

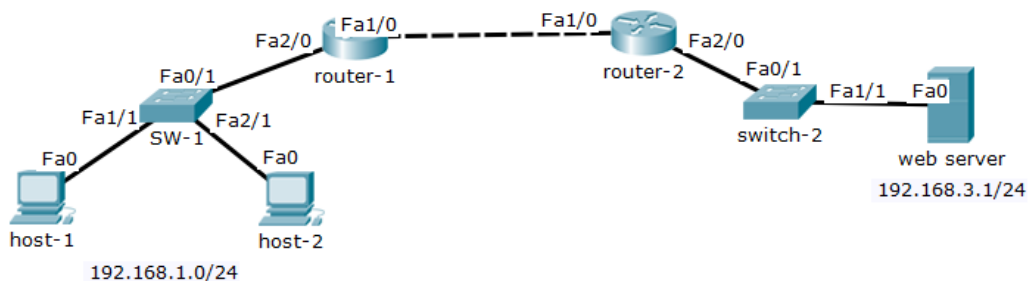
# Extended Named ACL

## Lab Summary

Configure a named access control list (ACL) to filter traffic based on the following requirements:

1. Configure a named ACL called **http-telnet-filter**
2. Add a remark that describes the purpose of the ACL
3. Deny all HTTP traffic from hosts on 192.168.1.0/24 subnets to web server
4. Deny Telnet sessions from hosts on 192.168.1.0/24 subnets to all routers
5. Permit all traffic not matching on any previous ACL statements
6. Apply the named ACL **http-telnet filter** inbound on router-1 interface Fa2/0

**Figure 1** Lab Topology



## Lab Configuration

Start Packet Tracer File: **Named ACL**

Verify that there is web server and Telnet access permitted from host-1 and host-2.

host-1: **http://192.168.3.1** (yes)

host-1: c:\> **telnet 192.168.2.1** (yes) password: *cisco*

host-1: c:\> **telnet 192.168.2.2** (yes) password: *cisco*

host-2: **http://192.168.3.1** (yes)

host-2: c:\> **telnet 192.168.2.1** (yes) password: *cisco*

host-2: c:\> **telnet 192.168.2.2** (yes) password: *cisco*

Click on the *router-1* icon and select the *CLI* folder. Hit the <enter> key for user mode prompt (>).

Step 1: Enter global configuration mode

```
router-1> enable  
Password: cisco  
router-1# configure terminal
```

Step 2: Create a named ACL called *http-telnet-filter* and add a remark to explain.

```
router-1(config)# ip access-list extended http-telnet-filter  
router-1(config-ext-nacl)# remark deny access to web server and telnet
```

Step 3: Deny HTTP from hosts on 192.168.1.0/24 subnet to the web server

```
router-1(config-ext-nacl)# deny tcp 192.168.1.0 0.0.0.255 host 192.168.3.1 eq  
www
```

Step 4: Deny Telnet sessions from hosts on 192.168.1.0/24 subnet to any router.

```
router-1(config-ext-nacl)# deny tcp 192.168.1.0 0.0.0.255 any eq telnet
```

Step 5: Permit all traffic that does not match any previous ACL statements.

```
router-1(config-ext-nacl)# permit ip any any  
router-1(config-ext-nacl)# exit
```

Step 6: Apply the named ACL inbound on router-1 interface Fa2/0 and save it to the running configuration.

```
router-1(config)# interface fastethernet2/0  
router-1(config-if)# ip access-group http-telnet-filter in  
router-1(config-if)# end  
router-1# copy running-config startup-config
```

Step 7: Verify Lab

Verify the ACL configuration is correct and enabled. Start the web browser from host-1 and host-2 and verify access to web server is permitted. Confirm there is no Telnet access permitted from hosts to any router. Telnet access is available from the routers. ACL is applied inbound on router-1 and does not filter past that point.

```
router-1# show running-config  
  
router-1# show access-lists  
  
Extended IP access list http-telnet-filter  
10 deny tcp 192.168.0.0 0.0.255.255 host 192.168.3.1 eq www  
20 deny tcp 192.168.0.0 0.0.255.255 any eq telnet  
30 permit ip any any
```

```
host-1: http://192.168.3.1 (no)
host-1: c:\> telnet 192.168.2.1 (no)
host-1: c:\> telnet 192.168.2.2 (no)
host-2: http://192.168.3.1 (no)
host-2: c:\> telnet 192.168.2.1 (no)
host-2: c:\> telnet 192.168.2.2 (no)
router-1# telnet 192.168.2.2 (yes)
```

Remove ACL from router-1 interface FastEthernet2/0 and verify that access is now available from hosts.

```
router-1(config)# no ip access-group http-telnet-filter in
```

### Lab Notes

Apply any extended ACL near the source to filter packets immediately and minimize bandwidth usage. The **show access-lists** command is available for troubleshooting packet filtering that includes each statement per ACL and the specific order. The access control list (ACL) statements are numbered starting with 10 and verify how packet matching is occurring. Note as well that deleting a statement and adding a new statement will assign the ACL to the bottom of the list. That could then affect how packet matching is occurring and inadvertently permit/deny wrong traffic.