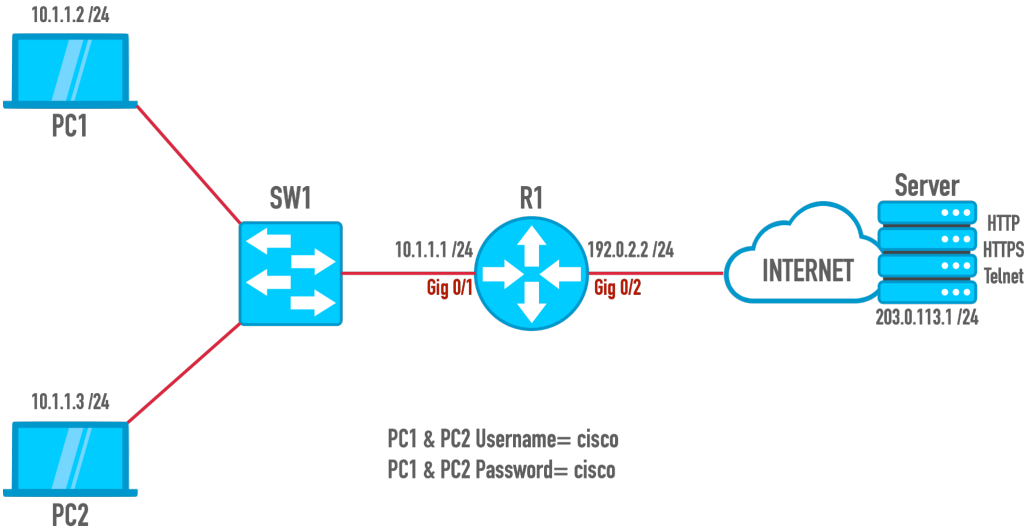
Extended Named ACL

Topology



Initial Configuration Commands

PC1:

```
sudo ifconfig eth0 10.1.1.2 netmask 255.255.255.0 up
sudo route add default gw 10.1.1.1
sudo hostname PC1
```

PC2:

```
sudo ifconfig eth0 10.1.1.3 netmask 255.255.255.0 up
sudo route add default gw 10.1.1.1
sudo hostname PC2
```

SW1:

```
enable
conf t
no ip domain-lookup
logging console
line con 0
logging synchronous
```

```
exec-timeout 0 0
hostname SW1
end
copy run star
```

R1:

```
enable
conf t
host R1
no banner motd
no banner login
no banner exec
no banner incoming
line vty 0 15
password cisco
login
exec-timeout 0 0
transport input telnet
line con 0
logging synchronous
exit
no ip domain-lookup
ipv6 unicast-routing
int gig 0/1
ip address 10.1.1.1 255.255.255.0
ipv6 address 2000:2::1/64
no shutdown
int gig 0/2
ip address 192.0.2.2 255.255.255.0
ipv6 address 2000:1::1/64
no shutdown
exit
ipv6 router rip ROUTE
int gig 0/1
ipv6 rip ROUTE enable
exit
int gig 0/2
ipv6 rip ROUTE enable
exit
router ospf 1
```

```
network 0.0.0.0 255.255.255.255 area 0
end
copy run star
```

SERVER:

```
enable
conf t
host SERVER
no banner motd
no banner login
no banner exec
no banner incoming
line vty 0 15
password cisco
login
exec-timeout 0 0
transport input telnet
line con 0
logging synchronous
exit
no ip domain-lookup
ipv6 unicast-routing
int gig 0/1
ip address 192.0.2.1 255.255.255.0
ipv6 address 2000:1::2/64
no shutdown
int lo0
ip address 203.0.113.1 255.255.255.0
ipv6 address 2000:A::1/64
exit
ipv6 router rip ROUTE
int gig 0/1
ipv6 rip ROUTE enable
exit
int lo0
ipv6 rip ROUTE enable
exit
router ospf 1
network 0.0.0.0 255.255.255.255 area 0
exit
```

```
ip http server
ip http secure-server
end
copy run star
```

Lab Tasks

- Ping the server from both PC1 and PC2 to make sure we can reach the server.
- Create an extended named access control list and name it BLOCK_PC1_SERVICES on R1.
- Keep PC1 from contacting the server using Telnet.
- Permit all other traffic to the server.
- Apply our access control list, BLOCK_PC1_SERVICES, on the inbound direction of interface Gig 0/1 on R1.
- Take a look at the ACL that we just created.
- Add a sequence number of 15 to our ACL to keep PC1 from contacting the server using HTTP.
- Take another look at the ACL that we just created to see if our new addition is there.
- Telnet to the server by using the port numbers of HTTPS, HTTP, and Telnet and see if we are able to connect to the server on PC1.
- Telnet to the server by using the port numbers of HTTPS, HTTP, and Telnet and see if we are able to connect to the server on PC2.

Solution

Step 1: Ping the server from both PC1 and PC2 to make sure we can reach the server.

PC1

```
PC1 login: cisco
Password: cisco

PC1:~$ ping 203.0.113.1
PING 203.0.113.1 (203.0.113.1): 56 data bytes
64 bytes from 203.0.113.1: seq=5 ttl=42 time=98.571 ms
64 bytes from 203.0.113.1: seq=6 ttl=42 time=22.357 ms
64 bytes from 203.0.113.1: seq=7 ttl=42 time=142.706 ms
64 bytes from 203.0.113.1: seq=8 ttl=42 time=63.266 ms
^C
--- 203.0.113.1 ping statistics ---
9 packets transmitted, 4 packets received, 55% packet loss
round-trip min/avg/max = 22.357/81.725/142.706 ms
```

PC2

```
PC2 login: cisco
Password: cisco

PC2:~$ ping 203.0.113.1
PING 203.0.113.1 (203.0.113.1): 56 data bytes
64 bytes from 203.0.113.1: seq=1 ttl=42 time=12.136 ms
64 bytes from 203.0.113.1: seq=2 ttl=42 time=11.563 ms
64 bytes from 203.0.113.1: seq=3 ttl=42 time=11.592 ms
64 bytes from 203.0.113.1: seq=4 ttl=42 time=10.863 ms
^C
--- 203.0.113.1 ping statistics ---
5 packets transmitted, 4 packets received, 20% packet loss
round-trip min/avg/max = 10.863/11.538/12.136 ms
```

Step 2: Create an extended named access control list and name it BLOCK_PC1_SERVICES on R1.

```
R1>en
R1#conf t
R1(config)#ip access-list ?
  extended    Extended Access List
  helper      Access List acts on helper-address
  log-update  Control access list log updates
  logging     Control access list logging
  resequence  Resequence Access List
  standard    Standard Access List

R1(config)#ip access-list extended BLOCK_PC1_SERVICES
R1(config-ext-nacl)#

(## See how we are now in Extended Named ACL configuration
mode.)
```

Step 3: Keep PC1 from contacting the server using Telnet.

```
R1(config-ext-nacl)#?
Ext Access List configuration commands:
  <1-2147483647>  Sequence Number
  default         Set a command to its defaults
  deny            Specify packets to reject
  dynamic         Specify a DYNAMIC list of PERMITs or DENYs
  evaluate        Evaluate an access list
  exit            Exit from access-list configuration mode
  no              Negate a command or set its defaults
```

```
  permit          Specify packets to forward
  remark          Access list entry comment

R1(config-ext-nacl)#deny tcp host 10.1.1.2 host 203.0.113.1 eq 23
```

Step 4: Permit all other traffic to the server from PC1.

```
R1(config-ext-nacl)#permit ip any any
```

Step 5: Apply our access control list, BLOCK_PC1_SERVICES, on the inbound direction of interface Gig 0/1 on R1.

```
R1(config-ext-nacl)#int gig 0/1
R1(config-if)#ip access-group BLOCK_PC1_SERVICES in
R1(config-if)#end
```

Step 6: Take a look at the ACL that we just created.

```
R1#show access-list
Extended IP access list BLOCK_PC1_SERVICES
    10 deny tcp host 10.1.1.2 host 203.0.113.1 eq telnet
    20 permit ip any any (39 matches)
```

Step 7: Add a sequence number of 15 to our ACL to keep PC1 from contacting the server using HTTP.

```
R1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#ip access-list extended BLOCK_PC1_SERVICES
R1(config-ext-nacl)#?
Ext Access List configuration commands:
  <1-2147483647>  Sequence Number
  default         Set a command to its defaults
  deny            Specify packets to reject
  dynamic         Specify a DYNAMIC list of PERMITs or DENYs
  evaluate        Evaluate an access list
  exit            Exit from access-list configuration mode
  no              Negate a command or set its defaults
  permit          Specify packets to forward
  remark          Access list entry comment

R1(config-ext-nacl)#15 deny tcp host 10.1.1.2 host 203.0.113.1 eq 80
R1(config-ext-nacl)#end
```

Step 8: Take another look at the ACL that we just created to see if our new addition is there.

```
R1#show access-list
Extended IP access list BLOCK_PC1_SERVICES
    10 deny tcp host 10.1.1.2 host 203.0.113.1 eq telnet
    15 deny tcp host 10.1.1.2 host 203.0.113.1 eq www
    20 permit ip any any (165 matches)
```

Step 9: Telnet to the server by using the port numbers of HTTPS, HTTP, and Telnet and see if we are able to connect to the server on PC1.

HTTPS

```
PC1:~$ telnet 203.0.113.1 443
Connected to 203.0.113.1
```

HTTP

```
PC1:~$ telnet 203.0.113.1 80
telnet: can't connect to remote host (203.0.113.1): Host is unreachable
```

Telnet

```
PC1:~$ telnet 203.0.113.1 23
telnet: can't connect to remote host (203.0.113.1): Host is unreachable
```

Step 10: Telnet to the server by using the port numbers of HTTPS, HTTP, and Telnet and see if we are able to connect to the server on PC2.

HTTPS

```
PC2:~$ telnet 203.0.113.1 443
Connected to 203.0.113.1
```

HTTP

```
PC2:~$ telnet 203.0.113.1 80
Connected to 203.0.113.1
```

Telnet

```
PC2:~$ telnet 203.0.113.1 23
Connected to 203.0.113.1
```