This chapter covers the following topics:

- Architectural overview
- Configuration of security contexts
- Deployment scenarios
- Monitoring and troubleshooting

CHAPTER 9

Security Contexts

The virtual firewall methodology enables a physical firewall to be partitioned into multiple standalone firewalls. Each standalone firewall acts and behaves as an independent entity with its own configuration, interfaces, security policies, routing table, and administrators. In Cisco ASA, these virtual firewalls are known as *security contexts*.

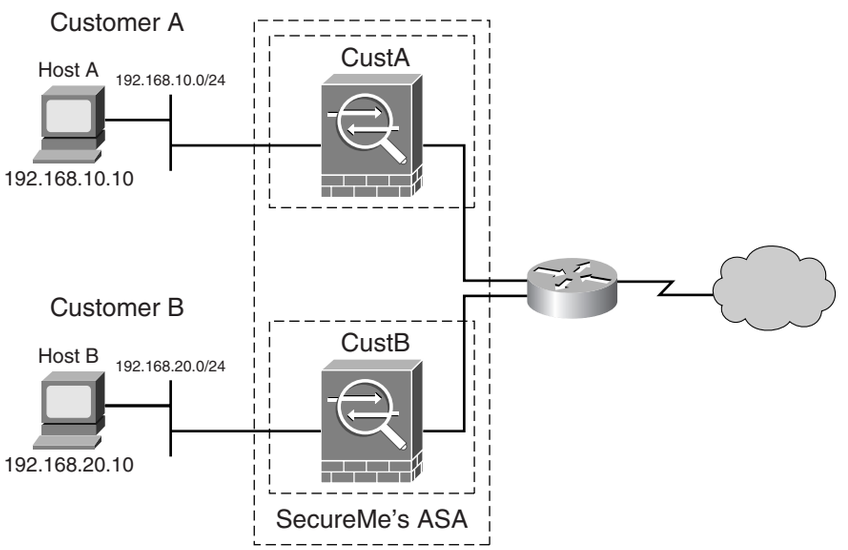
The following are some example scenarios in which security contexts are useful in network deployments:

- You act as a service provider and you want to provide firewall services to customers. However, you do not want to purchase additional physical firewalls for each customer.
- You manage an educational institution and you want to segregate student networks from faculty networks for improved security using one physical security appliance.
- You administer a large enterprise with different departmental groups, and each department wants to implement its own security policies.
- You have overlapping networks in your organization and you want to provide firewall services to all of those networks without changing the addressing scheme.
- You currently manage many physical firewalls and you want to integrate security policies into one physical firewall.

In Figure 9-1, SecureMe, an enterprise headquartered in Chicago, has a Cisco ASA providing firewall services to two of its customers. To implement a cost-effective solution, SecureMe has configured two security contexts in the security appliance: CustA for Customer A and CustB for Customer B. Each customer can manage and administer its own security context without interfering with the other context. On the other hand, the security appliance administrator manages the system execution space, which is discussed in the next section.

In this figure, each horizontal dotted box represents a security context that has a Cisco ASA inspecting and protecting the packets going through it, while the vertical box represents the physical Cisco security appliance with multiple security contexts.

Figure 9-1 *Security Contexts in the ASA*



Architectural Overview

In multiple security context mode, the Cisco security appliance can be divided into three types:

- A system execution space
- An admin context
- One or more customer contexts

All contexts must be configured correctly for proper function. Similar to a real network, in which one misconfigured device can affect the operations of other network devices, misconfiguration of a security context can impact the overall operation of a security appliance.

System Execution Space

Unlike other contexts, the system execution space does not have any Layer 2 or Layer 3 interfaces or any network settings. Rather, it is mainly used to define the attributes of other security context attributes. Here are the three important attributes configured for each context in the system execution space:

- Context name.
- Location of context's startup configuration. The configuration of each context is also known as a configlet.
- Interface allocation.

Additionally, many optional features, such as interface and boot parameters, can be configured within the system execution space. Table 9-1 lists the important features that can be set up through the system execution space.

Table 9-1 *Options Available in System Execution Space*

Feature	Description
Interface	Sets up physical interfaces for speed and duplex. Interfaces can be enabled or disabled.
Banner	Specifies a login or session banner when connecting to the security appliance.
Boot	Specifies boot parameters to load proper image.
Activation key	Enables or disables security appliance features.
File management	Adds or deletes the security context configurations that are stored locally on the security appliance.
Firewall mode	Configures single- or multiple-mode firewall in the system execution space.
Failover	Sets the failover parameters to accommodate multiple physical security appliances.

The system execution space configuration resides in the nonvolatile random-access memory (NVRAM) area of the security appliance, while the configurations for security contexts are stored either in local Flash memory or on a network storage server using one of the following protocols:

- TFTP
- FTP
- HTTPS
- HTTP

The system execution space designates one of the security contexts as the admin context, which is responsible for providing network access when the system needs to contact resources. The admin context is discussed next.

Admin Context

The admin context provides connectivity to network resources, as mentioned earlier. The IP addresses on the allocated interfaces can be used for remote management purposes, such as SSH or Telnet. The security appliance also uses the IP addresses to retrieve configurations for other contexts if they are located on a network share. A system administrator with access to the admin context can switch into the other contexts to manage them. The security appliance uses the admin context to send the syslog messages that relate to the system.

The admin context must be created before defining other contexts. Additionally, it must reside on the local disk. A new admin context can be designated at any time by using the **admin-context** command, which is discussed in the "Configuration of Security Context" section, later in this chapter.

When a Cisco ASA is converted from single mode to multi-mode, the network-related configuration of the single-mode security appliance is saved as the admin context. The security appliance names this context as, admin.

NOTE Changing the name of the admin context from admin is not recommended.

The admin context configuration is similar to a customer context. Aside from its relationship to the system execution space, it can be used as a regular context. However, using it as a regular context is not recommended, because of its significance.

Customer Context

Each customer context acts as a virtual firewall with its own configuration that contains almost all the options that are available in a standalone firewall. Table 9-2 lists the differences between a security appliance running in single mode and an appliance running in multiple mode.

Table 9-2 *Contrasting Single- and Multiple-Mode Firewalls*

Feature	Single Mode	Multiple Mode
Interface	All physical interfaces are available for use.	Only allocated interfaces are available in the contexts.
File management	Allows an administrator to copy system images and configurations.	Restricts a context administrator to manage the context configurations.
Firewall management	Allows a system administrator to fully manage the security appliance.	Allows a context administrator to manage the context.
Addressing scheme	Does not allow overlapping networks.	Allows overlapping networks between the contexts.
Routing protocols	Supports RIP and OSPF as the dynamic routing protocols.	Does not allow any dynamic routing protocols.
Licensing	There are no security contexts in single mode, hence no license is needed to turn on the security contexts.	Needs a license to activate more than two security contexts. The default license includes two customer security contexts and an admin context.
Resource allocation	The security appliance uses all the available resources.	The security appliance shares the system resources between the contexts.

Table 9-2 *Contrasting Single- and Multiple-Mode Firewalls (Continued)*

Feature	Single Mode	Multiple Mode
Failover	Does not allow Active/Active failover.	Allows Active/Active failover for redundancy and load-balancing.
Quality of service	Supports QoS.	Does not support QoS.
Multicast	Supports multicast using PIM-SM.	Does not support multicast.
VPN	Supports remote access and site-to-site VPN tunnels.	Does not support VPNs.

The number of customer contexts depends on the installed activation key. To find out how many customer contexts are allowed on a security appliance, look at the security context information in **show version**, as shown in Example 9-1. In this example, the ASA can have up to five customer contexts.

Example 9-1 *Verifying the Number of Security Contexts*

```
Chicago# show version | include Security Contexts
Security Contexts            : 5
```

NOTE The number of available contexts does not include the admin context, because of its significance to the system execution space.

Packet Flow in Multiple Mode

When the packets traverse through the security appliance in multiple mode, they are classified and forwarded to the right context. The packets are then processed based on the configured security policies on a context. The packet classification and the forwarding mechanism are discussed in the following subsections.

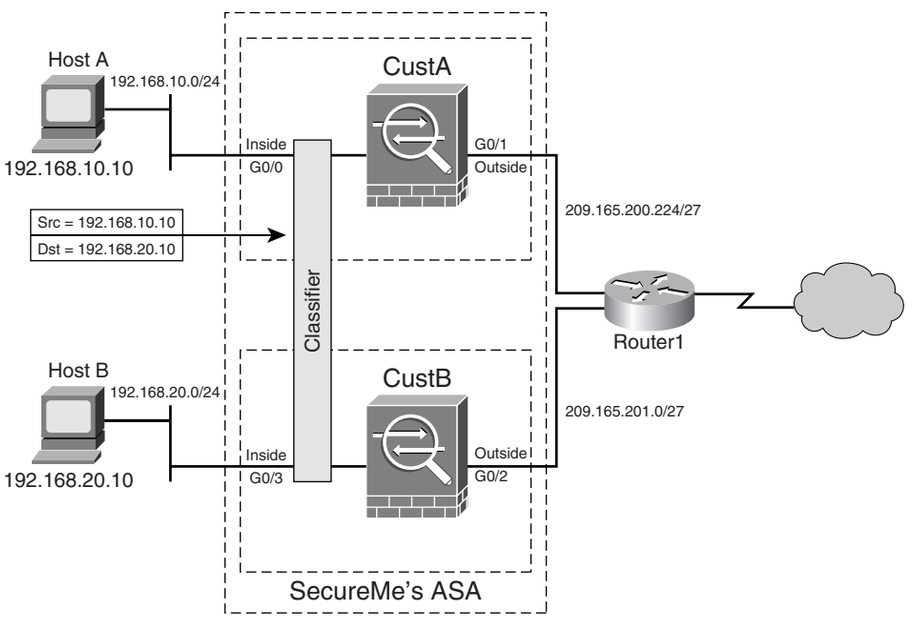
Packet Classification

In multiple mode, the security appliance must classify the packets to find out which context should operate on them. The packet classification is done at the ingress interface point that tags the packets using the source IP address, source port, destination IP address, destination port, and the interface or VLAN. The packet is processed based on the security policies configured in that context. Cisco ASA uses the following fields or packet identifiers to classify them properly:

- **Source interface**—If all the contexts in the Cisco ASA use unique interfaces, the packet classification becomes easier because the security appliance classifies these packets based on the source interface. As illustrated in Figure 9-2, when the packet is

sourced from 192.168.10.10, the classifier assigns the packet to context CustA because the packet originated from G0/0, which is a part of the CustA security context.

Figure 9-2 *Packet Classification Using Source Interface*



- **Destination IP address**—The security appliance allows you to share one or more interfaces between the security contexts. In this deployment model, the shared interface uses the same network space with unique IP addresses on the end hosts. If the security appliance is configured to use a shared ingress interface, then it uses the destination IP address to further clarify which of the security contexts using the shared interface should receive the packets. In this case, the security contexts within the Cisco ASA cannot use overlapping IP addresses, and therefore all destination IP addresses must be unique.

Packet Forwarding Between Contexts

In multiple mode, the two contexts communicate with each other as if two standalone appliances were communicating with one another. The security contexts can talk to each other in two ways:

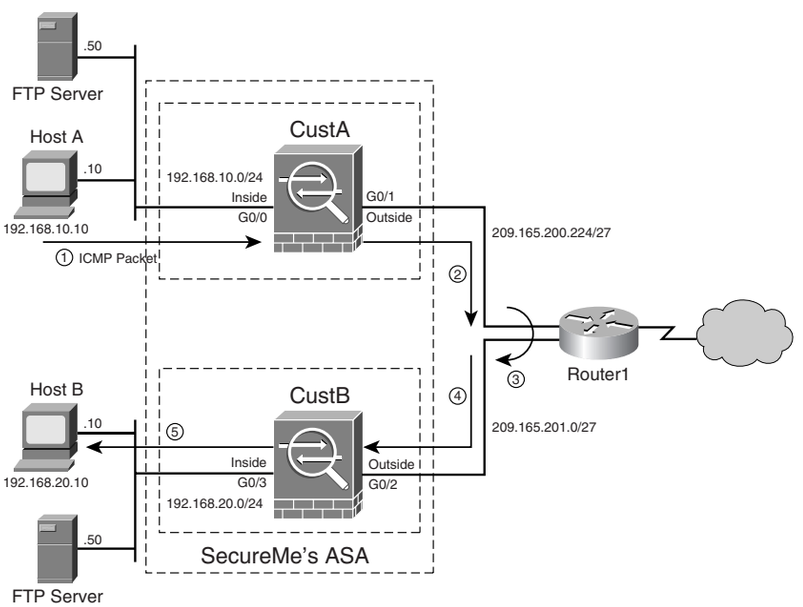
- Without a shared interface
- With a shared interface

Depending on what mode you use, the packet flow is different, as discussed in the following subsections.

Forwarding Without a Shared Interface

As Figure 9-3 illustrates, SecureMe's ASA has four interfaces: two of them belong to the CustA context and the remaining two are allocated to CustB. The outside interface of both contexts is connected to Router1, which is responsible for routing packets from one context to another.

Figure 9-3 *Security Contexts Without a Shared Interface*



If NAT and packet filtering are set up on the security appliance, then the following sequence of events takes place when Host A sends an ICMP ping packet to Host B:

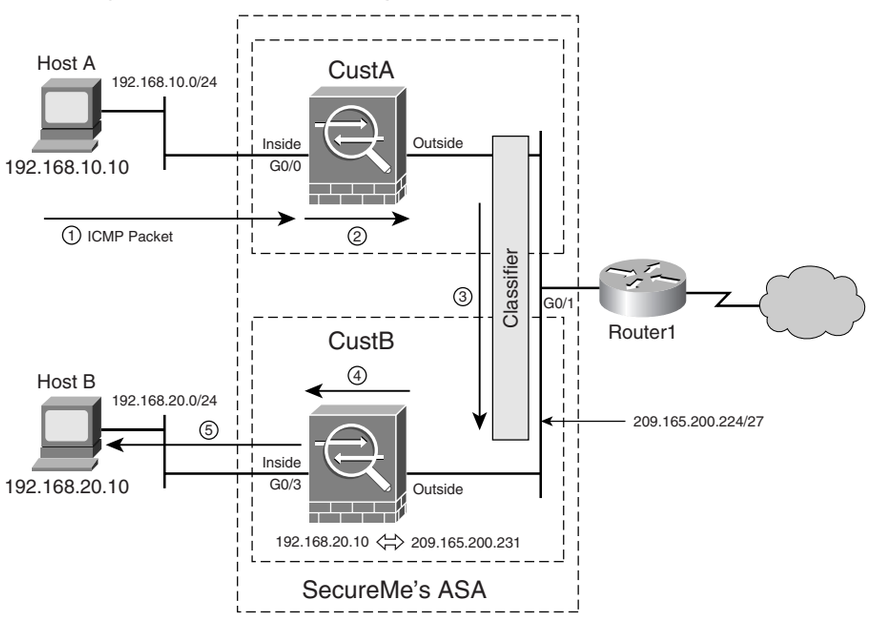
1. Host A sends an ICMP ping packet with a source address of 192.168.10.10 and a destination address of 192.168.20.10. The classifier tags the packet coming in on GigabitEthernet0/0 before sending it to the inside interface of CustA.
2. The packet is inspected by the inbound ACL and, if allowed, forwarded to the NAT engine for translation. The NAT engine translates the source address or leaves it unchanged as dictated by the configured policy. Before the security appliance forwards it to Router1, the packet is inspected by the outbound ACL to ensure that it is allowed to leave.
3. Router1 checks the destination IP address in the routing table and sends the packet to the G0/2 interface on the security appliance.
4. The appliance classifies the packet before sending it to the outside interface of the CustB context, where it is inspected by the inbound ACL. If it is allowed in, the packet passes through the NAT engine to determine if it needs to be translated.

5 The security context forwards the packet to Host B after verifying that the outbound ACL on the inside interface does not deny it.

Forwarding with a Shared Interface

Figure 9-4 illustrates another network topology, where SecureMe uses a shared outside LAN interface. To provide Internet connectivity, it has Router1 connected to the same shared interface. Using the shared interfaces, SecureMe can conserve the address space and the allocated interfaces. Additionally, shared contexts are useful when multiple security contexts need access to one public interface to get Internet connectivity.

Figure 9-4 *Security Contexts with a Shared Interface*



Using the previous example, when Host A sends an ICMP ping packet to Host B, the following steps are taken for successful communication:

1 Host A sends an ICMP ping packet with a source address of 192.168.10.10 and a destination address of 209.165.200.231, which can be translated by context CustB to 192.168.20.1. The classifier tags the packet coming in on GigabitEthernet0/0 before sending it to the inside interface of CustA.

2 The packet is inspected by the inbound ACL and, if allowed, forwarded to the NAT engine for translation. The NAT engine translates the source address or leaves it unchanged as dictated by the configured policy. The packet is then inspected by the outbound ACL to ensure that it is allowed to leave.

3 The packet passes through the context classifier, which looks at the destination IP address and forwards it to the outside interface of the CustB security context because 209.165.200.231 is owned by CustB.

> **Note** Because the security contexts reside on a physical security appliance, the packet never leaves the device when it moves from the outside interface of CustA to the outside interface of CustB.

4. The security context of CustB applies security policies after receiving the packet on the outside interface. The packet enters CustB's security context, where it is inspected by the inbound ACL. If it is allowed in, the NAT engine translates the destination address to 192.168.20.10.

> **Note** You will need to translate the destination IP addresses for the traffic traversing from a shared interface.

5. The security context forwards the packet to Host B after verifying that the outbound ACL on the inside interface does not deny it.

> **Note** Cisco ASA cannot classify packets if they are sourced from one shared interface and destined to another shared interface.

Configuration of Security Contexts

The configuration of a security context is broken down into seven steps:

1. Enable multiple security contexts globally.
2. Set up the system execution space.
3. Specify a configuration URL.
4. Allocate the interfaces.
5. Configure an admin context.
6. Configure a customer context.
7. Manage the security contexts (optional).

Refer to Figure 9-3 throughout this section to visualize how to configure a virtual firewall.

Step 1: Enabling Multiple Security Contexts Globally

The security context can be enabled by using the **mode multiple** command, as shown in Example 9-2. When this command is executed, the security appliance prompts the system administrator to verify mode conversion before proceeding further. This initiates the reboot process to complete mode conversion.

Example 9-2 *Enabling Security Context*

```
Chicago# configure terminal
Chicago(config)# mode ?
  multiple  Multiple mode; mode with security contexts
  single    Single mode; mode without security contexts
Chicago(config)# mode multiple
WARNING: This command will change the behavior of the device
WARNING: This command will initiate a Reboot
Proceed with change mode? [confirm]
Convert the system configuration? [confirm]
The old running configuration file will be written to disk0
The admin context configuration will be written to disk0
The new running configuration file was written to disk0
Security context mode: multiple
***
*** --- SHUTDOWN NOW ---
***
*** Message to all terminals:
***
***   change mode
Rebooting....
Booting system, please wait...
! Output omitted for brevity.
INFO: Context admin was created with URL disk0:/admin.cfg
INFO: Admin context will take some time to come up .... please wait.
Chicago>
```

When multiple-mode conversion is initiated, the security appliance prompts the administrator to convert the current running configuration into the system execution space and admin context. The appliance stores the system execution space in NVRAM and saves the admin context in the local Flash memory as admin.cfg. During conversion, it copies all the network-related information to the admin.cfg file, while all the device-related system information is stored in the NVRAM space.

NOTE The security appliance saves the running configuration of the single-mode firewall as old_running.cfg in the Flash memory during the conversion process.

Once the appliance comes online, you can use **show mode** to verify whether it is running in multiple mode. Example 9-3 shows the output of **show mode**.

Example 9-3 *Verifying Virtual Firewall Mode*

```
Chicago# show mode
Security context mode: multiple
```

NOTE If you do not have the license for multiple security contexts, the system key still lets you configure two customer contexts, in addition to one admin context. Refer to Chapter 4, “Initial Setup and System Maintenance,” for more information about licensing.

To convert the device back to single mode, you have to copy the saved old_running.cfg as the startup configuration. After that, you need to switch the security appliance to single mode. Both of these steps are shown in Example 9-4.

Example 9-4 *Reverting to Single-Mode Firewall*

```
Chicago# copy disk0:/old_running.cfg startup-config
Source filename [old_running.cfg]?
Copy in progress...C
1465 bytes copied in 0.250 secs
Chicago# configure terminal
Chicago(config)# mode single
WARNING: This command will change the behavior of the device
WARNING: This command will initiate a Reboot
Proceed with change mode? [confirm]
Security context mode: single ***
*** --- SHUTDOWN NOW ---
***
*** Message to all terminals:
***
***   change mode
Rebooting....
```

Step 2: Setting Up the System Execution Space

As mentioned earlier, the system execution space is created as soon as multiple mode is enabled. To access the system execution space, you can do either of the following:

- Access the security appliance via the console or the auxiliary port.
- Log into the admin context using SSH or Telnet, and then switch to the system execution space. (The admin context is discussed earlier under the "Architectural Overview" section.

  If you are logged into the admin context, you need to use the **changeto system** command to get access to the system execution space. Example 9-5 demonstrates how to log into the system from the admin context.

  If you are in a security context, the host name contains a /. The text before the / is the host name of the security appliance, while the text after is the name of the security context. If the host name does not contain a /, you are in the system execution space.

NOTE The system execution space can also be accessed through the GUI of ASA Device Manager (ASDM). Consult Chapter 19, "Firewall Management Using ASDM," for more information.

Example 9-5 *Switching to System Execution Space*

```
Chicago/admin# changeto system
Chicago#
```

The purpose of system execution space is to define the admin and customer contexts on the appliance. A context can be added by using the **context** command followed by the name of the context. Example 9-6 shows how to add CustA and CustB security contexts in the Chicago ASA. The security context name is case sensitive, so double-check it when adding the contexts. The appliance takes you into the context subconfiguration mode (config-ctx) to configure the necessary parameters.

Example 9-6 *Adding Customer Contexts in System Execution Space*

```
Chicago# configure terminal
Chicago(config)# context CustA
Creating context 'CustA'... Done. (2)
Chicago(config-ctx)# exit
Chicago(config)# context CustB
Creating context 'CustB'... Done. (3)
Chicago(config-ctx)# end
Chicago#
```

NOTE The security appliance will not allow a system administrator to change to the newly created context until it is initialized, as discussed next in "Step 3: Specifying a Configuration URL."

The Cisco appliance allows you to add a description to the configured contexts. It is recommended that you add a description under each context for references purposes, as illustrated in Example 9-7.

Example 9-7 *Configuring a Description on the Security Context*

```
Chicago(config)# context CustA
Chicago(config-ctx)# description Customer A's Security Context
Chicago(config-ctx)# context CustB
Chicago(config-ctx)# description Customer B's Security Context
```

CAUTION If you issue the **clear configure all** command from the system configuration, the Cisco ASA removes all security contexts from the device.

Step 3: Specifying a Configuration URL

The configuration URL specifies the location of the startup configuration for each context. The configured contexts (either admin or customer) are not active unless there is a configuration URL. The supported storage locations include the local disk and a network drive using the HTTP, HTTPS, FTP, or TFTP protocol. Once a configuration URL is specified, the Cisco ASA tries to retrieve the configuration from that location. If it does not find the configuration file, the Cisco security appliance creates a configuration file with the default settings.

An administrator can choose to specify different external servers as the configuration URL location for the security contexts. As shown in Figure 9-3, the Chicago ASA has an admin and two customer contexts. The system administrator prefers to use the following:

- A TFTP server, 192.168.10.50, to store the CustA configuration
- An FTP server, 192.168.20.50, to save the CustB configuration
- The local disk for the admin context configuration

By default, these configuration files are saved in the root directory of the network protocol used by the context. For example, if the root directory of an FTP server is C:\FTP\files, the configuration URL using the FTP protocol will save the configuration file at that location. The security appliance saves the configuration of these security contexts when either **write memory** or **copy running-config startup-config** is issued from within the security context. Example 9-8 shows the relevant configuration to define configuration URL.

Example 9-8 *Setting* **config-url** *for Security Contexts*

```
Chicago(config)# context admin
Chicago(config-ctx)# config-url disk0:/admin.cfg
Chicago(config-ctx)# context CustA
Chicago(config-ctx)# config-url tftp://192.168.10.50/CustA.cfg
Chicago(config-ctx)# context CustB
Chicago(config-ctx)# config-url ftp://cisco:cisco123@192.168.20.50/CustB.cfg
```

For the FTP protocol, you have to specify a username and a password to save and retrieve the configuration file. In the previous example, the CustB context is set up to use cisco as the username and cisco123 as the password.

NOTE You cannot change the location of the configuration URL from within a context. You have to be in the system execution space to accomplish this.

When the configuration URL is changed, the appliance merges the running configuration of a context with the new configuration specified in the URL. This may add unnecessary commands and may cause system instability. If you do not want to merge the two configurations, you can follow these guidelines:

1. Log into the security context whose URL is to be changed, and clear the running configuration.
2. Log into the system execution space and enter into the context configuration mode.
3. Specify the new configuration URL that you want to use.

As soon as the new URL is entered, the appliance loads the new configuration immediately in the running configuration. Example 9-9 shows how the security appliance in Chicago can be configured to use a new configuration URL. The CustA context is currently using a

TFTP server to retrieve the startup configuration; however, the administrator wants to use an FTP server instead.

Example 9-9 *Changing the Configuration URL*

```
Chicago# change context CustA
Chicago/CustA# configure terminal
Chicago/CustA(config)# clear configure all
Chicago/CustA(config)# changeto system
Chicago(config)# context CustA
Chicago(config-ctx)# config-url ftp://cisco:cisco123@192.168.10.50/CustA.cfg
```

Step 4: Allocating the Interfaces

After defining the configuration URL, the next step is to allocate interfaces to each of the security contexts. You can assign either a physical interface or a subinterface to a security context. Do this by entering into the context subconfiguration mode and using the **allocate-interface** command:

```
allocate-interface physical_interface [map_name] [visible | invisible]
allocate-interface physical_interface.subinterface[-physical_interface.
  subinterface]
[map_name[-map_name]] [visible | invisible]
```

The **allocate-interface** command can hide the physical interface name from the security context if a mapped name is used. This provides additional security by displaying only the mapped name to the context administrator.

Table 9-3 lists and defines the arguments used in the **allocate-interface** command.

Table 9-3 **allocate-interface** *Command Definition*

Syntax	Description
physical_interface	Physical interface that is being allocated to a context, such as GigabitEthernet0/0.
subinterface	Subinterface that is being allocated to a context, such as GigabitEthernet0/0.1. A range of subinterfaces can also be specified.
map_name	By default, the allocated interface is displayed as the interface ID in the context. If you want to display the name for an interface instead of the interface ID, you can specify an alphanumeric mapped name. This is extremely useful when you do not want the context administrator to find out which physical interface is being used as the inside or the outside interface. You can also specify a range of mapped names for the corresponding range of subinterfaces.
invisible	If the **invisible** keyword is configured, the appliance does not display the interface ID in the configuration or the **show interface** command. This default option only shows the mapped name.
visible	If the **visible** keyword is configured, the appliance displays the interface ID in the configuration and the **show interface** command.

In Example 9-10, the ASA in Chicago is configured to allocate GigabitEthernet0/0 and GigabitEthernet0/1 to CustA and are mapped as A_inside and A_outside, respectively. The **invisible** option is used at the end to hide the physical interface name when the context administrator looks at the interface configuration or statistics. The appliance is also set up to allocate GigabitEthernet0/2 and GigabitEthernet0/3 to CustB. The context administrators will see these interfaces as B_inside and B_outside.

Example 9-10 *Allocating Interfaces to Security Contexts*

```
Chicago(config)# context CustA
Chicago(config-ctx)# allocate-interface GigabitEthernet0/0 A_inside invisible
Chicago(config-ctx)# allocate-interface GigabitEthernet0/1 A_outside invisible
Chicago(config-ctx)# exit
Chicago(config)# context CustB
Chicago(config-ctx)# allocate-interface GigabitEthernet0/2 B_inside invisible
Chicago(config-ctx)# allocate-interface GigabitEthernet0/3 B_outside invisible
```

NOTE If the appliance is converted from a single- to multiple-mode firewall, it allocates all the non-shutdown interfaces to the admin context. It is highly recommended that you use the admin context only for management purposes. Reallocate the interfaces to the proper contexts if necessary.

Step 5: Configuring an Admin Context

Cisco ASA creates an admin context automatically, if you convert it from single to multiple mode and you answer "yes" to "Convert the system configuration?" The admin context is treated as any other customer context in the security appliance. To log into the admin context, use the **changeto context** command, as shown in Example 9-11, where an administrator logs into admin context called **admin**.

Example 9-11 *Logging into a Security Context*

```
Chicago# changeto context admin
Chicago/admin#
```

If you would rather designate a different context as the admin context, use the following command in the system execution space:

admin-context *context_name*

where *context_name* is the name of the context you want to designate as the admin context. Before a context is declared to be an admin context, it must meet two requirements:

1. The context must be predefined and have a **config-url**.
2. The **config-url** must point to a file in the local disk.

Example 9-12 shows how to designate CustA as the admin context in a security appliance. Because CustA used a TFTP server to store the startup configuration, the

administrator is modifying it to use the local disk0 before setting up the **admin-context** command.

Example 9-12 *Setting Up an Admin Context*

```
Chicago(config)# context CustA
Chicago(config-ctx)# config-url disk0:/CustA.cfg
Chicago(config-ctx)# exit
Chicago(config)# admin-context CustA
```

Not sure which context is set up as the admin context? Use one of the following three methods to find out:

- **show running-config | include admin-context**
- **show admin-context**
- **show context**, and look for the context name with an asterisk (*)

In Example 9-13, the highlighted entries indicate that CustA is currently set as the admin context.

Example 9-13 *Verifying the Admin Context*

```
Chicago# show running-config | include admin-context
admin-context CustA
Chicago# show admin-context
Admin: CustA disk0/:CustA.cfg
Chicago# show context
Context Name   Interfaces             URL
 admin         Management0/0          disk0:/admin.cfg
*CustA         GigabitEthernet0/0,    disk0:/CustA.cfg
               GigabitEthernet0/1
 CustB         GigabitEthernet0/2,     ftp://cisco:cisco123@192.168.20.50/CustB.cfg
               GigabitEthernet0/3
```

Step 6: Configuring a Customer Context

Any context that is not set up as the admin context is referred to as the customer context. As mentioned earlier in this chapter, a customer context is configured similarly to a standalone firewall, with a few exceptions that are listed in Table 9-1. When an administrator logs into a customer context, the command prompt displays the name of that context, as shown in Example 9-14.

Example 9-14 *Logging Into a Security Context*

```
Chicago# change to context CustA
Chicago/CustA#
```

After logging into the customer context, you can configure all the supported firewall-related options.

NOTE The security appliance does not save the configuration of all security contexts if copy running-config startup-config is executed from the system execution space. If the security appliance needs to be reloaded, log into all the security contexts to save configuration.

Step 7: Managing the Security Contexts (Optional)

Cisco ASA provides many ways to manage and optimize system resources. For example, if a context name is mistyped or if it needs to be deleted, you can remove it by typing **no context** followed by the name of that context. In Example 9-15, the administrator of the Chicago ASA does not want to use CustB as a customer context anymore; instead, the administrator wants to remove it from system configuration. By deleting any unused security context, you do not waste security contexts, which are restricted by the system license. Additionally, the system does not have to allocate CPU and memory resources to maintain the unused contexts.

Example 9-15 *Removing a Security Context*

```
Chicago(config)# no context CustB
WARNING: Removing context 'CustB'
Proceed with removing the context? [confirm]
Removing context 'CustB' (4)... Done
```

In a situation where all contexts need to be removed, you can use the **clear configure context** command, as shown in Example 9-16.

Example 9-16 *Removing All Security Contexts*

```
Chicago(config)# clear configure context
```

CAUTION The **clear configure context** command also removes the designated admin context. If you are remotely logged into the appliance over a telnet or a SSH session, you will lose connectivity to the security appliance.

Deployment Scenarios

The virtual firewall solution is useful in deployments where multiple firewalls are needed to protect traffic to and from the trusted networks. Although virtual firewalls can be deployed in many ways, for ease of understanding, we cover two design scenarios:

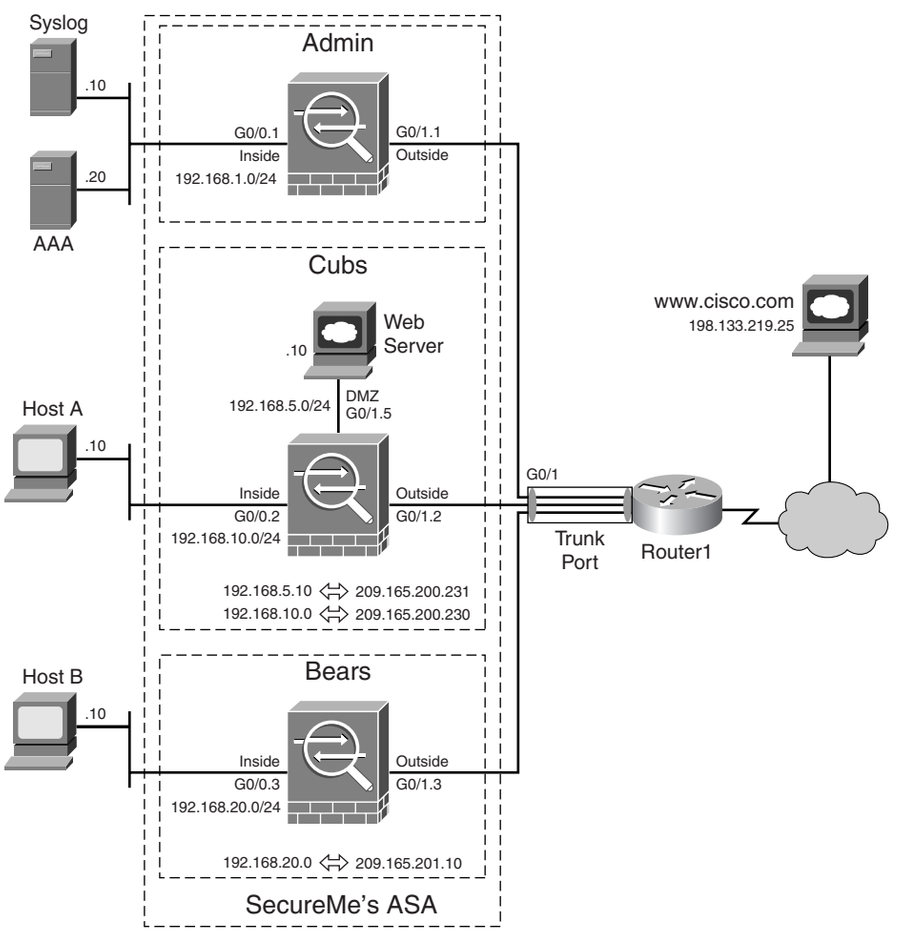
- Virtual firewall using two customer contexts
- Virtual firewall using a shared interface

NOTE The design scenarios discussed in this section are used solely to reinforce learning. They should be used for reference purposes only.

Virtual Firewall Using Two Customer Contexts

SecureMe has an office in Brussels that provides firewall services to two small companies, Cubs and Bears. SecureMe's office is located in the same building as the offices of these companies. Cubs and Bears have specific requirements that SecureMe is obliged to meet. However, the appliance in Brussels has two active physical interfaces and, as a result, SecureMe wants to use subinterfaces to accommodate these customers. To conserve public addresses on the outside interfaces, the administrator uses a subnet mask of 255.255.255.248. Figure 9-5 shows SecureMe's new topology that will be set up in Brussels.

Figure 9-5 *SecureMe Brussels Multimode Topology*



The security requirements for SecureMe, along with Cubs and Bears, are as follows:

SecureMe security requirements:

- For SSH and Telnet user authentication, use a AAA server.
- Log all the system-generated messages to a syslog server.

Cubs security requirements:

- All hosts on 192.168.10.0/24 should be able to access the Internet.
- The source IP address should be translated to 209.165.200.230 using PAT.
- Allow HTTP clients from the Internet to access Cub's web server (192.168.5.10) on the DMZ network. This address should appear as 209.165.200.231 for the Internet users.
- Deny and log all other inbound traffic on the outside interface.

Bears security requirements:

- Allow hosts on the 192.168.20.0/24 subnet to access www.cisco.com only. All other web traffic should be blocked.
- The source IP address should be translated to 209.165.201.10 using interface PAT.
- Block and log all inbound traffic on the outside interface.

Example 9-17 shows the relevant configuration to achieve the goals just listed.

Example 9-17 *ASA's Relevant Configuration with Multiple Security Contexts*

System Execution Space

```
Brussels# show run
ASA Version 7.0(1) <system>
! Main GigabitEthernet0/0 interface
interface GigabitEthernet0/0
! Sub-interface assigned to the admin context as the inside interface. A VLAN ID is
   !assigned to the interface
interface GigabitEthernet0/0.1
 vlan 5
! Sub-interface assigned to the Cubs context as the inside interface. A VLAN ID is
   !assigned to the interface
interface GigabitEthernet0/0.2
 vlan 10
! Sub-interface assigned to the Bears context as the inside interface. A VLAN ID is
   !assigned to the interface
interface GigabitEthernet0/0.3
 vlan 20
! Main GigabitEthernet0/1 interface
interface GigabitEthernet0/1

! Sub-interface assigned to the admin context as the outside interface. A VLAN ID
   is !assigned to the interface
interface GigabitEthernet0/1.1
 vlan 101
```

continues

Example 9-17 *ASA's Relevant Configuration with Multiple Security Contexts (Continued)*

```
! Sub-interface assigned to the Cubs context as the outside interface. A VLAN ID is
   !assigned to the interface
interface GigabitEthernet0/1.2
 vlan 110
! Sub-interface assigned to the Bears context as the outside interface. A VLAN ID
   is !assigned to the interface
interface GigabitEthernet0/1.3
 vlan 120
! Sub-interface assigned to the Cubs context as the DMZ interface. A VLAN ID is
   !assigned to the interface
interface GigabitEthernet0/1.5
 vlan 130
!
hostname Brussels
! context named "admin" is the designated Admin context
admin-context admin
! "admin" context definition along with the allocated interfaces.
context admin
  description admin Context for admin purposes
  allocate-interface GigabitEthernet0/0.1
  allocate-interface GigabitEthernet0/1.1
  config-url disk0:/admin.cfg
! "Cubs" context definition along with the allocated interfaces.
context Cubs
  description Cubs Customer Context
  allocate-interface GigabitEthernet0/0.2
  allocate-interface GigabitEthernet0/1.2
  allocate-interface GigabitEthernet0/1.5
  config-url disk0:/Cubs.cfg
! "Bears" context definition along with the allocated interfaces.
context Bears
  description Bears Customer Context
  allocate-interface GigabitEthernet0/0.3
  allocate-interface GigabitEthernet0/1.3
  config-url disk0:/Bears.cfg
```

Admin Context

```
Brussels/admin# show running
ASA Version 7.0(1) <context>
!inside interface of the admin context with security level set to 100
interface GigabitEthernet0/0.1
 nameif inside
 security-level 100
 ip address 192.168.1.1 255.255.255.0
!outside interface of the admin context with security level set to 0
interface GigabitEthernet0/1.1
 nameif outside
 security-level 0
 ip address 209.165.202.130 255.255.255.248
!
hostname admin
!configuration of a syslog server with logging level set to emergencies with
   timestamp
logging enable
```

Example 9-17 *ASA's Relevant Configuration with Multiple Security Contexts (Continued)*

```
logging timestamp
logging trap emergencies
logging host inside 192.168.1.10
!
route outside 0.0.0.0 0.0.0.0 209.165.202.129 1
!configuration of a AAA server using RADIUS for authentication
aaa-server uauth protocol radius
aaa-server uauth host 192.168.1.20
 key cisco123
!setting up telnet and SSH authentication
aaa authentication telnet console uauth
aaa authentication ssh console uauth
!Telnet to the admin context is allowed from the inside interface
telnet 192.168.1.0 255.255.255.0 inside
telnet timeout 5
!SSH to the admin context is allowed from the outside interface
ssh 0.0.0.0 0.0.0.0 outside
ssh timeout 5
```

Cubs Context

```
Brussels/Cubs# show running
ASA Version 7.0(1) <context>
!inside interface of the Cubs context with security level set to 100
interface GigabitEthernet0/0.2
 nameif inside
 security-level 100
 ip address 192.168.10.1 255.255.255.0
!outside interface of the Cubs context with security level set to 0
interface GigabitEthernet0/1.2
 nameif outside
 security-level 0
 ip address 209.165.200.225 255.255.255.248
!DMZ interface of the Cubs context with security level set to 50
interface GigabitEthernet0/1.5
 nameif dmz
 security-level 50
 ip address 192.168.5.1 255.255.255.0
!
hostname Cubs
!Access-list configuration to allow web traffic. The access-list is applied to the
   outside interface.
access-list outside-in extended permit tcp any host 209.165.200.231 eq www
access-list outside-in extended deny ip any any log
access-group outside-in in interface outside
!NAT configuration to allow inside hosts to get Internet connectivity
global (outside) 1 209.165.200.230
nat (inside) 1 192.168.10.0 255.255.255.0

!Static address translation for the Web-Server
static (dmz,outside) 209.165.200.231 192.168.5.10 netmask 255.255.255.255
!
route outside 0.0.0.0 0.0.0.0 209.165.200.226 1
```

continues

Example 9-17 *ASA's Relevant Configuration with Multiple Security Contexts (Continued)*

Bears Context

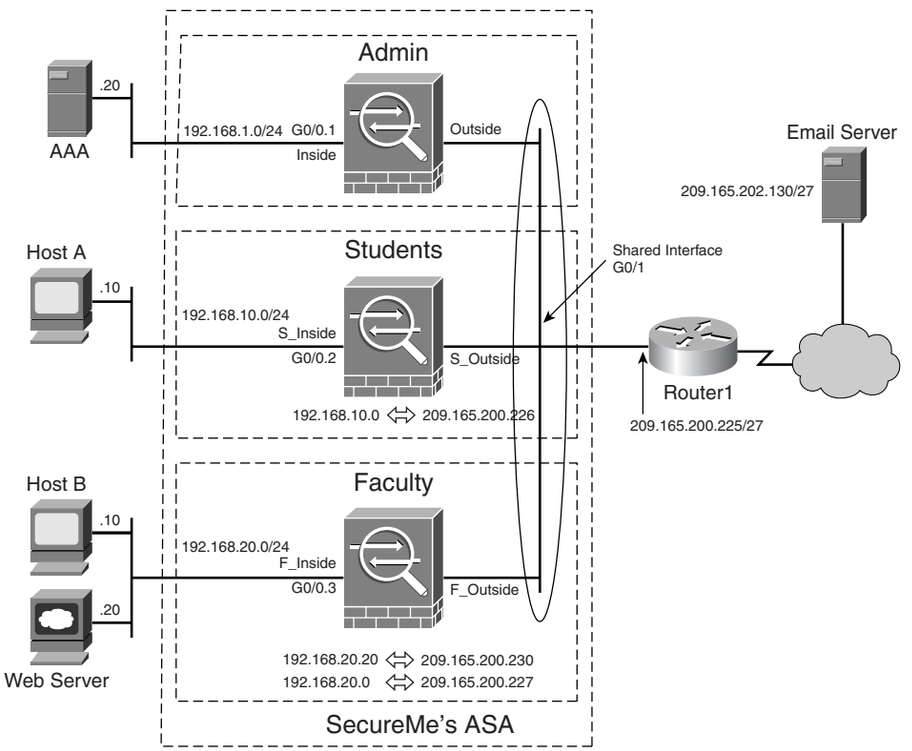
```
Brussels/Bears# show running
ASA Version 7.0(1) <context>
!inside interface of the Bears context with security level set to 100
interface GigabitEthernet0/0.3
 nameif inside
 security-level 100
 ip address 192.168.20.1 255.255.255.0
!outside interface of the Bears context with security level set to 0
interface GigabitEthernet0/1.3
 nameif outside
 security-level 0
 ip address 209.165.201.2 255.255.255.224
!
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname Bears
!Access-list configuration to permit web traffic initiated from the inside host and
   destined to 198.133.219.25. Deny all other traffic. The access-list is applied
   to the inside interface.
access-list inside-in extended permit tcp 192.168.20.0 255.255.255.0 host
198.133.219.25   eq 80
access-group inside-in in interface inside
!Access-list configuration to deny and log all inbound traffic. The access-list is
   applied to the outside interface
access-list outside-in extended deny ip any any log
access-group outside-in in interface outside
!NAT configuration to allow inside hosts to get Internet connectivity
global (outside) 1 interface
nat (inside) 1 192.168.20.0 255.255.255.0
!
route outside 0.0.0.0 0.0.0.0 209.165.201.1 1
```

Virtual Firewall Using a Shared Interface

An educational institute contacts SecureMe to provide firewall services for two of its departments—faculty and students—over a shared outside interface. The hosts in the student context are allowed to access a web server in the faculty context. Additionally, they are allowed to check their e-mail messages from 209.165.202.130. The faculty context, on the other hand, does not restrict anything going out to the Internet.

The SecureMe global policy restricts access of the security appliance to the valid and authorized users on the AAA servers. SecureMe does not have many public addresses available, so it is using interface PAT for address translation. Additionally, SecureMe does not want the administrators of the individual security contexts to be able to determine the interface assignment for their contexts. Figure 9-6 shows SecureMe's proposed topology for this institute.

Figure 9-6 *Security Contexts Using a Shared Interface*



Example 9-18 shows the relevant configuration for the Cisco ASA used in this deployment.

Example 9-18 *ASA's Relevant Configuration Using a Shared Outside Interface*

System Execution Space

```
SecuremeInstitute# show run
ASA Version 7.0(1) <system>
! Main GigabitEthernet0/0 interface
interface GigabitEthernet0/0

! Sub-interface assigned to the admin context as the inside interface. A VLAN ID is
   !assigned to the interface
interface GigabitEthernet0/0.1
 vlan 5
! Sub-interface assigned to the Students context as the inside interface. A VLAN ID
   is !assigned to the interface
interface GigabitEthernet0/0.2
 vlan 10
! Sub-interface assigned to the Faculty context as the inside interface. A VLAN ID
   is !assigned to the interface
interface GigabitEthernet0/0.3
 vlan 20
```

continues

Example 9-18 *ASA's Relevant Configuration Using a Shared Outside Interface (Continued)*

```
! Main GigabitEthernet0/1 interface to be used as the shared interface
interface GigabitEthernet0/1
!
hostname SecuremeInstitute
! context named "admin" is the designated Admin context
admin-context admin
! "admin" context definition along with the allocated interfaces.
context admin
  description admin Context for admin purposes
  allocate-interface GigabitEthernet0/0.1 inside invisible
  allocate-interface GigabitEthernet0/1 outside invisible
  config-url disk0:/admin.cfg
! "Students" context definition along with the allocated interfaces.
context Students
  description Students Customer Context
  allocate-interface GigabitEthernet0/0.2 S_inside invisible
  allocate-interface GigabitEthernet0/1 S_outside invisible
  config-url disk0:/Students.cfg
! "Faculty" context definition along with the allocated interfaces.
context Faculty
  description Faculty Customer Context
  allocate-interface GigabitEthernet0/0.3 F_inside invisible
  allocate-interface GigabitEthernet0/1 F_outside invisible
  config-url disk0:/Faculty.cfg
```

Admin Context

```
SecuremeInstitute/admin# show running
ASA Version 7.0(1) <context>
!inside interface of the admin context with security level set to 100
interface inside
 nameif inside
 security-level 100
 ip address 192.168.1.1 255.255.255.0
!outside interface of the admin context with security level set to 0
interface outside
 nameif outside
 security-level 0
 ip address 209.165.200.225 255.255.255.224
!
hostname admin
!
route outside 0.0.0.0 0.0.0.0 209.165.200.230 1
!configuration of a AAA server using RADIUS for authentication
aaa-server uauth protocol radius
aaa-server uauth host 192.168.1.20
 key cisco123
aaa authentication telnet console uauth
aaa authentication ssh console uauth
!Telnet to the admin context is allowed from the inside interface
telnet 192.168.1.0 255.255.255.0 inside
telnet timeout 5
!SSH to the admin context is allowed from the outside interface
ssh 0.0.0.0 0.0.0.0 outside
ssh timeout 5
route outside 0.0.0.0 0.0.0.0 209.165.200.225 1
```

Example 9-18 *ASA's Relevant Configuration Using a Shared Outside Interface (Continued)*

Students Context

```
SecuremeInstitute/Students# show running
ASA Version 7.0(1) <context>
!inside interface of the Students context with security level set to 100
interface S_inside
 nameif inside
 security-level 100
 ip address 192.168.10.1 255.255.255.0
!outside interface of the Students context with security level set to 0
interface S_outside
 nameif outside
 security-level 0
 ip address 209.165.200.226 255.255.255.224
!Access-list configuration to allow email and web traffic. The access-list is
   applied to the inside interface.
access-list inside-in extended permit tcp 192.168.10.0 255.255.255.0 host
   209.165.202.130 eq smtp
access-list inside-in extended permit tcp 192.168.10.0 255.255.255.0 host
   209.165.200.230 eq www
access-group inside-in in interface S_inside
!
hostname Students
!NAT configuration to allow inside hosts to get Internet connectivity
global (S_outside) 1 interface
nat (S_inside) 1 192.168.10.0 255.255.255.0
!
route S_outside 0.0.0.0 0.0.0.0 209.165.200.225 1
```

Faculty Context

```
SecuremeInstitute/Faculty# show running
ASA Version 7.0(1) <context>
!inside interface of the Faculty context with security level set to 100
interface F_inside
 nameif inside
 security-level 100
 ip address 192.168.20.1 255.255.255.0
!outside interface of the Faculty context with security level set to 0
interface F_outside
 nameif outside
 security-level 0
 ip address 209.165.200.227 255.255.255.224
!
enable password 8Ry2YjIyt7RRXU24 encrypted
passwd 2KFQnbNIdI.2KYOU encrypted
hostname Faculty
!Access-list configuration to allow web traffic. The access-list is applied to the
   outside interface.
access-list outside-in extended permit tcp host 209.165.200.226 host 209.165.200.230
eq www
access-group outside-in in interface F_outside
!NAT configuration to allow inside hosts to get Internet connectivity
global (F_outside) 1 interface
```

continues

Example 9-18 *ASA's Relevant Configuration Using a Shared Outside Interface (Continued)*

```
nat (F_inside) 1 192.168.20.0 255.255.255.0
!Static address translation for the Web-Server
static (F_inside,F_outside) 209.165.200.230 192.168.20.20 netmask 255.255.255.255
!
route F_outside 0.0.0.0 0.0.0.0 209.165.200.225 1
```

Monitoring and Troubleshooting the Security Contexts

Cisco ASA provides **show** and **debug** commands that are useful to check the health of the appliance or to isolate a problem. The necessary **show** and **debug** commands that are used to manage multiple security contexts in the appliance are discussed here.

Monitoring

After the system is converted to multiple contexts, the first thing to verify is that the system is using the new mode by using the **show mode** command, as shown in Example 9-19.

Example 9-19 *Output of* **show mode**

```
Sydney# show mode
Security context mode: multiple
```

Once you verify that the system is running in multiple mode, configure the necessary contexts and assign the appropriate interfaces. A good way to check if the interfaces have been correctly assigned to the right context is to use the **show context** command. It lists all the configured contexts, the allocated interfaces, and the configuration URL. Example 9-20 shows the output of **show context** in the Chicago ASA, while logged into the system execution space.

Example 9-20 *Output of* **show context** *in the System Execution Space*

```
Chicago# show context
Context Name      Interfaces                URL
*admin            GigabitEthernet0/0.1      disk0:/admin.cfg
                  GigabitEthernet0/1.1
 Cubs             GigabitEthernet0/0.2,     disk0:/Cubs.cfg
                  GigabitEthernet0/1.2,
                  GigabitEthernet0/1.5
 Bears            GigabitEthernet0/0.3      disk0:/Bears.cfg
                  GigabitEthernet0/1.3
Total active Security Contexts: 3
```

The asterisk (*) next to admin indicates that this is an admin context. Another way to find out which context is designated as the admin context is to use the **show admin-context** command, as illustrated in Example 9-21.

Example 9-21 *Output of* **show admin-context** *in the System Execution Space*

```
Chicago# show admin-context
Admin: admin disk0:/admin.cfg
```

A context administrator can view the context settings from within his security context. In Example 9-22, the administrator of the Cubs context is verifying the allocated interfaces and the configuration URL.

Example 9-22 *Output of* **show context** *from a Security Context*

```
Chicago/Cubs# show context
Context Name      Interfaces                URL
 Cubs             GigabitEthernet0/0.2,     disk0:/Cubs.cfg
                  GigabitEthernet0/1.2,
                  GigabitEthernet0/1.5
```

Cisco ASA allows monitoring of CPU usage per security context. This is useful to determine which context is utilizing the most of the CPU cycles. Use the **show cpu usage context all** command to check the CPU utilization on each of the configured security contexts. In Example 9-23, the total system CPU utilization is 9.5 percent averaged over 5 seconds, 9.2 percent averaged over 1 minute, and 9.3 percent averaged over 5 minutes. The Cubs security context is using the most of the CPU cycles, averaging 5 percent over 5 seconds, 1 minute, and 5 minutes.

Example 9-23 *Output of* **show cpu usage context all** *from the System Execution Space*

```
Chicago# show cpu usage context all
5 sec  1 min  5 min  Context Name
9.5%   9.2%   9.3%   system
0.3%   0.0%   0.1%   admin
5.0%   5.0%   5.0%   Cubs
4.2%   4.2%   4.2%   Bears
```

Troubleshooting

For troubleshooting purposes, Cisco ASA includes a number of important debug and syslog messages to help you isolate the issue. This section discusses four troubleshooting scenarios related to security contexts:

- **Security contexts are not added**—When adding new contexts, the Cisco security appliance displays a message that the new security contexts creation failed, as shown in Example 9-24.

Example 9-24 *Security Context Creation Failure*

```
Chicago(config)# context  WhiteSox
Creating context 'WhiteSox'...
Cannot create context 'WhiteSox': limit of 3 contexts exceeded
ERROR: Creation for context 'WhiteSox' failed
```

The Cisco ASA appliance complains about exceeding the maximum number of security contexts allowed in this device. To verify the maximum number of allowed security contexts, use the **show version** command, as shown in Example 9-25.

Example 9-25 *Verifying the Maximum Number of Security Contexts*

```
Chicago# show version | include Security Contexts
Security Contexts            : 10
```

Depending on the security appliance model number, the administrator can add the maximum allowed security context number. Refer to Chapter 3, "Hardware Overview," for more information about the allowed number of security contexts in a Cisco ASA.

- **Security contexts are not saved on the local disk**—If the security context configuration files are stored locally on the disk, and the appliance is having trouble either retrieving or saving them, you can enable **debug disk** to gather information.

  In Example 9-26, **debug disk file**, **file-verbose**, and **filesystem** are enabled with a log level of 255. In this example, the administrator saves the running configuration into the Flash file system. The highlighted entries show that the appliance opens up the running configuration file from the disk and writes the new contents. If Flash is corrupt, the administrator will see failed attempts to read or write files. These messages are analyzed by the Cisco Technical Assistance (TAC) engineers.

Example 9-26 *Output of* **debug disk**

```
Chicago# debug disk file 255
Chicago# debug disk file-verbose 255
Chicago# debug disk filesystem 255
Chicago# write memory
Building IFS: Opening: file system:/running-config, flags 1, mode 0
IFS: Opened: file system:/running-config as fd 0
IFS: Fioctl: fd 0, fn 5, arg 370e7e0
configuration...
IFS: Read: fd 1, bytes 147456
IFS: Read: fd 1, bytes 146664
IFS: disk0:/.private/startup-config 100% chance ascii text
<snip>
1047 IFS: Close: fd 0
bytes copied in 4.40 secs (261 bytes/sec)IFS: Write: fd 0, bytes 1
```

- **Security contexts are not saved on the FTP server**—If the security appliance is having issues when saving and retrieving configuration files from an FTP server, use the **debug ftp client** command to isolate the issue. In Example 9-27, the appliance is being configured to use an FTP server. The debug shows that the user password is incorrect in the configuration URL.

Example 9-27 *Output of* **debug ftp client**

```
Chicago(config)# debug ftp client
Chicago# context CustB
Chicago(config-ctx)# config-url ftp://cisco:cisco123@172.18.124.27/CustB.cfg
IFS: Opening: file ftp://cisco:cisco123@192.168.20.50/CustB.cfg, flags 1, mode 0
IFS: Opened: file ftp://cisco:cisco123@192.168.20.50/CustB.cfg as fd 0
IFS: Fioctl: fd 0, fn 5, arg 279bc64
Loading CustB.cfg
FTP: 220 Please enter your user name.
FTP: ---> USER cisco
FTP: 331 User name okay, Need password.
FTP: ---> PASS *
FTP: 530 Password not accepted.
FTP: ---> QUIT
FTP: 221 Goodbye. Control connection closed.
IFS: Close: fd 0
```

- **User connectivity issues using shared security contexts**—As shown in Figure 9-6, when Host A in the Students context is not able to reach Host B in the Faculty context, the administrator can take the following steps to isolate the issue:

  Step 1 Ping the inside IP address of the S_inside interface from Host A. If successful, move to Step 2; otherwise, check to see if there is an inbound ACL applied to the inside interface. Also verify that physical connectivity exists between the host and the inside interface.

  Step 2 Ping the outside IP address of the F_outside interface from Host A. If successful, move to Step 3; otherwise, check the outbound ACL and NAT configuration on the Students context. Additionally, verify that the inbound ACL on the F_outside interface allows ICMP traffic from Host A.

  Step 3 Because this topology uses a shared interface, check the NAT configuration on the Faculty context. Ping from Host A to Host B and verify that the outbound ACL on F_inside does not block the ICMP packets.

  Step 4 If Host A is still not able to communicate with Host B in the other context, follow Step 1 through Step 3 and ping from Host B to Host A to verify connectivity and the contexts' configuration.

Summary

Security context is a robust feature available in Cisco ASA. It provides a cost-effective solution by having multiple firewalls integrated into one physical appliance. Each security context has its own interfaces, security policies, and routing tables. The packets traversed through the security contexts are classified based on the source interface or the destination IP address. This chapter discussed the configuration steps and provided deployment scenarios to help you to understand this concept better. For troubleshooting purposes, the chapter introduced the relevant **show** commands and walked you through how to isolate the issues related to security contexts.