

Two–Interface Router with NAT Cisco IOS Firewall Configuration

Document ID: 5143

Contents

Introduction

Prerequisites

- Requirements
- Components Used
- Conventions

Background Information

Configure

- Network Diagram
- Configuration

Verify

Troubleshoot

- Problem
- Solution

Related Information

Introduction

This sample configuration works for a very small office connected directly to the Internet. The assumption is that Domain Name Service (DNS), Simple Mail Transfer Protocol (SMTP) and web services are provided by a remote system run by the Internet Service Provider (ISP). There are no services on the inside network, which makes this one of the simplest firewall configurations, as there are only two interfaces. There is no logging, because there is no host available to provide logging services.

Refer to Three–interface Router without NAT Cisco IOS Firewall Configuration in order to configure a three interface router without NAT using the Cisco IOS® Firewall.

Refer to Two–interface Router without NAT Using Cisco IOS Firewall Configuration in order to configure a two interface router without NAT using the Cisco IOS Firewall.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on these software and hardware versions:

- Cisco IOS Software Release 12.2
- Cisco 3640 router

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure

that you understand the potential impact of any command.

Conventions

Refer to the Cisco Technical Tips Conventions for more information on document conventions.

Background Information

Since this configuration uses only input access lists, it does both anti-spoofing and traffic filtering with the same access list (101). This configuration only works for a two-port router. Ethernet 1 is the "inside" network. Serial 0 is the outside interface. The access list (112) on Serial 0 illustrates this using the Network Address Translation (NAT) global IP addresses (150.150.150.x) as destinations.

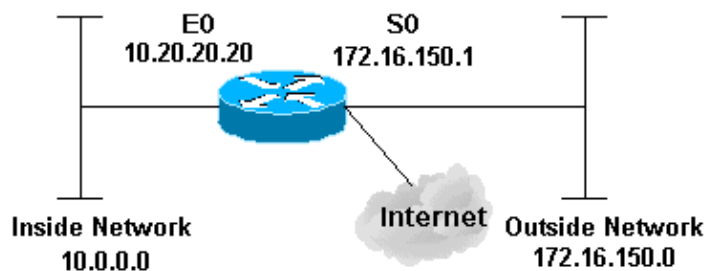
Configure

In this section, you are presented with the information to configure the features described in this document.

Note: Use the Command Lookup Tool (registered customers only) to obtain more information on the commands used in this section.

Network Diagram

This document uses this network setup.



Configuration

This document uses this configuration.

```
3640 Router
version 12.2
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
no service password-encryption
!
hostname pig
!
boot system flash flash:c3640-jk9o3s-mz.122-21a.bin
logging buffered 4096 debugging
```

```
enable secret 5 $1$chHU$wiC58FP/IDloZuorCkzEz1
enable password ww
!
clock timezone CET 1
clock summer-time CET recurring
ip subnet-zero
!
!
no ip domain-lookup
!
```

```
!--- This is the Cisco IOS Firewall
!--- configuration and what to inspect.
```

```
ip inspect name ethernetin cuseeme timeout 3600
ip inspect name ethernetin ftp timeout 3600
ip inspect name ethernetin h323 timeout 3600
ip inspect name ethernetin http timeout 3600
ip inspect name ethernetin rcmd timeout 3600
ip inspect name ethernetin realaudio timeout 3600
ip inspect name ethernetin smtp timeout 3600
ip inspect name ethernetin sqlnet timeout 3600
ip inspect name ethernetin streamworks timeout 3600
ip inspect name ethernetin tcp timeout 3600
ip inspect name ethernetin tftp timeout 30
ip inspect name ethernetin udp timeout 15
ip inspect name ethernetin vdolive timeout 3600
ip audit notify log
ip audit po max-events 100
!
call rsvp-sync
!
!
!
!
!
!
!
```

```
!--- This is the inside of the network.
```

```
interface Ethernet0/0
 ip address 10.20.20.20 255.255.255.0
 ip access-group 101 in
 ip nat inside
 ip inspect ethernetin in
 half-duplex
!
interface Ethernet0/1
 no ip address
 shutdown
 half-duplex
!
interface Serial1/0
 no ip address
 shutdown
!
interface Serial1/1
 no ip address
 shutdown
!
interface Serial1/2
 no ip address
 shutdown
!
```

```

!--- This is the outside of the interface.

interface Serial1/3
 ip address 172.16.150.1 255.255.255.0
 ip access-group 112 in
 ip nat outside
!

!--- Define the NAT pool.

ip nat pool mypool 172.16.150.3 172.16.150.255 netmask 255.255.255.0
ip nat inside source list 1 pool mypool
ip classless
ip route 0.0.0.0 0.0.0.0 172.16.150.2
ip http server
!
access-list 1 permit 10.0.0.0 0.255.255.255

!--- Access list applied on the inside for anti-spoofing reasons.

access-list 101 permit tcp 10.0.0.0 0.255.255.255 any
access-list 101 permit udp 10.0.0.0 0.255.255.255 any
access-list 101 permit icmp 10.0.0.0 0.255.255.255 any
access-list 101 deny ip any any log

!--- Access list applied on the outside for security reasons.

access-list 112 permit icmp any 172.16.150.0 0.0.0.255 unreachable
access-list 112 permit icmp any 150.150.150.0 0.0.0.255 echo-reply
access-list 112 permit icmp any 172.16.150.0 0.0.0.255 packet-too-big
access-list 112 permit icmp any 172.16.150.0 0.0.0.255 time-exceeded
access-list 112 permit icmp any 172.16.150.0 0.0.0.255 traceroute
access-list 112 permit icmp any 172.16.150.0 0.0.0.255 administratively-prohibited
access-list 112 permit icmp any 172.16.150.0 0.0.0.255 echo
access-list 112 deny ip any any log
!
!
dial-peer cor custom
!
!
!
!
!
line con 0
 exec-timeout 0 0
line 97 102
line aux 0
line vty 0 4
 exec-timeout 0 0
 password ww
 login
!
end

```

Verify

Use this section to confirm that your configuration works properly.

The Output Interpreter Tool (registered customers only) (OIT) supports certain **show** commands. Use the OIT to view an analysis of **show** command output.

- **show version** Displays information about the currently loaded software version along with hardware and device information.
- **debug ip nat** Displays information about IP packets translated by the IP NAT feature.
- **show ip nat translations** Displays active NATs.
- **show log** Displays logging information.
- **show ip access-list** Displays the contents of all current IP access lists.
- **show ip inspect session** Displays existing sessions that are currently tracked and inspected by the Cisco IOS Firewall.
- **debug ip inspect tcp** Displays messages about Cisco IOS Firewall events.

This is sample command output from the **show version** command.

```

pig#show version
Cisco Internetwork Operating System Software
IOS (tm) 3600 Software (C3640-JK9O3S-M), Version 12.2(21a), RELEASE SOFTWARE (fc2)
Copyright (c) 1986-2004 by cisco Systems, Inc.
Compiled Fri 09-Jan-04 16:23 by kellmill
Image text-base: 0x60008930, data-base: 0x615DE000

ROM: System Bootstrap, Version 11.1(19)AA, EARLY DEPLOYMENT RELEASE SOFTWARE (fc1)

pig uptime is 59 minutes
System returned to ROM by reload at 16:05:44 CET Wed Jan 14 2004
System image file is "flash:c3640-jk9o3s-mz.122-21a.bin"

```

This product contains cryptographic features and is subject to United States and local country laws governing import, export, transfer and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at: <http://www.cisco.com/wwl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to export@cisco.com.

```

cisco 3640 (R4700) processor (revision 0x00) with 126976K/4096K bytes of memory.
Processor board ID 10577176
R4700 CPU at 100Mhz, Implementation 33, Rev 1.0
MICA-6DM Firmware: CP ver 2730 - 5/23/2001, SP ver 2730 - 5/23/2001.
Bridging software.
X.25 software, Version 3.0.0.
SuperLAT software (copyright 1990 by Meridian Technology Corp).
TN3270 Emulation software.
2 Ethernet/IEEE 802.3 interface(s)
4 Low-speed serial(sync/async) network interface(s)
6 terminal line(s)
1 Virtual Private Network (VPN) Module(s)
DRAM configuration is 64 bits wide with parity disabled.
125K bytes of non-volatile configuration memory.
32768K bytes of processor board System flash (Read/Write)

```

First, verify NAT works correctly using **debug ip nat** and **show ip nat translations** as shown in this output.

```

pig#debug ip nat
IP NAT debugging is on
pig#
*Mar  1 01:40:47.692 CET: NAT: s=10.0.0.1->172.16.150.4, d=172.16.150.2 [80]

```

```

*Mar 1 01:40:47.720 CET: NAT*: s=172.16.150.2, d=172.16.150.4->10.0.0.1 [80]
*Mar 1 01:40:47.720 CET: NAT*: s=10.0.0.1->172.16.150.4, d=172.16.150.2 [81]
*Mar 1 01:40:47.748 CET: NAT*: s=172.16.150.2, d=172.16.150.4->10.0.0.1 [81]
*Mar 1 01:40:47.748 CET: NAT*: s=10.0.0.1->172.16.150.4, d=172.16.150.2 [82]
*Mar 1 01:40:47.784 CET: NAT*: s=172.16.150.2, d=172.16.150.4->10.0.0.1 [82]
*Mar 1 01:40:47.784 CET: NAT*: s=10.0.0.1->172.16.150.4, d=172.16.150.2 [83]
*Mar 1 01:40:47.836 CET: NAT*: s=172.16.150.2, d=172.16.150.4->10.0.0.1 [83]
*Mar 1 01:40:47.836 CET: NAT*: s=10.0.0.1->172.16.150.4, d=172.16.150.2 [84]
*Mar 1 01:40:47.884 CET: NAT*: s=172.16.150.2, d=172.16.150.4->10.0.0.1 [84]

```

```
pig#show ip nat translations
```

```

Pro Inside global      Inside local      Outside local      Outside global
--- 172.16.150.4        10.0.0.1          ---                ---

```

Without adding the **ip inspect** statement, confirm that the access lists work correctly. The **deny ip any any** with the **log** keyword tells you what packets are blocked.

In this case, this is the return traffic from a Telnet session to 172.16.150.2 from 10.0.0.1 (translated to 172.16.150.4).

This is sample output of the **show log** command.

```
pig#show log
```

```
Syslog logging: enabled (0 messages dropped, 0 messages rate-limited,
0 flushes, 0 overruns)
```

```
  Console logging: level debugging, 92 messages logged
```

```
  Monitor logging: level debugging, 0 messages logged
```

```
  Buffer logging: level debugging, 60 messages logged
```

```
  Logging Exception size (4096 bytes)
```

```
  Trap logging: level informational, 49 message lines logged
```

```
Log Buffer (4096 bytes):
```

```
*Mar 1 01:24:08.518 CET: %SYS-5-CONFIG_I: Configured from console by console
```

```
*Mar 1 01:26:47.783 CET: %SYS-5-CONFIG_I: Configured from console by console
```

```
*Mar 1 01:27:09.876 CET: %SEC-6-IPACCESSLOGP: list 112 denied tcp 172.16.150.2(23)
-> 172.16.150.4(11004), 1 packet
```

```
*Mar 1 01:33:03.371 CET: %SEC-6-IPACCESSLOGP: list 112 denied tcp 172.16.150.2(23)
-> 172.16.150.4(11004), 3 packets
```

Use the **show ip access-lists** command in order to see how many packets match the access list.

```
pig#show ip access-lists
```

```
Standard IP access list 1
```

```
  permit 10.0.0.0, wildcard bits 0.255.255.255 (28 matches)
```

```
Extended IP access list 101
```

```
  permit tcp 10.0.0.0 0.255.255.255 any (32 matches)
```

```
  permit udp 10.0.0.0 0.255.255.255 any
```

```
  permit icmp 10.0.0.0 0.255.255.255 any (22 matches)
```

```
  deny ip any any log
```

```
Extended IP access list 112
```

```
  permit icmp any 172.16.150.0 0.0.0.255 unreachable
```

```
  permit icmp any 172.16.150.0 0.0.0.255 echo-reply (10 matches)
```

```
  permit icmp any 172.16.150.0 0.0.0.255 packet-too-big
```

```
  permit icmp any 172.16.150.0 0.0.0.255 time-exceeded
```

```
  permit icmp any 172.16.150.0 0.0.0.255 traceroute
```

```
  permit icmp any 172.16.150.0 0.0.0.255 administratively-prohibited
```

```
  permit icmp any 172.16.150.0 0.0.0.255 echo
```

```
  deny ip any any log (12 matches)
```

```
pig#
```

Once you have added the **ip inspect** statement, you can see that this line has dynamically been added in the access list in order to allow this Telnet session:

```

permit tcp host 172.16.150.2 eq telnet host 172.16.150.4 eq 11004 (16 matches)

pig#show ip access-lists
Standard IP access list 1
  permit 10.0.0.0, wildcard bits 0.255.255.255 (44 matches)
Extended IP access list 101
  permit tcp 10.0.0.0 0.255.255.255 any (50 matches)
  permit udp 10.0.0.0 0.255.255.255 any
  permit icmp 10.0.0.0 0.255.255.255 any (22 matches)
  deny ip any any log
Extended IP access list 112
  permit tcp host 172.16.150.2 eq telnet host 172.16.150.4 eq 11004 (16 matches)
  permit icmp any 172.16.150.0 0.0.0.255 unreachable
  permit icmp any 172.16.150.0 0.0.0.255 echo-reply (10 matches)
  permit icmp any 172.16.150.0 0.0.0.255 packet-too-big
  permit icmp any 172.16.150.0 0.0.0.255 time-exceeded
  permit icmp any 172.16.150.0 0.0.0.255 traceroute
  permit icmp any 172.16.150.0 0.0.0.255 administratively-prohibited
  permit icmp any 172.16.150.0 0.0.0.255 echo
  deny ip any any log (12 matches)
pig#

```

You can also check using the **show ip inspect session** command which shows the current sessions that have been established through the firewall.

```

pig#show ip inspect session
Established Sessions
  Session 624C31A4 (10.0.0.1:11006)=>(172.16.150.2:23) tcp SIS_OPEN

```

Eventually, at a more advanced level, you can also enable the **debug ip inspect tcp** command.

```

pig#debug ip inspect tcp
INSPECT TCP Inspection debugging is on
pig#
*Mar  1 01:49:51.756 CET: CBAC sis 624C31A4 pak 624D0FA8 TCP S
      seq 2890060460(0) (172.16.150.4:11006) => (172.16.150.2:23)
*Mar  1 01:49:51.776 CET: CBAC sis 624C31A4 pak 624D0CC4 TCP S
      ack 2890060461 seq 1393191461(0) (10.0.0.1:11006) <= (172.16.150.2:23)
*Mar  1 01:49:51.776 CET: CBAC* sis 624C31A4 pak 62576284 TCP
      ack 1393191462 seq 2890060461(0) (172.16.150.4:11006) => (172.16.150.2:23)
*Mar  1 01:49:51.776 CET: CBAC* sis 624C31A4 pak 62576284 TCP P ack
      1393191462 seq 2890060461(12) (172.16.150.4:11006) => (172.16.150.2:23)
*Mar  1 01:49:51.780 CET: CBAC* sis 624C31A4 pak 62576284 TCP ack
      1393191462 seq 2890060473(0) (172.16.150.4:11006) => (172.16.150.2:23)

```

Troubleshoot

After you configure the IOS Firewall router, if the connections do not work, ensure that you have enabled inspection with the **ip inspect (name defined) in or out** command on the interface. In this configuration, **ip inspect ethernetin in** is applied for the interface **Ethernet0/0**.

For general troubleshooting on this configuration, refer to [Troubleshooting Cisco IOS Firewall Configurations and Troubleshooting Authentication Proxy](#).

Problem

You cannot perform http downloads because it fails or is timed out. How is this resolved?

Solution

The issue can be resolved by removing **ip inspect** for http traffic so that the http traffic is not inspected and the download occurs as expected.

Related Information

- [IOS Firewall Support Page](#)
 - [IOS Firewall in IOS Documentation](#)
 - [Technical Support & Documentation – Cisco Systems](#)
-

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2014 – 2015 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Feb 20, 2007

Document ID: 5143
