude is disabled
R1#
```

Once everything had been setup properly for the router on a stick configuration, DHCP, and ACL configurations, we confirmed interconnectivity (or the lack there off) between VLANs. This ACL would deny IP packets coming from other hosts in the 192.168.20.0 network (i.e.

VLAN 20 subnet). To verify, end node PC1 in VLAN 10 pinged the PC2 end in VLAN 20 subnetwork. The ping connectivity was denied after applied the ACL inbound. Thus, successfully completing all supplemental tasks to verify inter-VLAN, DHCP, and ACL configuration.

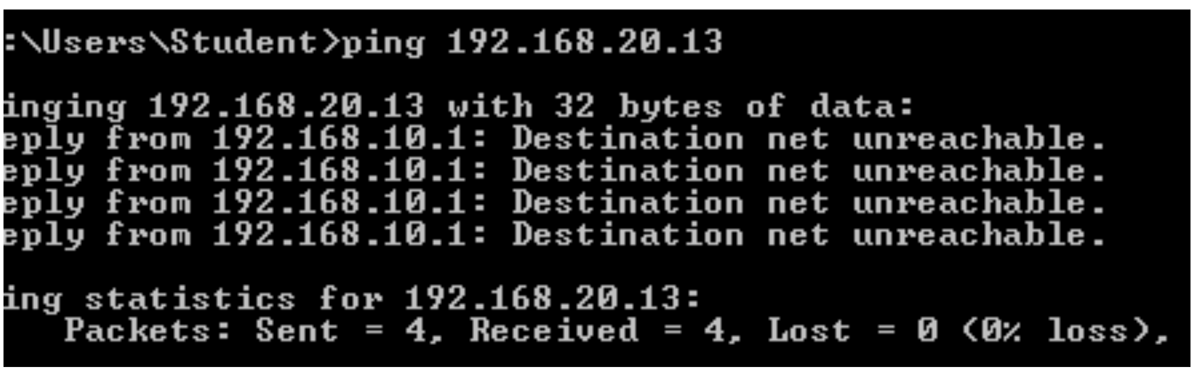
Figure 15: Verifying ACL Ping Deny

Ping from PC1 to PC2 BEFORE implementing ACL



```
:\Users\Student>ping 192.168.20.13

inging 192.168.20.13 with 32 bytes of data:
eply from 192.168.20.13: bytes=32 time<1ms TTL=127
eply from 192.168.20.13: bytes=32 time<1ms TTL=127
eply from 192.168.20.13: bytes=32 time<1ms TTL=127
eply from 192.168.20.13: bytes=32 time<1ms TTL=127

ing statistics for 192.168.20.13:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
pproximate round trip times in milli-seconds:
   Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Ping from PC1 to PC2 AFTER implementing ACL



```
:\Users\Student>ping 192.168.20.13

inging 192.168.20.13 with 32 bytes of data:
eply from 192.168.10.1: Destination net unreachable.
eply from 192.168.10.1: Destination net unreachable.
eply from 192.168.10.1: Destination net unreachable.
eply from 192.168.10.1: Destination net unreachable.

ing statistics for 192.168.20.13:
   Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

Note: Ping command from PC1 to PC2 was successfully denied

Analysis

For this laboratory exercise, there were serval steps of router configuration for inter-VLAN routing through exploration and implementation. However, the analysis for the Router on the Stick configuration was covered in "Laboratory exercise #6: Router on a Stick" and therefore will not be focus on for this section of the paper. For additional details on results, for the submitted paper, "Laboratory exercise #6: Router on a Stick". After configuring ROAS, the next step on the router was to implement DHCP. According the Odom (2013), Dynamic Host Configuration Protocol (DHCP) works automatically for user hosts which has many advantages over static IPv4 settings. It is a form of technology feature in IPv4 settings that will automatically implement permanent IP address assignments as well as temporary lease of IP addresses. For to allow DHCP configurations on a router to efficiently assignment addresses, specific IPv4 settings must first be indicated to provide proper information.

Part A: DHCP Configuration

To begin the configurations, DHCP IP address pool was created for the valid IP addresses on VLAN10. The pool name for sub-interface fa0/0.10 was mypool. This step was resulted for VLAN 20 on sub-interface fa0/0.20. DHCP requires the and collection (or pool) of IP address be created for the addresses the network administrator wishes to use for the network. The network administrator creates a pool name and then on a given sub-interfaces must indicate the subnet ID and subnet mask. This allows router R0 running DHCP to use this information to know the desire IP addresses in the subnetwork (Odom, 2013). By default, the router assumes all IP addresses are valid for lease unless explicitly excluded (Odom, 2013). We Specified the DNS domain name for practice as seen in the book *Cisco CCENT/CCNA ICND1 100-101 official Cert guide* to learn how to define the DNS domain name on the command line (Odom, 2013). Next, we indicated the primary and secondary DNS servers. DHCP allows the network administrator to create a list of DNS server IP addresses (Odom, 2013). In this exercise, we create two DNS server IP addresses manually. Afterwards, we specified the default router (i.e. the default gateway) for the both sub-interfaces. For each sub-interface the given default router (i.e. default gateway) was the IP address of the router on that subnetwork corresponding the VLAN subnets.

Then, we entered the lease duration for the address using the IP addresses that may be potentially used from the previously created pool. The length of the lease is formatted in the order of days, hours, and minutes (Odom, 2013). Thus, sub-interface fa0/0.10 the lease was 7 days and the lease for sub-interface fa0/0.20 was 1 day, 2 hours, and 3 minutes. Once completed, DHCP added excluded ip addresses from the pool ranges. The DHCP configure router R0, must know which IP addresses in the subnets for the VLANs should not be leased, otherwise DHCP will assume all address assignments are valid (Odom, 2013). The list allowed some of the addresses, such as DNS servers and default router address to be reserve and unable to be assigned automatically (Odom, 2013). We could confirm the DHCP configurations including configuration of the domain, default gateway, and excluded IP address pool using the end nodes. As seen in the Results sections, as estimated, the DHCP leased the first non-reserved IP address on each end node. In addition, the end nodes could ping each other's ip addresses in opposing VLANs subnetworks. Thus, inter-VLAN routing and DHCP router configuration had been successfully confirmed.

Part B: Firewall on Router Configuration

A firewall is a product used at Layer 3 and 4 of the OSI model, to protect and improve network security from lesser secure Internet networks (Davis, 2009). Firewalls can keep track of traffic and deter unwanted network traffic from the external networks from penetrating the private LANs created (Davis, 2009). Therefore, firewalls are a security feature of routers that may be apply to regulation and filter packet transmissions.

The last supplemental task involved creating a firewall on the router by implementing the feature of access-control lists or ACLs. For this exercise, we created a Standard ACL for inbound communication between VLAN 10's subnet and VLAN 20. Furthermore, we implemented an access-control list that would internally block any pings (communication) from VLAN 10 to VLAN 20. As stated, routers may apply regulation to packet transmissions which includes inbound and outbound traffic filtrations. The security feature of firewalls specifically known as Access Control Lists (ACLS) are used to implement this form of traffic control. ACLs act as a firewall for control traffic coming in and going out of one or more subnetwork. A network firewall will use multiple ACLs as traffic control mechanisms. ACLs are subsequently a set of rules used to regulate/specify directional traffic permissions ("Routers and Routing Basics"). Furthermore, ACLs are sequential lists of permits or deny statements that apply information received for the packet headers to deliberate access. Once the ACLs have been compile, the routers may be using them to compare incoming packets against each statement in the list, sequentially from top-down ("Routers and Routing Basics").

The inbound Standard numbered ACL denial was created and implemented on sub-interface fa0/0.10. Standard ACLs are capable of being cached in the various high-speed caches that router have supported, including fast switching, autonomous switching, silicon switching, optimum switching, and others ("Types of ACLs"). The Standard ACL statements are group either by number of by name ("Types of ACLs"). In addition, with Standard ACLs, you can also specify only a source address and a wildcard mask ("Types of ACLs). The IP lists uses numbers 1 through 99 to which statements are applied closest to the destination host/subnet whose access is to be restricted (“Routers and Routing Basics”). This ACL (i.e. ACL: access-list deny 2 192.168.20.0 0.0.0.255) would deny IP packets coming from other hosts in the 192.168.20.0 network (i.e. VLAN 20 subnet). To verify, end node PC1 in VLAN 10 pinged the PC2 end in VLAN 20 subnetwork. The ping connectivity as shown in the Result section, was denied after applied the ACL inbound. Therefore, the network filter control used to show a simplistic firewall on the router was successfully created and conducted. Output of supplemental tasks confirmed all inter-VLAN, DHCP, and ACL configurations.

Conclusion

The objective of the laboratory exercise was to explore the structure, mechanisms, and implementation of IPv4 routing's configurations for Router on a Stick (ROAS), DHCP, and firewalls. This laboratory exercise was represented by the goal of gaining further knowledge about router configuration, implementation, and interconnectivity tests for router dynamic network management specifically using the Router on a Stick configuration while increasing network security features by creating a firewall on the router. From these tasks, students gained additional knowledge behind router management and network application.

Through troubleshooting, students validated accurate configurations and connectivity, students used essential concepts to properly dynamically configure a single router for multiple virtual subnetworks. As a result, we saw how specific IPv4 routing principles can efficiently implement network communication. In addition, students implemented essential concepts behind dynamic router management and network application. Furthermore, we observe how firewalls are a security feature of routers that may be apply to regulation and filter packet transmissions. As a result, we will saw how using firewalls on routers can enhance protection of internal LANs from the outside networks, regulation traffic transmissions inbound and outbound in the internal networks as well as implementation traffic control using ACLs. Through troubleshooting and system commands, students validated accurate configurations and connectivity for a single router interconnected to multiple virtual subnetworks. Overall, this exercise showed how the configuration and implementation of inter-VLAN, DHCP, and ACL configurations may give routers enhanced network security and efficiency in packet transmission.

Reference

Odom, W. (2013). *Cisco CCENT/CCNA ICND1 100-101 official Cert guide*. Indianapolis, IN: Cisco Press.

Davis, D. *Routers, Switches & Firewalls – Learn how they are different*. (2009, February 11). Retrieved April 25, 2017, from https://www.petri.com/csc_routers_switches_and_firewalls

Permitting or Denying Network Access.(n.d.) Retrieved April 25, 2017, from http://www.cisco.com/c/en/us/td/docs/security/fwsm/fwsm31/configuration/guide/fwsm_cfg/nwacc_f.pdf

W. (n.d.). Routers and Routing Basics - Access Control Lists (ACLs). Retrieved April 25, 2017, from http://iamechatronics.com/notes/general-engineering/551-routers-and-routing-basics-access-control-lists-acls

Types of ACLs. (n.d.). Retrieved April 25, 2017, from http://etutorials.org/Networking/Router firewall security/Part III Nonstateful Filtering Technologies/Chapter 7. Basic Access Lists/Types of ACLs/

Killian, S. (n.d.). *Router on a Stick CCNA Tutorial*. Retrieve April 24th, 2017 from http://www.shanekillian.net

How to configure Router on a Stick. (2017, February 15). Retrieved April 23, 2017, from https://networklessons.com/cisco/ccna-routing-switching-icnd1-100-105/how-to-configure-router-on-a-stick/