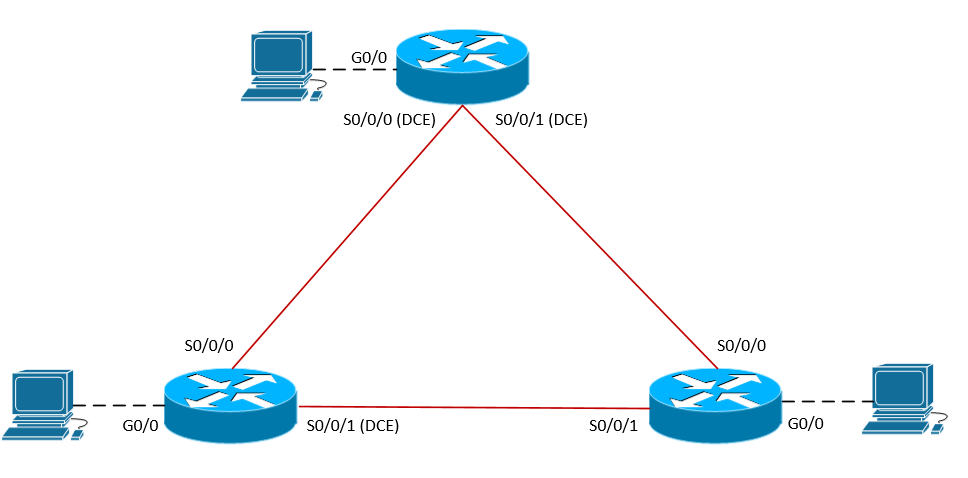
Packet Tracer – Advanced IPv6 (DHCPv6 & Access Control)

(Instructor Version)

**Instructor Note**: Red font color indicate text that appears in the instructor copy only.

Topology



1. Scenario

NetVise Corporation has hired a new security manager who would like to implement several new control policies for the IPv6 network. You have been placed in charge of implementing the required access control lists according to the given specifications alongside a new DHCPv6 server. The security manager has also requested that the IPv6 security implementation is fully documented and verified.   
  
Prior to starting this lab, your manager provided you with some supporting training material. You are expected to review the provided material thoroughly before starting this lab.

1. Addressing Table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Device | Interface | Type | IP Address | Prefix | Default Gateway |
| S1-RTR | S0/0/0 | Global Unicast | 2001:c1c0:34:12::1 | /64 | N/A |
| S0/0/1 | Global Unicast | 2001:c1c0:34:13::1 | /64 | N/A |
| G0/0 | Global Unicast | 2001:c1c0:34:1::1 | /64 | N/A |
| S1-PC | NIC | Static | 2001:c1c0:34:1::100 | /64 | 2001:c1c0:34:1::1 |
| S2-RTR | S0/0/0 | Global Unicast | 2001:c1c0:34:12::2 | /64 | N/A |
| S0/0/1 | Global Unicast | 2001:c1c0:34:23::2 | /64 | N/A |
| G0/0 | Global Unicast | 2001:c1c0:34:2::1 | /64 | N/A |
| S2-PC | NIC | Static | 2001:c1c0:34:2::100 | /64 | 2001:c1c0:34:2::1 |
| S3-RTR | S0/0/0 | Global Unicast | 2001:c1c0:34:13::3 | /64 | N/A |
| S0/0/1 | Global Unicast | 2001:c1c0:34:23::3 | /64 | N/A |
| G0/0 | Global Unicast | 2001:c1c0:34:3::1 | /64 | N/A |
| S3-PC | NIC | DHCPv6 |  |  |  |

1. Objectives

* Configure DHCPv6 server and verify address allocation.
* Verify routing and connectivity (preconfigured).
* Configure IPv6 access control lists (ACLs)
* Securing VTY Lines using Access Lists

**Task 1: Configure DHCPv6 server and verify address allocation.**

**Step 1:** Configure S3-RTR with the following parameters to allocate an IPv6 address to S3-PC.

* IPv6 Address Pool: 2001:c1c0:34:3::/64 (assigned with a prefix length of 64).
* Domain Name: netspace.com
* DNS Server: 2001:4860:4860::8888

**S3-RTR(config)#** ipv6 dhcp pool DHCPV6

S3-RTR(config-dhcp)# prefix-delegation pool ADDRESS\_POOL

S3-RTR(config-dhcp)# dns-server 2001:4860:4860::8888

S3-RTR(config-dhcp)# domain-name netspace.com

!  
S3-RTR(config)# ipv6 local pool ADDRESS\_POOL 2001:c1c0:34:3::/64 64

!

S3-RTR(config)# interface GigabitEthernet0/0

S3-RTR(config-if)# ipv6 dhcp server DHCPv6

**S3-RTR# show ipv6 dhcp pool**

DHCPv6 pool: DHCPV6

Prefix pool: ADDRESS\_POOL

preferred lifetime 604800, valid lifetime 2592000

DNS server: 2001:4860:4860::8888

Domain name: netspace.com

Active clients: 1

**S3-RTR# show ipv6 dhcp binding**

Client: 0010.11C2.36AA (GigabitEthernet0/0)

DUID: 00030001001011C236AA

IA PD: IA ID 0, T1 0, T2 0

Prefix: 2001:C1C0:34:3::1/64

preferred lifetime 604800, valid lifetime 2592000

expires at August 29 2013 3:22:32 pm (2592000 seconds)

**S3-RTR# show ipv6 dhcp interface**

GigabitEthernet0/0 is in server mode

Using pool: DHCPv6\_POOL

Preference value: 0

Hint from client: ignored

Rapid-Commit: disabled

**S3-PC>ipv6config**

FastEthernet0 Connection:(default port)

Link-local IPv6 Address.........: FE80::210:11FF:FEC2:36AA

IPv6 Address....................: 2001:C1C0:34:3:210:11FF:FEC2:36AA/64

Default Gateway.................: FE80::207:ECFF:FE26:ED01

**Task 2: Verify routing and connectivity**

**Step 1:** All interfaces have been preconfigured according to the address table and routing has been enabled using EIGRPv6. After DHCPv6 is configured to allocate an address to S3-PC, you should be able to ping both S1-PC and S2-PC sourcing from S3-PC.

**S3-PC>ping 2001:c1c0:34:1::100**

Pinging 2001:c1c0:34:1::100 with 32 bytes of data:

Reply from 2001:C1C0:34:1::100: bytes=32 time=1ms TTL=126

Reply from 2001:C1C0:34:1::100: bytes=32 time=2ms TTL=126

Reply from 2001:C1C0:34:1::100: bytes=32 time=5ms TTL=126

Reply from 2001:C1C0:34:1::100: bytes=32 time=5ms TTL=126

Ping statistics for 2001:C1C0:34:1::100:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 1ms, Maximum = 5ms, Average = 3ms

**S3-PC>ping 2001:c1c0:34:2::100**

Pinging 2001:c1c0:34:2::100 with 32 bytes of data:

Reply from 2001:C1C0:34:2::100: bytes=32 time=1ms TTL=126

Reply from 2001:C1C0:34:2::100: bytes=32 time=7ms TTL=126

Reply from 2001:C1C0:34:2::100: bytes=32 time=5ms TTL=126

Reply from 2001:C1C0:34:2::100: bytes=32 time=4ms TTL=126

Ping statistics for 2001:C1C0:34:2::100:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 1ms, Maximum = 7ms, Average = 4ms

**Task 3: Configure DHCPv6 server and verify address allocation.**

**Step 1:** Configure an IPv6 access control list (ACL) to block all traffic sourcing from S3-PC DHCP pool to S1-PC. Permit all other traffic. Remember, you should place source based access-control lists as close as possible to the destination.

**S1-RTR(config)#** ipv6 access-list DENY\_S3\_DHCP

S1-RTR(config-ipv6-acl)# deny ipv6 2001:C1C0:34:3::/64 any

S1-RTR(config-ipv6-acl)# permit ipv6 any any

!

S1-RTR(config)# interface g0/0

S1-RTR(config-if)# ipv6 traffic-filter DENY\_S3\_DHCP out

S1-RTR# show ipv6 access-list

IPv6 access list DENY\_S3\_DHCP

deny ipv6 2001:C1C0:34:3::/64 any (5 match(es))

permit ipv6 any any (5 match(es))

**S3-PC>** ping 2001:c1c0:34:1::100

Pinging 2001:c1c0:34:1::100 with 32 bytes of data:

Reply from 2001:C1C0:34:13::1: Destination host unreachable.

Reply from 2001:C1C0:34:13::1: Destination host unreachable.

Reply from 2001:C1C0:34:13::1: Destination host unreachable.

Reply from 2001:C1C0:34:13::1: Destination host unreachable.

Ping statistics for 2001:C1C0:34:1::100:

Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

**Step 2:** Configure an access control list (ACL) to block TCP applications HTTP & FTP traffic sourcing from S3-PC’s specific IPv6 address when destined for S2-RTR’s G0/0 LAN subnet. Permit all other traffic. Remember, you should place source/destination based access-control lists as close as possible to the source. Using a traffic generator on S3-PC you can send traffic with destination ports of 80 and 21 respectfully, if you notice the access-list matches increment, you have configured the task correctly and the appropriate traffic is being blocked.

**S3-RTR(config)#** ipv6 access-list DENY\_S3\_TO\_S2

S3-RTR(config-ipv6-acl)# deny tcp host 2001:C1C0:34:3:210:11FF:FEC2:36AA 2001:C1C0:34:2::/64 eq www

S3-RTR(config-ipv6-acl)# deny tcp host 2001:C1C0:34:3:210:11FF:FEC2:36AA 2001:C1C0:34:2::/64 eq ftp

S3-RTR(config-ipv6-acl)# permit ipv6 any any

**S3-RTR#** show ipv6 access-list

IPv6 access list DENY\_S3\_TO\_S2

deny tcp host 2001:C1C0:34:3:210:11FF:FEC2:36AA 2001:C1C0:34:2::/64 eq www **(24 match(es))**

deny tcp host 2001:C1C0:34:3:210:11FF:FEC2:36AA 2001:C1C0:34:2::/64 eq ftp **(4 match(es))**

permit ipv6 any any

**S3-PC>** ping 2001:c1c0:34:2::100

Pinging 2001:c1c0:34:2::100 with 32 bytes of data:

Reply from 2001:C1C0:34:2::100: bytes=32 time=2ms TTL=126

Reply from 2001:C1C0:34:2::100: bytes=32 time=1ms TTL=126

Reply from 2001:C1C0:34:2::100: bytes=32 time=5ms TTL=126

Reply from 2001:C1C0:34:2::100: bytes=32 time=1ms TTL=126

Ping statistics for 2001:C1C0:34:2::100:

Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

Minimum = 1ms, Maximum = 5ms, Average = 2ms

**Task 4: Securing VTY lines using access lists**

**Step 1:** Telnet is preconfigured on S1-RTR, restrict access to only allow management access from S2-PC and deny all other sources.

**S1-RTR(config)#** ipv6 access-list MGMT\_ACCESS

S1-RTR(config-ipv6-acl)# permit tcp 2001:C1C0:34:2::/64 any eq telnet

S1-RTR(config-ipv6-acl)# deny ipv6 any any

!

S1-RTR(config)# line vty 0 4

S1-RTR(config-line)# ipv6 access-class MGMT\_ACCESS in

S1-RTR# show access-lists

IPv6 access list MGMT\_ACCESS

permit tcp 2001:C1C0:34:2::/64 any eq telnet

deny ipv6 any any