ASA Release 9.(x) Connection of Three Internal Networks with Internet Configuration Example

Contents

Introduction

This document provides information on how to set up the Cisco Adaptive Security Appliance (ASA) Version 9.1(5) for use with three internal networks. Static routes are used on the routers for simplicity.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on the Cisco Adaptive Security Appliance (ASA) Version 9.1(5).

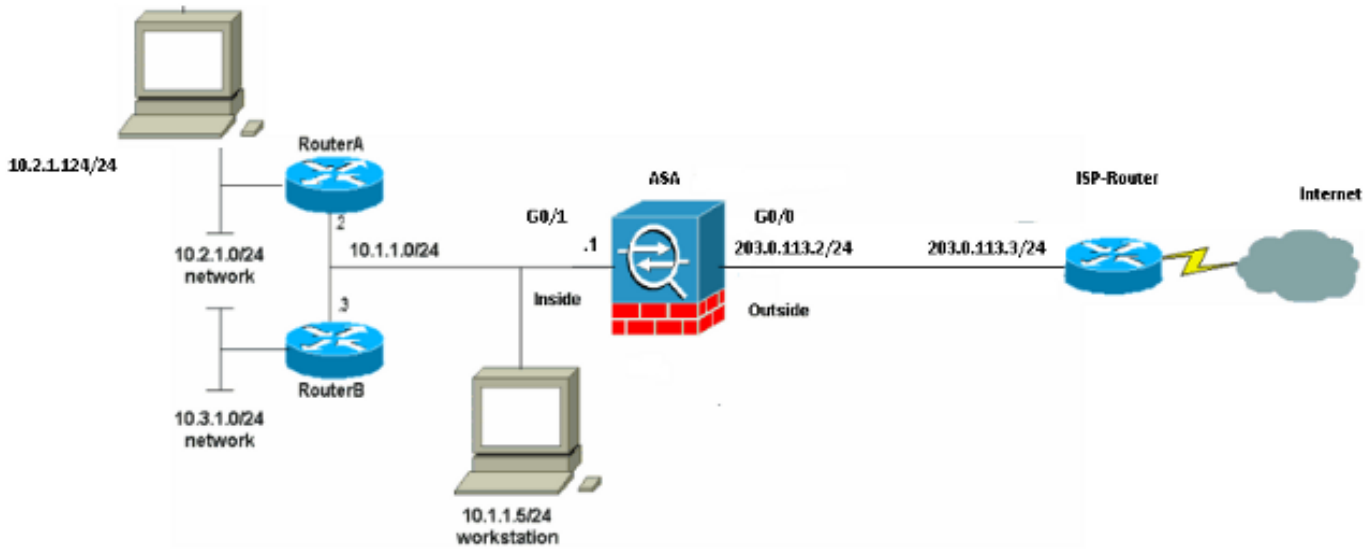
The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

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) in order to obtain more information on the commands used in this section.

Network Diagram



> **Note**: The IP addressing schemes used in this configuration are not legally routable on the Internet. They are RFC 1918 addresses that have been used in a lab environment.

ASA 9.1 Configuration

This document uses these configurations. If you have the output of a **write terminal** command from your Cisco device, you can use Output Interpreter (registered customers only) to display potential issues and fixes.

Configurations

- Router A Configuration
- Router B Configuration
- ASA Revision 9.1 and Later Configuration

Router A Configuration

```
RouterA#show running-config
Building configuration...

Current configuration : 1151 bytes
!
version 12.4
```

```
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname RouterA
!
boot-start-marker
boot-end-marker
!
enable password cisco
!
memory-size iomem 25
no network-clock-participate slot 1
no network-clock-participate wic 0
no network-clock-participate wic 1
no network-clock-participate wic 2
no network-clock-participate aim 0
no network-clock-participate aim 1
no aaa new-model
ip subnet-zero
ip cef
!
!
!
!
ip audit po max-events 100
no ftp-server write-enable
!
!
!
!
!
no crypto isakmp enable
!
!
!
interface FastEthernet0/0
ip address 10.1.1.2 255.255.255.0
duplex auto
speed auto
!
interface FastEthernet0/1
ip address 10.2.1.1 255.255.255.0
duplex auto
speed auto
!
interface IDS-Sensor1/0
no ip address
shutdown
hold-queue 60 out
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.1.1.1
ip route 10.3.1.0 255.255.255.0 10.1.1.3
no ip http server
no ip http secure-server
!
!
!
!
!
control-plane
!
!
```

```
!
line con 0
line 33
no activation-character
no exec
transport preferred none
transport input all
transport output all
line aux 0
line vty 0 4
password ww
login
!
!
end

RouterA#
```

Router B Configuration

```
RouterB#show running-config
Building configuration...

Current configuration : 1132 bytes
!
version 12.4
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname RouterB
!
boot-start-marker
boot-end-marker
!
!
no network-clock-participate slot 1
no network-clock-participate wic 0
no network-clock-participate wic 1
no network-clock-participate wic 2
no network-clock-participate aim 0
no network-clock-participate aim 1
no aaa new-model
ip subnet-zero
ip cef
!
!
!
!
ip audit po max-events 100
no ip domain lookup
no ftp-server write-enable
!
!
!
!
!
no crypto isakmp enable
!
!
!
interface FastEthernet0/0
ip address 10.1.1.3 255.255.255.0
```

```
duplex auto
speed auto
no cdp enable
!
interface FastEthernet0/1
ip address 10.3.1.1 255.255.255.0
duplex auto
speed auto
!
interface IDS-Sensor1/0
no ip address
shutdown
hold-queue 60 out
!
ip classless
ip route 0.0.0.0 0.0.0.0 10.1.1.2
no ip http server
no ip http secure-server
!
!
!
!
!
control-plane
!
!
!
line con 0
stopbits 1
line 33
no activation-character
no exec
transport preferred none
transport input all
transport output all
line aux 0
line vty 0 4
password cisco
login
!
!
end

RouterB#
```

ASA Revision 9.1 and Later Configuration

```
ASA#show run
: Saved
:
ASA Version 9.1(5)
!
hostname ASA
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
names
!
interface GigabitEthernet0/0
nameif outside
security-level 0
ip address 203.0.113.2 255.255.255.0
!
interface GigabitEthernet0/1
nameif inside
```

```
security-level 100
ip address 10.1.1.1 255.255.255.0
!
boot system disk0:/asa915-k8.bin

ftp mode passive

!--- Enable informational logging to see connection creation events

logging on
logging buffered informational

!--- Output Suppressed

!--- Creates an object called OBJ_GENERIC_ALL.
!--- Any host IP not already matching another configured
!--- object will get PAT to the outside interface IP
!--- on the ASA (or 10.165.200.226) for internet bound traffic.


object network OBJ_GENERIC_ALL
subnet 0.0.0.0 0.0.0.0
nat (inside,outside) source dynamic OBJ_GENERIC_ALL interface

!--- Output Suppressed

!--- Define a default route to the ISP router.


route outside 0.0.0.0 0.0.0.0 203.0.113.3 1

!--- Define a route to the INTERNAL router with network 10.2.1.0.

route inside 10.2.1.0 255.255.255.0 10.1.1.2 1

!--- Define a route to the INTERNAL router with network 10.3.1.0.

route inside 10.3.1.0 255.255.255.0 10.1.1.3 1

: end
```

Verify

Use this section to confirm that your configuration works properly.

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

Try to access a web site via HTTP with a web browser. This example uses a site that is hosted at 198.51.100.100. If the connection is successful, this output can be seen on the ASA CLI.

Connection

```
ASA(config)# show connection address 10.2.1.124
16 in use, 918 most used
TCP outside 198.51.100.100:80 inside 10.2.1.124:18711, idle 0:00:16, bytes 1937,
flags UIO
```

The ASA is a stateful firewall, and return traffic from the web server is allowed back through the firewall because it matches a ***connection*** in the firewall connection table. Traffic that matches a connection that pre-exists is allowed through the firewall and is not blocked by an interface ACL.

In the previous output, the client on the inside interface has established a connection to the 198.51.100.100 host off of the outside interface. This connection is made with the TCP protocol and has been idle for six seconds. The connection flags indicate the current state of this connection. More information about connection flags can be found in ASA TCP Connection Flags.

Syslog

```
ASA(config)# show log | include 10.2.1.124

Apr 27 2014 11:31:23: %ASA-6-305011: Built dynamic TCP translation from inside:
10.2.1.124/18711 to outside:203.0.113.2/18711

Apr 27 2014 11:31:23: %ASA-6-302013: Built outbound TCP connection 2921 for outside:
198.51.100.100/80 (198.51.100.100/80) to inside:10.2.1.124/18711 (203.0.113.2/18711)
```

The ASA Firewall generates syslogs during normal operation. The syslogs range in verbosity based on the logging configuration. The output shows two syslogs that are seen at level six, or 'informational' level.

In this example, there are two syslogs generated. The first is a log message that indicates that the firewall has built a translation, specifically a dynamic TCP translation (PAT). It indicates the source IP address and port and the translated IP address and port as the traffic traverses from the inside to the outside interfaces.

The second syslog indicates that the firewall has built a connection in its connection table for this specific traffic between the client and server. If the firewall was configured in order to block this connection attempt, or some other factor inhibited the creation of this connection (resource constraints or a possible misconfiguration), the firewall would not generate a log that indicates that the connection was built. Instead it would log a reason for the connection to be denied or an indication about what factor inhibited the connection from being created.

NAT Translations

```
ASA(config)# show xlate local 10.2.1.124
2 in use, 180 most used
Flags: D - DNS, e - extended, I - identity, i - dynamic, r - portmap,
s - static, T - twice, N - net-to-net
TCP PAT from inside:10.2.1.124/18711 to outside:203.0.113.2/18711 flags ri idle
0:12:03 timeout 0:00:30
```

As part of this configuration, PAT is configured in order to translate the internal host IP addresses to addresses that are routable on the Internet. In order to confirm that these translations are created, you can check the NAT translations (xlate) table. The command **show xlate,** when combined with the **local** keyword and the internal host's IP address, shows all of the entries present in the translation table for that host. The previous output shows that there is a translation currently built for this host between the inside and outside interfaces. The inside host IP and port are translated to the 203.0.113.2 address per our configuration. The flags listed, r i ,indicate that the translation is **dynamic** and a **portmap**. More information about different NAT configurations can be found in Information About NAT.

Troubleshoot

This section provides information you can use to troubleshoot your configuration.

The ASA provides multiple tools with which to troubleshoot connectivity. If the issue persists after

you verify the configuration and check the output listed previously, these tools and techniques might help determine the cause of your connectivity failure.

Packet Tracer

```
ASA(config)# packet-tracer input inside tcp 10.2.1.124 1234 198.51.100.100 80


--Omitted--

Result:
input-interface: inside
input-status: up
input-line-status: up
output-interface: outside
output-status: up
output-line-status: up
Action: allow
```

The packet tracer functionality on the ASA allows you to specify a simulated packet and see all of the various steps, checks, and functions that the firewall goes through when it processes traffic. With this tool, it is helpful to identify an example of traffic you believe should be allowed to pass through the firewall, and use that 5-tupple in order to simulate traffic. In the previous example, the packet tracer is used in order to simulate a connection attempt that meets these criteria:

- The simulated packet arrives on the **inside**.
- The protocol used is **TCP**.
- The simulated client IP address is **10.2.1.124**.
- The client sends traffic sourced from port **1234**.
- The traffic is destined to a server at IP address **198.51.100.100**.
- The traffic is destined to port **80**.

Notice that there was no mention of the interface **outside** in the command. This is by packet tracer design. The tool tells you how the firewall processes that type of connection attempt, which includes how it would route it, and out of which interface. More information about packet tracer can be found in Tracing Packets with Packet Tracer.

Capture

```
ASA# capture capin interface inside match tcp host 10.2.1.124 host 198.51.100.100
ASA# capture capout interface outside match tcp any host 198.51.100.100

ASA#   show capture capin

3 packets captured

  1: 11:31:23.432655       10.2.1.124.18711 > 198.51.100.100.80: S 780523448:
780523448(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>
  2: 11:31:23.712518       198.51.100.100.80 > 10.2.1.124.18711: S 2123396067:
2123396067(0) ack 780523449 win 8192 <mss 1024,nop,nop,sackOK,nop,wscale 8>
  3: 11:31:23.712884       10.2.1.124.18711 > 198.51.100.100.80: . ack 2123396068
win 32768



ASA# show capture capout

3 packets captured
```

```
  1: 11:31:23.432869       203.0.113.2.18711 > 198.51.100.100.80: S 1633080465:
1633080465(0) win 8192 <mss 1380,nop,wscale 2,nop,nop,sackOK>
  2: 11:31:23.712472       198.51.100.100.80 > 203.0.113.2.18711: S 95714629:
95714629(0) ack 1633080466 win 8192 <mss 1024,nop,nop,sackOK,nop,wscale 8>
  3: 11:31:23.712914       203.0.113.2.18711 > 198.51.100.100.80: . ack 95714630
win 32768/pre>
```

The ASA firewall can capture traffic that enters or leaves its interfaces. This capture functionality is fantastic because it can definitively prove if traffic arrives at, or leaves from, a firewall. The previous example showed the configuration of two captures named **capin** and **capout** on the inside and outside interfaces respectively. The capture commands used the **match** keyword, which allows you to be specific about what traffic you want to capture.

For the capture **capin,** it was indicated that you wanted to match traffic seen on the inside interface (ingress or egress) that matches **tcp host 10.2.1.124 host 198.51.100.100**. In other words, you want to capture any TCP traffic that is sent from **host 10.2.1.124** to **host 198.51.100.100** or **vice versa**. The use of the **match** keyword allows the firewall to capture that traffic bidirectionally. The capture command defined for the outside interface does not reference the internal client IP address because the firewall conducts PAT on that client IP address. As a result, you cannot **match** with that client IP address. Instead, this example uses **any** in order to indicate that all possible IP addresses would match that condition.

After you configure the captures, you would then attempt to establish a connection again, and proceed to view the captures with the **show capture <*capture_name*>** command. In this example, you can see that the client was able to connect to the server as evident by the TCP 3-Way handshake seen in the captures.