Firewall Configuration and Testing Report

1. Objective

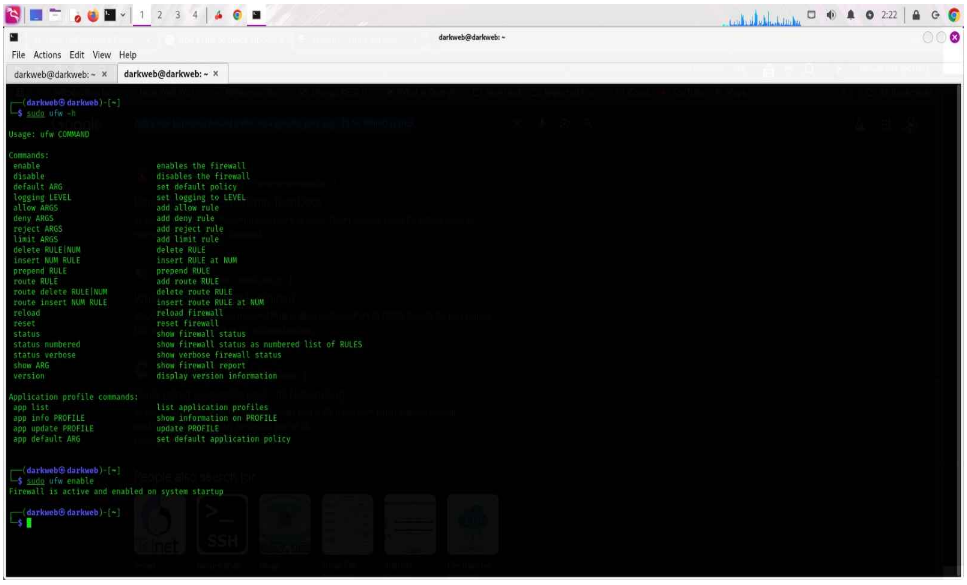
To configure the firewall to block specific ports, test the configuration, and understand how firewall rules filter traffic.

2. Tools Used

OS: Linux (UFW - Uncomplicated Firewall)
Testing Tools: telnet, nc (Netcat)

Step 1 – Check UFW Status

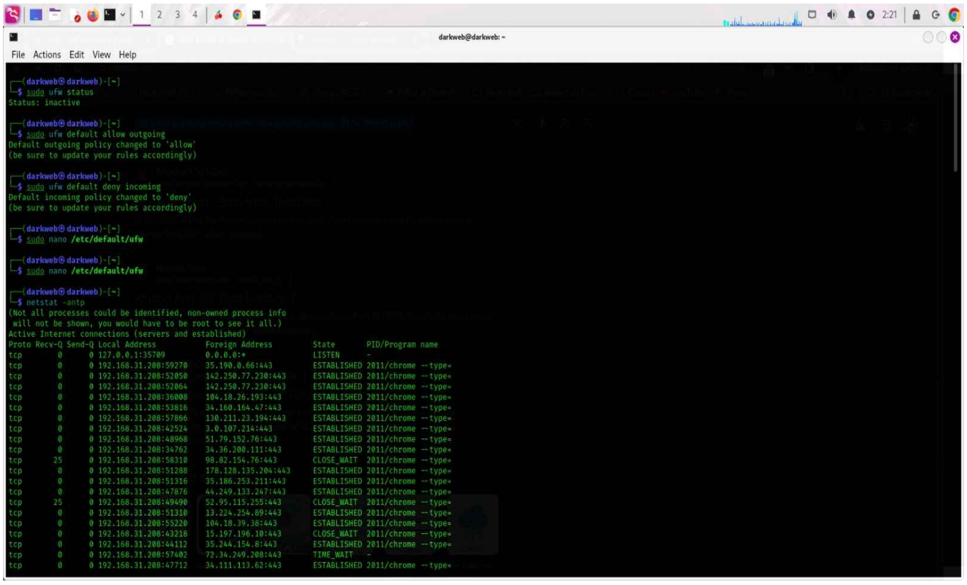
Command:
`sudo ufw status`



```
┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw -h

Usage: ufw COMMAND

Commands:
 enable                          enables the firewall
 disable                         disables the firewall
 default ARG                     set default policy
 logging LEVEL                   set logging to LEVEL
 allow ARGS                      add allow rule
 deny ARGS                       add deny rule
 reject ARGS                     add reject rule
 limit ARGS                      add limit rule
 delete RULE|NUM                 delete RULE
 insert NUM RULE                 insert RULE at NUM
 prepend RULE                    prepend RULE
 route RULE                      add route RULE
 route delete RULE|NUM           delete route RULE
 route insert NUM RULE           insert route RULE at NUM
 reload                          reload firewall
 reset                           reset firewall
 status                          show firewall status
 status numbered                 show firewall status as numbered list of RULES
 status verbose                  show verbose firewall status
 show ARG                        show firewall report
 version                         display version information

Application profile commands:
 app list                        list application profiles
 app info PROFILE                show information on PROFILE
 app update PROFILE              update PROFILE
 app default ARG                 set default application policy


┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw enable
Firewall is active and enabled on system startup

┌──(darkweb㉿darkweb)-[~]
└─$ 
```

Step 2 – List Current Firewall Rules

Command:
`sudo ufw status numbered`

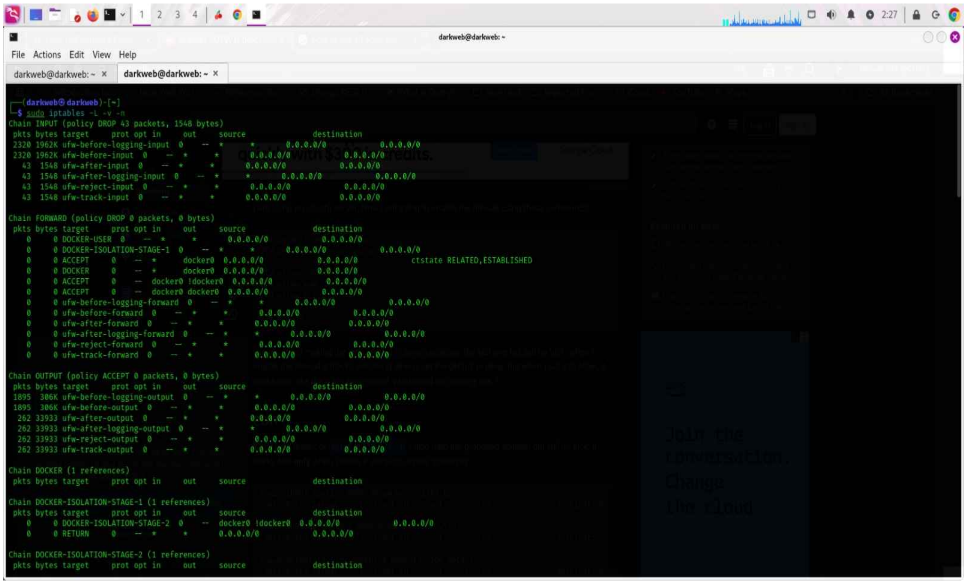


```
┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw status
Status: inactive

┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw default allow outgoing
Default outgoing policy changed to 'allow'
(be sure to update your rules accordingly)

┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)

┌──(darkweb㉿darkweb)-[~]
└─$ sudo nano /etc/default/ufw

┌──(darkweb㉿darkweb)-[~]
└─$ sudo nano /etc/default/ufw

┌──(darkweb㉿darkweb)-[~]
└─$ netstat -antp
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:35709         0.0.0.0:*               LISTEN      -
tcp        0      0 192.168.31.208:59270    35.190.0.66:443         ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:52050    142.250.77.230:443      ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:52064    142.250.77.230:443      ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:36008    104.18.26.193:443       ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:53816    34.160.164.47:443       ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:57866    130.211.23.194:443      ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:42524    3.0.107.214:443         ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:48968    51.79.152.76:443        ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:34762    34.36.200.111:443       ESTABLISHED 2011/chrome --type=
tcp       25      0 192.168.31.208:58310    98.82.154.76:443        CLOSE_WAIT  2011/chrome --type=
tcp        0      0 192.168.31.208:51288    178.128.135.204:443     ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:51316    35.186.253.211:443      ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:47876    44.249.133.247:443      ESTABLISHED 2011/chrome --type=
tcp       25      0 192.168.31.208:49490    52.95.115.255:443       CLOSE_WAIT  2011/chrome --type=
tcp        0      0 192.168.31.208:51310    13.224.254.89:443       ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:55220    104.18.39.38:443        ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:43218    15.197.196.10:443       CLOSE_WAIT  2011/chrome --type=
tcp        0      0 192.168.31.208:44112    35.244.154.8:443        ESTABLISHED 2011/chrome --type=
tcp        0      0 192.168.31.208:57402    72.34.249.208:443       TIME_WAIT   -
tcp        0      0 192.168.31.208:47712    34.111.113.62:443       ESTABLISHED 2011/chrome --type=
```

Step 3 – Add Rule to Block Port 23 (Telnet)

Command:

```
sudo ufw deny 23/tcp
```

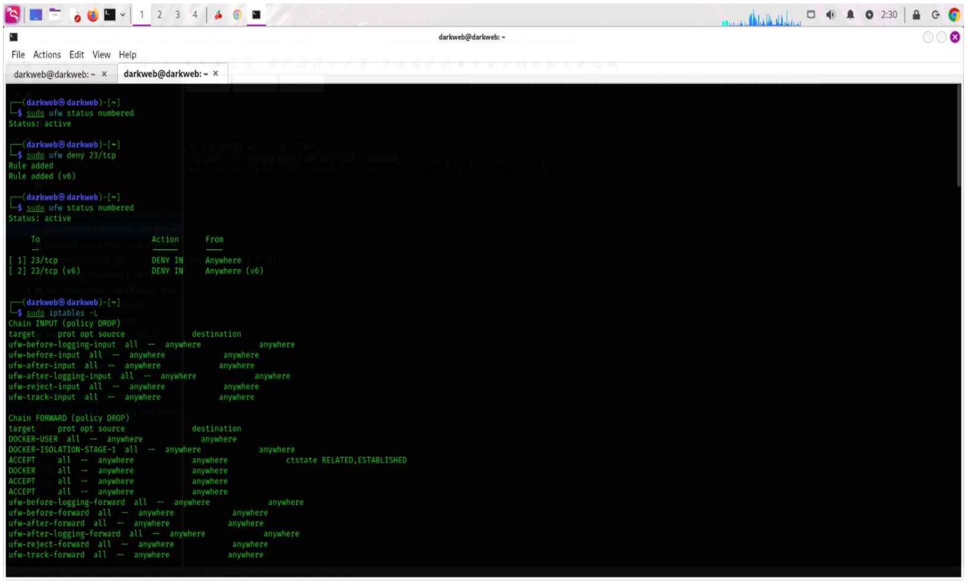


```
┌──(darkweb㉿darkweb)-[~]
└─$ sudo iptables -L -v -n
Chain INPUT (policy DROP 43 packets, 1548 bytes)
 pkts bytes target     prot opt in     out     source               destination
 2320 1962K ufw-before-logging-input  0    --  *      *       0.0.0.0/0            0.0.0.0/0
 2320 1962K ufw-before-input  0    --  *      *       0.0.0.0/0            0.0.0.0/0
   43  1548 ufw-after-input  0    --  *      *       0.0.0.0/0            0.0.0.0/0
   43  1548 ufw-after-logging-input  0    --  *      *       0.0.0.0/0            0.0.0.0/0
   43  1548 ufw-reject-input  0    --  *      *       0.0.0.0/0            0.0.0.0/0
   43  1548 ufw-track-input  0    --  *      *       0.0.0.0/0            0.0.0.0/0

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 DOCKER-USER  0    --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 DOCKER-ISOLATION-STAGE-1  0    --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ACCEPT     0    --  *      docker0  0.0.0.0/0            0.0.0.0/0            ctstate RELATED,ESTABLISHED
    0     0 DOCKER     0    --  *      docker0  0.0.0.0/0            0.0.0.0/0
    0     0 ACCEPT     0    --  docker0 !docker0  0.0.0.0/0            0.0.0.0/0
    0     0 ACCEPT     0    --  docker0 docker0  0.0.0.0/0            0.0.0.0/0
    0     0 ufw-before-logging-forward  0    --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-before-forward  0    --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-after-forward  0    --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-after-logging-forward  0    --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-reject-forward  0    --  *      *       0.0.0.0/0            0.0.0.0/0
    0     0 ufw-track-forward  0    --  *      *       0.0.0.0/0            0.0.0.0/0

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
 1895  306K ufw-before-logging-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
 1895  306K ufw-before-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
  262 33933 ufw-after-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
  262 33933 ufw-after-logging-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
  262 33933 ufw-reject-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0
  262 33933 ufw-track-output  0    --  *      *       0.0.0.0/0            0.0.0.0/0

Chain DOCKER (1 references)
 pkts bytes target     prot opt in     out     source               destination

Chain DOCKER-ISOLATION-STAGE-1 (1 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 DOCKER-ISOLATION-STAGE-2  0    --  docker0 !docker0  0.0.0.0/0            0.0.0.0/0
    0     0 RETURN     0    --  *      *       0.0.0.0/0            0.0.0.0/0

Chain DOCKER-ISOLATION-STAGE-2 (1 references)
 pkts bytes target     prot opt in     out     source               destination
```

Step 4 – Verify Rule Added

Command:

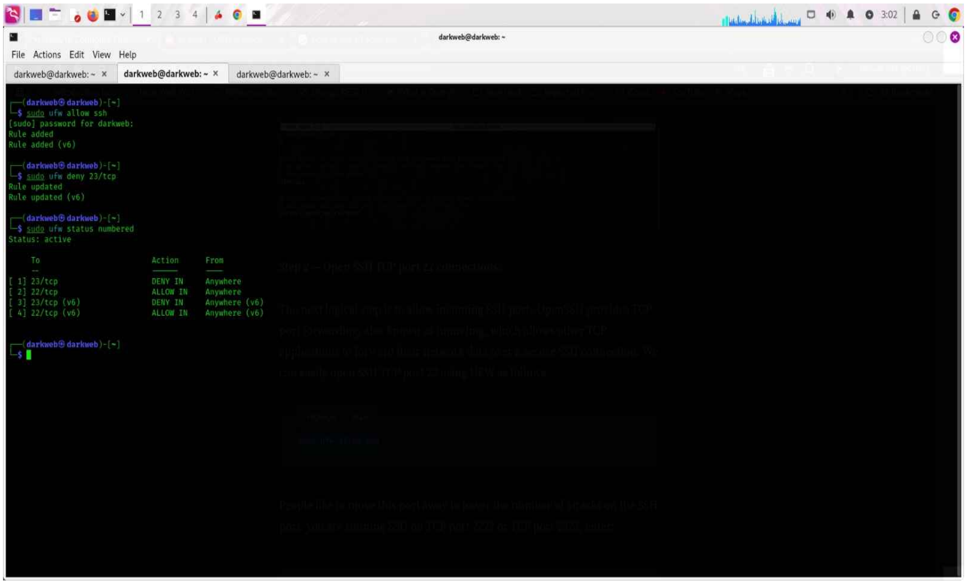
```
sudo ufw status numbered
```



```
┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw status numbered
Status: active

┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw deny 23/tcp
Rule added
Rule added (v6)

┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 23/tcp                     DENY IN     Anywhere
[ 2] 23/tcp (v6)                DENY IN     Anywhere (v6)


┌──(darkweb㉿darkweb)-[~]
└─$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ufw-before-logging-input  all  --  anywhere             anywhere
ufw-before-input  all  --  anywhere             anywhere
ufw-after-input  all  --  anywhere             anywhere
ufw-after-logging-input  all  --  anywhere             anywhere
ufw-reject-input  all  --  anywhere             anywhere
ufw-track-input  all  --  anywhere             anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination
DOCKER-USER  all  --  anywhere             anywhere
DOCKER-ISOLATION-STAGE-1  all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             ctstate RELATED,ESTABLISHED
DOCKER     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ufw-before-logging-forward  all  --  anywhere             anywhere
ufw-before-forward  all  --  anywhere             anywhere
ufw-after-forward  all  --  anywhere             anywhere
ufw-after-logging-forward  all  --  anywhere             anywhere
ufw-reject-forward  all  --  anywhere             anywhere
ufw-track-forward  all  --  anywhere             anywhere
```

Step 5 – Test the Rule

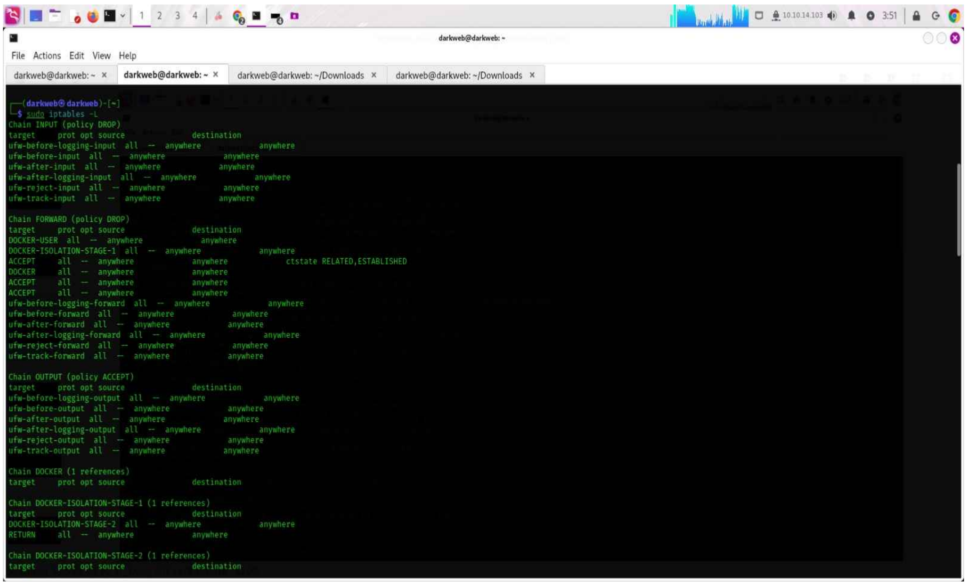
Command:
`nc -v 127.0.0.1 23` or `telnet 127.0.0.1 23`



```
┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw allow ssh
[sudo] password for darkweb:
Rule added
Rule added (v6)

┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw deny 23/tcp
Rule updated
Rule updated (v6)

┌──(darkweb㉿darkweb)-[~]
└─$ sudo ufw status numbered
Status: active

     To                         Action      From
     --                         ------      ----
[ 1] 23/tcp                     DENY IN     Anywhere
[ 2] 22/tcp                     ALLOW IN    Anywhere
[ 3] 23/tcp (v6)                DENY IN     Anywhere (v6)
[ 4] 22/tcp (v6)                ALLOW IN    Anywhere (v6)


┌──(darkweb㉿darkweb)-[~]
└─$
```

Step 6 – Remove Test Rule

Command:
`sudo ufw delete deny 23/tcp`



darkweb@darkweb: ~
File Actions Edit View Help
darkweb@darkweb: ~
darkweb@darkweb: ~
darkweb@darkweb: ~/Downloads
darkweb@darkweb: ~/Downloads

```
┌──(darkweb㉿darkweb)-[~]
└─$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source               destination
ufw-before-logging-input  all  --  anywhere             anywhere
ufw-before-input  all  --  anywhere             anywhere
ufw-after-input  all  --  anywhere             anywhere
ufw-after-logging-input  all  --  anywhere             anywhere
ufw-reject-input  all  --  anywhere             anywhere
ufw-track-input  all  --  anywhere             anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination
DOCKER-USER  all  --  anywhere             anywhere
DOCKER-ISOLATION-STAGE-1  all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere             ctstate RELATED,ESTABLISHED
DOCKER     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere
ufw-before-logging-forward  all  --  anywhere             anywhere
ufw-before-forward  all  --  anywhere             anywhere
ufw-after-forward  all  --  anywhere             anywhere
ufw-after-logging-forward  all  --  anywhere             anywhere
ufw-reject-forward  all  --  anywhere             anywhere
ufw-track-forward  all  --  anywhere             anywhere

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
ufw-before-logging-output  all  --  anywhere             anywhere
ufw-before-output  all  --  anywhere             anywhere
ufw-after-output  all  --  anywhere             anywhere
ufw-after-logging-output  all  --  anywhere             anywhere
ufw-reject-output  all  --  anywhere             anywhere
ufw-track-output  all  --  anywhere             anywhere

Chain DOCKER (1 references)
target     prot opt source               destination

Chain DOCKER-ISOLATION-STAGE-1 (1 references)
target     prot opt source               destination
DOCKER-ISOLATION-STAGE-2  all  --  anywhere             anywhere
RETURN     all  --  anywhere             anywhere

Chain DOCKER-ISOLATION-STAGE-2 (1 references)
target     prot opt source               destination
```

7. Summary – How Firewall Filters Traffic

A firewall acts as a barrier between a trusted internal network and untrusted external networks, such as the internet. It examines inbound and outbound data packets and allows or blocks them based on pre-configured rules. Filtering can be based on port numbers, protocols, IP addresses, and application-specific rules. In this lab, we blocked Telnet traffic on port 23, verified the block, and then restored the original configuration.