CHAPTER 5

Implementation of Security

Overview

The number of skilled hackers has multiplied, and a variety of sophisticated hacking tools are freely available on the Internet. These tools exploit the way the network is designed to work, and are simple enough for even a novice to use. This combination has dramatically increased the risk to networks.

Some of the more dangerous types of attacks include the following:

- Packet sniffer—Software application that uses a network adapter card in promiscuous mode (a mode in which the network adapter card sends all packets received on the physical network wire to an application for processing) to capture all network packets that are sent across a particular collision domain. Sniffers are used legitimately in networks today to aid in troubleshooting and traffic analysis. However, because several network applications send data in clear text (Telnet, File Transfer Protocol [FTP], Simple Message Transfer Protocol [SMTP], Post Office Protocol [POP3], and so on), a packet sniffer can provide meaningful and often sensitive information, such as usernames and passwords. One serious problem with acquiring usernames and passwords is that users often reuse their login names and passwords across multiple applications and systems.
- IP spoofing—A hacker inside or outside a network impersonates the conversations of a trusted computer. The hacker uses either an IP address that is within the range of trusted IP addresses for a network, or an authorized external IP address that is trusted and to which access is provided to specified resources on a network. IP spoofing attacks are often a launch point for other attacks. The classic example is to launch a denial-of-service (DoS) attack using spoofed source addresses to hide the identity of the hacker.
- Distributed denial-of-service (DDoS) attacks—Multiple compromised systems flood the bandwidth or resources of a targeted system, usually one or more web servers. Although the attack does not flood the entire network with traffic, it overwhelms the specific device and takes it out of service. These systems are compromised by attackers using a variety of methods. Malware can carry DDoS attack mechanisms; one of the more well-known examples of this was MyDoom. Its DoS mechanism was triggered on a specific date and time. This type of DDoS attack involved hard-coding the target IP address before release of the malware. No further interaction was necessary to launch the attack.
- Network reconnaissance—Learning information about a target network by using publicly available information and applications. When hackers attempt to penetrate a particular network, they often need to learn as much information as possible about the network before launching attacks. This can take the form of DNS queries, ping sweeps, and port scans. DNS queries can reveal such information as who owns a particular domain and what addresses have been assigned to that domain. Ping sweeps of the addresses revealed by the DNS queries can present a picture of the live hosts in a particular environment. After such a list is generated, port scanning tools can cycle through all well-known ports to provide a complete list of all services running on the hosts discovered by the

ping sweep. Finally, the hackers can examine the characteristics of the applications that are running on the hosts. This scenario can lead to specific information that is useful when the hacker attempts to compromise that service.

- Unauthorized access—Although unauthorized access attacks are not a specific type of attack, they refer to most attacks executed in networks today. A brute-force attack on a Telnet login requires the Telnet prompt on a system. On connection to the Telnet port, a message might indicate "authorization required to use this resource." If the hacker continues to attempt access, their actions become "unauthorized." These kinds of attacks can be initiated on both the outside and inside of a network.
- Virus and Trojan horse applications—The primary vulnerabilities for end-user workstations are viruses and Trojan horse attacks. Viruses refer to malicious software that is attached to another program to execute a particular unwanted function on a user workstation. An example of a virus is a program that is attached to command.com (the primary interpreter for Windows systems), which deletes certain files and infects any other versions of command.com that it can find. A Trojan horse is different only in that the entire application is written to look like something else, when in fact it is an attack tool. An example of a Trojan horse is a software application that runs a simple game on the user workstation. While the user is occupied with the game, the Trojan horse mails a copy of itself to every other user in the address book of the user. Other users then get the game and play it, thus spreading the Trojan horse.
- Password attacks—Hackers can implement password attacks using several methods, including brute-force attacks, Trojan horse programs, IP spoofing, and packet sniffers. Although packet sniffers and IP spoofing can yield user accounts and passwords, password attacks usually refer to repeated attempts to identify a user account or password. These repeated attempts are called brute-force attacks. Often, a brute-force attack is performed using a program that runs across the network and attempts to log into a shared resource, such as a server. When hackers successfully gain access to resources, they have the same rights as the users whose accounts have been compromised to gain access to those resources. If the compromised accounts have sufficient privileges, the hackers can create back doors for future access without concern for any status and password changes to the compromised user accounts.

The goal of the comprehensive model provided here is to prevent attacks by keeping the outsiders out and the insiders honest. Specific goals include the following:

- Prevent external hackers from getting access to the network
- Allow only authorized users into the network
- Prevent those inside the network from executing deliberate or inadvertent attacks
- Provide various levels of access for various types of users

To be truly effective, the security policy must do this in a way that is transparent to the users and easy to administer, and that does not disrupt the operations of the plant floor.

To accomplish all this, the solution needs to provide the following:

- Network-wide security that is fully embedded into the network infrastructure
- Protection, prevention, and self-protection
- Control over who has network access and what they can do

The following security components of the EttF 1.1 solution address the major security concerns of defending against threat, establishing trust boundaries and verifying identity, and securing business communications:

- Device hardening
- Threat defense—Guard the network against malicious as well as unintentional attack. Threat defense can be further broken down into the following goals:
  - Defending the edge—Using Cisco Adaptive Security Appliance (ASA) integrated firewalls and intrusion detection systems (IDS) to fortify the network edge against intrusion and attack.
  - Protecting the interior—Enabling Cisco IOS security features on routers and switches to protect the network against emerging internal attacks.
  - Guarding the endpoints—Using the Cisco Security Agent (CSA) to proactively defend against infection and damage to hosts, such as human-machine interfaces (HMIs), servers, and PCs.
  - Trust and identity—Controlling who has access from the enterprise network to the plant floor network. This control is provided by CiscoSecure Access Control Server (ACS).
- Secure communications—Protecting the confidentiality of internal and external data communication.

Network Device Hardening

Device hardening refers to changing the default posture of a system out of the box to make it more secure. These network devices include, among others, routers, switches, firewalls, and network-based intrusion detection system (NIDS). The default security of these devices can differ, which changes the amount of work required to harden a particular device.

An important characteristic of all these devices is the availability of a console port. The console port has privileged access to these devices because it generally implies physical access to the device (though this could be a modem). The console port defaults to having initial authentication that is weak or nonexistent and is able to send a break signal to the device upon boot. This is used to reset most of these types of devices or to recover from a lost password.

Because of the capabilities of a console port, it is important to control physical access to networking devices whenever possible.

Note This section on network devices assumes that the devices are not running on general-purpose operating systems. If they are, be sure to run the host operating system-hardening as well as the network device-hardening steps.

From a configuration perspective, the methods for hardening a router or switch are very similar.

Table 5-1 summarizes the device hardening techniques needed for the platforms supported by the EttF 1.1 solution. The detailed configuration is presented in the following sections.

Table 5-1 Device Hardening Techniques

	Catalyst 2955	Catalyst 3750	Catalyst 4500
Disable unneeded services—DNS lookup	Yes	Yes	Yes
Disable unneeded services—Small services	Yes	Yes	Yes
Disable unneeded services—BootP server	N/A	Yes	Yes

Table 5-1 Device Hardening Techniques (continued)

Disable unneeded services—Source routing and directed broadcast	N/A	Yes	Yes
Disable unneeded services—Proxy ARP	N/A	Yes	Yes
Disable unneeded services—ICMP redirects	N/A	Yes	Yes
Password encryption	Yes	Yes	Yes
Authentication settings—Enable secret	Yes	Yes	Yes
Authentication settings—Login banner	Yes	Yes	Yes
Authentication settings—Line access	Yes	Yes	Yes
Authentication settings—Set up usernames	Yes	Yes	Yes
Authentication settings—Secure Shell (SSH)	Yes (supported only by crypto image)	Yes (supported only by crypto image)	Yes (supported only by crypto image)
Management access—HTTP server	Yes	Yes	Yes
Management access—NTP	Yes	Yes	Yes
Management access—ACL Options	Yes	Yes	Yes

Router

Router hardening has recently gained attention because attacks have increasingly targeted routed infrastructure. This section outlines steps to take when hardening a router; configuration examples are for Cisco IOS devices. For more information about router hardening, see the following URLs:

- Infrastructure Protection on Cisco IOS Software-Based Platforms: http://www.cisco.com/en/US/prod/collateral/iosswrel/ps8802/ps6970/ps1838/prod_white_paper0900aecd804ac831.pdf
- Improving Security on Cisco Routers: http://www.cisco.com/en/US/tech/tk648/tk361/technologies_tech_note09186a0080120f48.shtml

Basic Hardening Settings

The following hardening steps are useful on almost every router you deploy in a network. These steps include disabling unneeded services and ensuring that passwords are encrypted whenever possible.

Disable Unneeded Services

Turn off DNS lookups for the router with the following command:

```
Router(config)#no ip domain-lookup
```

Although not strictly security-related, this is the first command to type on a fresh router before doing any other configuration (assuming, of course, you do not need domain resolution for a feature you plan to use). Otherwise, be careful to avoid input errors. Typing the command **enadle** instead of **enable** results in a long timeout while the router tries to find host “enadle” and communicate with it.

Disable small services such as echo, chargen, and discard, as well as the finger service. After Cisco IOS Release 11.3, these services are disabled by default, but it never hurts to have these commands as part of the script you use to harden a device. These small services should almost always be turned off because they have no legitimate use.

```
Router(config)#no service tcp-small-servers
Router(config)#no service udp-small-servers
Router(config)#no service finger
```

Disable the BootP server with the following command if you are not using it on your network (most do not):

```
Router(config)#no ip bootp server
```

Disable source routing and directed broadcast. These should be off by default on reasonably current routers, but verify this with the following commands:

```
Router(config-if)#no ip directed-broadcast
Router(config)#no ip source-route
```

You can disable Proxy ARP in most situations, assuming your devices are routing aware:

```
Router(config-if)#no ip Proxy-arp
```

ICMP redirects should be sent only to end systems that have multiple outbound routes from which to choose. In situations in which IP redirects are unnecessary, disable them with the following command:

```
Router(config-if)#no ip redirects
```

Password Encryption

The following command enables a simple Vigenere cipher, which encrypts most passwords on a router that would otherwise be shown as clear text in the configuration:

```
Router(config)#service password-encryption
```

This cipher, as implemented on Cisco routers, is very weak and can easily be broken. It is enabled primarily to prevent a casual observer from noting your passwords. For example, you might not want a coworker observing your work to learn the password for your router after you type **wr t.**

Authentication Settings

This section outlines authentication-related settings, including the use of **enable secret**, login banners, line access, usernames stored locally or through AAA servers, and device access by SSH.

Enable Secret

Enable strong MD5-hashed passwords for router enable mode. The following password should be used instead of the basic **enable** *password* encrypted by using **service password-encryption**. It is much more secure, though it has the same susceptibility to dictionary attacks as any hashed password. Choosing strong passwords mitigates dictionary attacks.

```
Router(config)#enable secret password
```

Login Banner

Enable a warning banner to be presented to users when they connect to the device. This sort of banner can aid in prosecution in some jurisdictions and should generally at least include a statement saying that unauthorized access is prohibited. Be sure not to disclose any information that would be useful to the attacker such as platform type, software version, owner, location, and so on.

```
Router(config)#banner motd ^
Enter TEXT message. End with the character '^'.
```

```
Enter your warning banner message here.
^
```

Line Access

On a standard Cisco router, there are three primary ways to log on:

- VTY line (**line vty 0 4**, though some routers go to 15)
- Console port (**line con 0**)
- Auxiliary port (**line aux 0**)

Fresh out of the box, only the console and aux ports can be used to access the device. Generally, only the console port is needed and not the aux port. To set up the console port, enter the following commands:

```
Router(config)#line con 0
Router(config-line)#exec-timeout 5 0
Router(config-line)#password password
Router(config-line)#login
```

These commands enable login with a local password and time out the connection after 5 minutes and 0 seconds of inactivity.

To disable the aux port, type the following commands:

```
Router(config)#line aux 0
Router(config-line)#no exec
```

Turning off exec prevents logon to the device. Additional commands such as **transport input none** or **exec-timeout 0 1** are not going to make you more secure. Controlling VTY access is separate and requires the following commands:

```
Router(config)#line vty 0 3
Router(config-line)#exec-timout 5 0
Router(config-line)#password password
Router(config-line)#login
Router(config-line)#transport input protocol
```

Typically, a router has 5 VTY lines. The preceding four commands set up access in a very similar fashion to the console port. Replace *protocol* with your method of access, preferably SSH.

Note SSH is supported only by the IOS crypto images of the respective Catalyst switching platforms.

The following eight lines reserve the last VTY port for a specific IP address. This is useful if someone is attempting to deny service to the login process on the router (which can be done without the password). You can use the access class settings referenced here for lines 0 to 3 as well. If you do, open the access control list (ACL0) to allow a wider range of IP addresses to access (for instance, your entire management subnet).

```
Router(config)#line vty 4
Router(config-line)#exec-timeout 5 0
Router(config-line)#password password
Router(config-line)#login
Router(config-line)#transport input protocol
Router(config-line)#access-class 99 in
Router(config)#access-list 99 permit host adminIP
Router(config)#access-list 99 deny any logs
```

Setting Up Usernames

If you do not have access to TACACS+ or RADIUS, local usernames can be configured on a system as follows:

```
Router(config)#username username password password
Router(config)#line vty 0 4
Router(config-line)#login local
```

The preceding commands set up a local username and password and then configure the VTY lines to use a local database.

To configure TACACS+ access to a system, you must first enable the AAA system:

```
Router(config)#aaa new-model
```

You must then define the TACACS+ host and password:

```
Router(config)#tacacs-server host ipaddr
Router(config)#tacacs-server key password
```

After setting up the host, you must define the authentication methods. The following uses TACACS+ as the default authentication but also defines the authentication method **no-tacacs**, which can be used for the console port. Using AAA for the console port is not recommended because if the network is down, you are not able to log on to the box.

```
Router(config)#aaa authentication login default group tacacs+
Router(config)#aaa authentication login no-tacacs line
```

The line parameters can then be modified based on which method you want to use to authenticate:

```
Router(config)#line vty 0 4
Router(config-line)#login authentication default
Router(config)#line con 0
Router(config-line)#login authentication no-tacacs
```

So far, these authentication, authorization, and accounting (AAA) commands have dealt only with authentication. Assume, for example, that you wanted to have a detailed log of every command typed on a router as well as when an administrator logged in or out. The following commands enable TACACS+ accounting for these events:

```
! Enable login and logout tracking for router administrators
Router(config)#aaa accounting exec default start-stop group tacacs+
! Enable command logging for exec level 1 commands (basic telnet)
Router(config)# aaa accounting commands 1 default start-stop group tacacs+
! Enable command logging for exec level 15 commands (enable mode)
Router(config)# aaa accounting commands 15 default start-stop group tacacs+
```

AAA can be very complicated, with many options. For more information about configuring AAA on Cisco devices, see the following URL: http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsaaa/index.htm.

Secure Shell (SSH)

Use SSH instead of Telnet whenever possible. To configure it, you must first define a hostname and domain name, and generate keys:

```
Router(config)#hostname hostname
Router(config)#ip domain-name yourdomain.com
Router(config)#crypto key generate rsa
```

From here, you can refer to the **transport input** command in Line Access, page 5-6. To set up the VTY lines to accept only SSH, enter the following command:

```
Router(config)#line vty 0 4
Router(config)#transport input ssh
```

There are a few other options with respect to SSH configuration. For more information, see the following URL: http://www.cisco.com/en/US/docs/ios/12_2/security/configuration/guide/scfssh.html.

Management Access

This section outlines basic settings for hardening management access, including security settings for the HTTP server, Simple Network Management Protocol (SNMP), Cisco Discovery Protocol (CDP), syslog, Network Time Protocol (NTP), and various ACL logging options.

HTTP Server

If not in use, disable the HTTP server for router management with the following command:

```
Router(config)#no ip http server
```

The embedded web server in routers has had vulnerabilities in the past, so unless you have a specific need for the HTTP functionality (such as a specific management application), it is best to disable it. If you need access to the HTTP server, use the **http access-class** command as shown:

```
Router(config)#ip http access-class 10
Router(config)#access-list 10 permit host http-mgmnt-ip
Router(config)#access-list 10 deny any log
```

You should also require HTTP authentication with the following command:

```
Router(config)#ip http authentication ?
enable Use enable passwords
local Use local username and passwords
tacacs Use tacacs to authorize user
```

TACACS+ is preferred; otherwise, a local username and password can be used. Try to avoid using the enable password.

SNMP

SNMP is widely used as a network management protocol. Unfortunately, it is UDP-based (port 161) and, until version 3, had no real security options. Earlier versions of SNMP use a community string for authentication, and it is sent in the clear with the rest of the SNMP datagram. Although version 3 offers more security, most network management applications use SNMP version 1 or version 2c.

In EttF 1.1, you need to enable SNMP if CS-MAR is implemented. If you do not plan to deploy CS-MARS or to manage a device with SNMP, you should disable it:

```
Router(config)#no snmp-server
```

If you must use SNMP v1 or v2c, consider using read-only as opposed to read-write. Much of the damage an attacker can cause with SNMP is prevented if you remove the ability to write changes. In either case, the community string should be set and managed like the root password on any system (change it regularly, and so on). At the bare minimum, an ACL should be defined that allows only your SNMP devices to query the management agents on the network device, as follows:

```
Router(config)#snmp-server community password ro 98
Router(config)#snmp-server community password rw 98
Router(config)#access-list 98 permit host snmp-server-ip
Router(config)#access-list 98 deny any log
```

If you are using SNMP v3 or want more information on the rest of the SNMP configuration, see the following URL:
http://www.cisco.com/en/US/docs/ios/12_2/configfun/configuration/guide/fcf014.html.

CDP

CDP is a proprietary Cisco protocol that provides a mechanism for Cisco devices to exchange information. The following two commands show how to globally disable CDP or, alternately, to disable it only on a specific interface:

```
Router(config)#no cdp run
Router(config-if)#no cdp enable
```

Note One exception is that CDP should remain enabled when a device supports CDP and sends its capabilities information over the protocol to the switch to which the device is attached. This enables the switch to set up the proper configuration on this access port for this endpoint. An example of such a device is the Cisco IP Phone. This is not relevant to the current EttF architecture.

Syslog

Using syslog on a router is one of the easiest ways to troubleshoot your network. Syslog servers are free (besides the hardware), and the messages generated by syslog are usually easy to understand. If you are using any kind of ACLs on a router, you need syslog; even if you are not, it is a very good idea. Enabling syslog is easy. Just enter one or more logging hosts and make sure timestamps are enabled:

```
Router(config)#service timestamps log datetime localtime msec show-timezone
Router(config)#logging syslog-ip-addr
```

Sometimes viewing messages locally on the router can be useful. Besides viewing messages as they are generated on the console, you can optionally have them buffered to router memory. You do not need a larger buffer here because these are simple text messages; even 512 KB saves lots of messages. Be sure you do not use up a significant portion of your device memory, or you might affect packet forwarding. (That is, if you have 8 MB of memory on your router, do not set the buffer size to 6 MB.) Enter the following command to enable this functionality:

```
Router(config)#logging buffered buffersize
```

You can use the **logging trap** command to set the level of logging information you receive; there is no rule for where to set this, except that the highest level of logging is almost always too much information and the lowest level does not provide enough information. Try a few different levels on your own device to determine the amount of information that makes sense in your environment. Syslog has a number of

additional options. For more information, see the following URL: http://www.cisco.com/en/US/docs/ios/12_2/configfun/configuration/guide/fcf013_ps1835_TSD_Products_Configuration_Guide_Chapter.html.

NTP

Without proper timestamps, router syslog messages are nearly useless in troubleshooting. Your networking devices can be synchronized to the same clock with NTP. Configuring NTP on a router is a simple matter of locally configuring the time zone and then pointing the router to the NTP server. In the following example, NTP authentication is enabled, and an ACL restricting NTP access to the configured NTP server is applied:

```
Router(config)#clock timezone PST -8
Router(config)#clock summer-time PDT recurring
Router(config)#ntp authenticate
Router(config)#ntp authentication-key 1 md5 password
Router(config)#ntp trusted-key 1
Router(config)#ntp access-group peer 96
Router(config)#ntp server ntp-svr-ip key 1
Router(config)#access-list 96 permit host ntp-svr-ip
Router(config)#access-list 96 deny any log
```

Although there are several free NTP services on the Internet, it is not advisable to use them for security reasons. If your time source is corrupted, your log data is useless. Instead, consider setting up a local time source that connects to a reliable, known atomic clock to maintain accurate time. NTP can be disabled on interfaces that do not expect to receive valid NTP information. Use the following command:

```
Router(config-if)#ntp disable
```

More information on NTP is available at the following URL: http://www.cisco.com/en/US/docs/ios/12_2/configfun/configuration/guide/fcf012.html.

ACL Options

By default, the last line in an ACL is an implicit deny all. Matches to this list are not logged, however. If you want to enable logging, a manual entry should be added to the ACL denying all traffic and informing the ACL to log the violation. It is possible to log permits as well, but this tends just to fill up a syslog server. To drop all traffic and log violations in a standard IP ACL, use the following command:

```
Router(config)#access-list 1 deny any log
```

For an extended IP ACL, use this command:

```
Router(config)#access-list 101 deny ip any any log
```

In addition to the basic log keyword, log input is usually available for extended ACLs. Log input adds the source interface and MAC address to the usual IP address and port number message associated with the ACL entry.

Note After hardening a router, it is a good idea to scan it with your favorite port scanner. This ensures that you are not running any services you thought you turned off.

Layer 2 Security Design

Unlike hubs, switches are able to regulate the flow of data between their ports by creating almost "instant" networks that contain only the two end devices communicating with each other at that moment in time. Data frames are sent by end systems, and their source and destination addresses are not changed throughout the switched domain. Switches maintain content-addressable memory (CAM) lookup tables to track the source addresses located on the switch ports. These lookup tables are populated by an address-learning process on the switch. If the destination address of a frame is not known or if the frame received by the switch is destined for a broadcast or multicast address, the switch forwards the frame out all ports, except for the port that the frame entered to the switch. With their ability to isolate traffic and create the "instant" networks, switches can be used to divide a physical network into multiple logical or virtual LANs (VLANs) through the use of Layer 2 traffic segmentation. In general, Layer 2 of the OSI reference model is subject to network attacks in unique ways that include the following:

- Vulnerability of the use of VLAN 1
- Spanning tree attack
- MAC flooding attack
- VLAN hopping
- 802.1Q tagging attack
- ARP attacks
- MAC spoofing attack
- DHCP starvation attack
- Rogue DHCP server attack

In EttF 1.1, the implementation of Layer 2 security protection is needed on all switches (that is, Catalyst 3750 and Catalyst 2955) in the following network areas:

- Cell/area zone
- Server farm in the manufacturing zone
- Server farm in the DMZ

Equally important is that all switches need hardening. Network Device Hardening, page 5-3 discussed how to harden network devices. This section focuses on Layer 2 security for these devices.

Note For details on how to protect the Catalyst 3750 against L3 security threats from the manufacturing zone perspective, see Security Design for the Manufacturing Zone, page 5-19.

Table 5-2 summaries the L2 vulnerabilities for which these platforms can provide protection.

Table 5-2 Layer 2 Security Threats and Switch Protection

L2 Attacks/Vulnerabilities	Catalyst 3750	Catalyst 2955
Vulnerability of the use of VLAN 1	Yes	Yes
Trust level of switch ports	Yes	Yes
Spanning tree attack	Yes	Yes
MAC flooding attack (Port Security)	Yes	Yes
Broadcast/multicast storm control	Yes	Maybe

Table 5-2 Layer 2 Security Threats and Switch Protection

VLAN hopping	Yes	Yes
ARP attack	Yes	Lack of Dynamic ARP Inspection feature support
MAC spoofing attack	Yes	Yes
DHCP starvation attack	Yes	Yes
Rogue DHCP server attack	Yes	Yes

Precautions for the Use of VLAN 1

The reason VLAN 1 became a special VLAN is that L2 devices needed to have a default VLAN to assign to their ports, including their management port(s). In addition to that, many L2 protocols such as Cisco Discovery Protocol (CDP), Port Aggregation Protocol (PAgP), and VLAN Trunking Protocol (VTP) needed to be sent on a specific VLAN on trunk links. For all these purposes VLAN 1 was chosen.

As a consequence, VLAN 1 may sometimes end up unwisely spanning the entire network if not appropriately pruned and, if its diameter is large enough, the risk of instability can increase significantly. In addition, the practice of using a potentially omnipresent VLAN for management purposes puts trusted devices to higher risk of security attacks from untrusted devices that by misconfiguration or pure accident gain access to VLAN 1 and try to exploit this unexpected security hole.

To redeem VLAN 1 from its bad reputation, a simple common sense security principle can be used: as a general security rule, the network administrator should prune any VLAN, and in particular VLAN 1, from all the ports where that VLAN is not strictly needed.

Therefore, with regard to VLAN 1, the above rule simply translates into the following recommendations:

- Do not use VLAN 1 for inband management traffic; preferably pick a different, specially-dedicated VLAN that keeps management traffic separate from industrial Ethernet and other user data traffic.
- Prune VLAN 1 from all the trunks and from all the access ports that do not require it (including not connected and shutdown ports).

As a general design rule, it is desirable to prune unnecessary traffic from particular VLANs. For example, it is often desirable to apply VLAN ACLs and/or IP filters to the traffic carried in the management VLAN to prevent all Telnet connections and to allow only SSH sessions. Alternatively, it may be desirable to apply QoS ACLs to rate limit the maximum amount of ping traffic allowed.

If VLANs other than VLAN 1 or the management VLAN represent a security concern, automatic or manual pruning should be applied as well. In particular, configuring VTP in transparent or off mode is commonly considered as the most effective method:

```
Switch(config)#vtp mode transparent
```

Note The discussion of having more than one VLAN applies only to the server farm in the manufacturing zone in EttF 1.1. The cell/area zone in this solution phase is assumed to have only one VLAN.

Trust Level of Switch Ports

After proper handling of VLAN 1 has been decided on and implemented, the next logical step is to consider other equally important best practices commonly used in secure environments. The general security principle applied here is to connect untrusted devices to untrusted ports, trusted devices to trusted ports, and to disable all the remaining ports.

The recommendations are as follows:

- If a port on a Catalyst switch in the cell ring is connected to a "foreign" device, such as a drive, HMI, I/O, PAC, or historian, make sure to disable CDP, DTP, PAgP, UDLD, and any other unnecessary protocol, and to enable switch port mode access, PortFast, and BPDU Guard on it, as in the following example:

```
Switch(conf)#vtp mode transparent
Switch(conf)#interface type/slot port
Switch(config-if)#switchport access vlan vlan number
Switch(config-if)#switchport mode access
Switch(config-if)#no cdp enable
Switch(config-if)#spanning-tree portfast
Switch(config-if)#spanning-tree bpdufilter enable
Switch(config-if)#spanning-tree bpduguard enable
```

- Enable Root Guard on the Catalyst 3750 interfaces to which the cell ring is connected. This prevents a directly or indirectly connected STP-capable device from affecting the Catalyst 3750 being the root bridge:

```
Switch(config)#interface type/slot port
Switch(config-if)# spanning-tree guard root
```

- Configure the VTP domains appropriately or turn off VTP altogether if you want to limit or prevent possible undesirable protocol interactions with regard to network-wide VLAN configuration. This precaution can limit or prevent the risk of an administrator error propagating to the entire network, and the risk of a new switch with a higher VTP revision overwriting by accident the VLAN configuration of the entire domain.

```
Switch(conf)#vtp mode transparent
```

- By default, all the LAN ports on all the Catalyst switches are configured as "untrusted". This prevents attached devices from manipulating QoS values inappropriately. In the EttF 1.1 design, only the trunk ports are recommended to be set as "trusted" if QoS is ever implemented in the network. All the access ports (for example, those on the Catalyst 2955) should remain "untrusted."

```
Switch(conf)#interface type/slot port
Switch(config-if)#no mls qos trust
```

- Disable unused ports and put them in an unused VLAN. By not granting connectivity or by placing a device into a VLAN not in use, unauthorized access can be prevented by fundamental physical and logical barriers.

```
Switch(conf)#interface type/slot port
Switch(config-if)#shutdown
```

Spanning Tree Protocol Security

STP is a useful protocol, but it does not implement any authentication and encryption to protect the exchange of Bridge Protocol Data Units (BPDUs). Because of the lack of authentication, anyone can speak to an STP-enabled device. An attacker could very easily inject fraudulent BPDUs, triggering a topology recalculation. A forced change to the STP topology could lead to a DoS condition, or leave the attacker as a man-in-the-middle. In addition, because BPDUs are not encrypted, it is fairly simple to intercept BPDUs in transit, revealing important topology information.

Catalyst 3750 and 2955 Series switches support a set of features that help protect bridged networks using the Spanning Tree Protocol. The following are the recommended best practices:

- Disable VLAN auto-negotiated trunking on user ports
- Disable unused ports and put them into an unused VLAN (as explained in the previous section)

- Use Per-VLAN Spanning Tree (PVST)
- Implement Port Security (as explained in a subsequent section)
- Configure BPDU Guard
- Configure STP Root Guard

Disabling Auto-negotiated Trunking

By default, all Ethernet ports on Catalyst switches are set to auto-negotiated trunking mode, which allows switches to automatically negotiate ISL and 802.1Q trunks. The negotiation is managed by Dynamic Trunking Protocol (DTP). Setting a port to auto-negotiated trunking mode makes the port willing to convert the link into a trunk link, and the port becomes a trunk port if the neighboring port is set as a trunk or configured in desirable mode.

Although the auto-negotiation of trunks facilitates the deployment of switches, somebody can take advantage of this feature and easily set up an illegitimate trunk. For this reason, auto-negotiation trunking should be disabled on all ports connecting to end users.

To disable auto-negotiated trunking, use the **switchport mode access** command. Setting the port mode to **access** makes the port a nontrunking, nontagged single VLAN Layer 2 interface. The following example shows how to set a port as nontrunking, nontagged single-VLAN Layer-2:

```
Switch(config)# interface type slot/port
Switch(config-if)# switchport mode access vlan 10
Switch(config-if)#
```

BPDU Guard

BPDU Guard is a feature that prevents a host port from participating in spanning tree. Under normal circumstances, Layer 2 access ports connected to a single workstation or server should not participate in spanning tree. When enabled on a port, BPDU Guard shuts down the port as soon as a BPDU is received in that port. In this way, BPDU Guard helps prevent unauthorized access and the illegal injection of forged BPDUs.

BPDU Guard requires STP PortFast to be configured on the port first. STP PortFast causes a Layer 2 LAN port configured as an access port to enter the forwarding state immediately, bypassing the listening and learning states. PortFast can be used on Layer 2 access ports connected to a single workstation or server to allow those devices to connect to the network immediately, instead of waiting for STP to converge.

BPDU can be configured per port or globally. When configured globally, BPDU Guard is effective only on ports in the operational PortFast state.

To enable BPDU Guard on an interface, use the **spanning-tree bpduguard** command. Make sure to first enable PortFast on the port.

```
Switch(config)# interface type/slot port
Switch(config-if)# spanning-tree portfast
Switch(config-if)# spanning-tree bpduguard enable
```

BPDU Guard can be globally enabled on systems running Cisco IOS by using the **spanning-tree portfast bpduguard default** command. When enabled globally, BPDU Guard applies to all interfaces that are in an operational PortFast state:

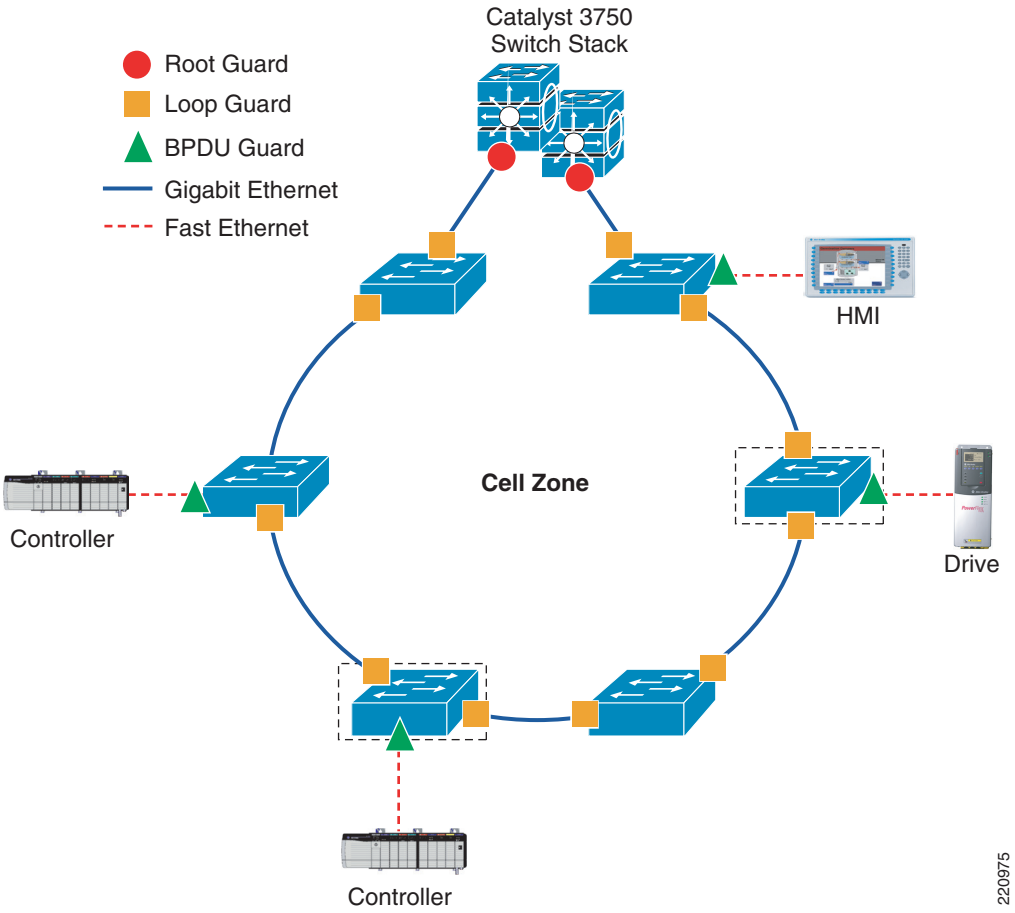
```
Switch(config)# spanning-tree portfast bpduguard
```

STP Root Guard

STP Root Guard is a feature that enforces the placement of the root bridge. STP Root Guard is a feature that is enabled on selected ports to prevent surrounding switches from becoming the root switch. The Root Guard feature forces a port to become a designated port so that no switch on the other end of the link can become a root switch. If a port configured for Root Guard receives a superior BPDU, the port immediately goes into a root-inconsistent (blocked) state. In this way, STP Root Guard blocks other devices trying to become the root bridge by sending superior BPDUs.

Figure 5-1 illustrates the placement of the STP Root Guards in the ring topology.

Figure 5-1 Placement of STP Guards



Note Do not enable Loop Guard and Root Guard on a port at the same time. Root Guard forces a port to always be designated as the root port. Loop Guard is effective only if the port is a root port or an alternate port.

To enable STP Root Guard on an interface, use the **spanning-tree guard root** command. Make sure to first enable PortFast on the port. The following example shows how to enable STP Root Guard on an interface:

```
Switch(config)# interface type/slot port
Switch(config-if)# spanning-tree guard root
```

MAC Flooding Attack

All switches have a finite hardware learning table to store the source addresses of all received packets; when this table becomes full, the traffic that is directed to addresses that can no longer be learned is permanently flooded. Packet flooding, however, is constrained within the VLAN of origin; therefore, no VLAN hopping is permitted.

One corner case behavior can be exploited by a malicious user that wants to turn the switch to which the user is connected into a dumb pseudo-hub and sniff all the flooded traffic. On non-intelligent switches, this problem arises because the L2 identity of a sender is not checked; therefore, the sender is allowed to impersonate an unlimited number of devices simply by counterfeiting packets. Cisco switches support a variety of features whose only goal is to identify and control the identities of connected devices. The security principle on which they are based is very simple: authentication and accountability are critical for all untrusted devices, including PACs, I/Os, drives, and human-machine interfaces (HMIs) attached to a switch in the cell network.

Port Security can be used to constrain the connectivity of a device. With Port Security, preventing any MAC flooding attack becomes as simple as limiting the number of MAC addresses that can be used by a single port; the identification of the traffic of a device is thereby directly tied to its port of origin.

In EttF 1.1, Cisco recommends enabling Port Security on an access port (not a trunk port that is used to form the L2 backbone network) and to set the maximum number of secure addresses to 1. The violation mode is the default; no static secure MAC addresses are configured.

```
Switch(config)# interface type/slot port
Switch(config-if)# switchport mode access vlan vlan number
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 1
Switch(config-if)# switchport port-security violation restrict
Switch(config-if)# end
```

It is a security violation when one of the following situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.

Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. In this mode, you are notified that a security violation has occurred. Specifically, an SNMP trap is sent, a syslog message is logged, and the violation counter increments.

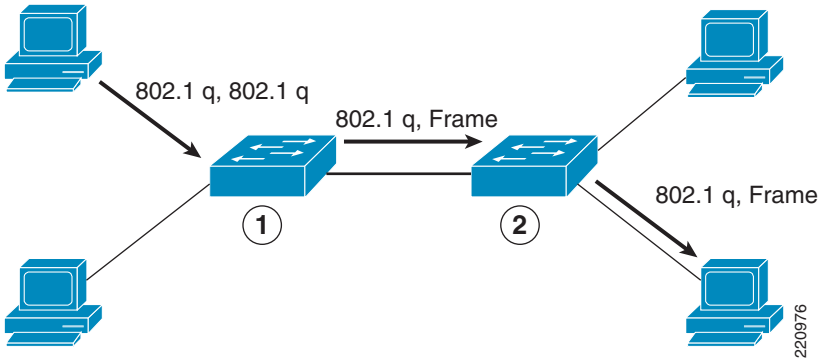
More information about the feature can be found at the following URL: http://www.cisco.com/en/US/docs/switches/lan/catalyst3750/software/release/12.2_40_se/configuration/guide/swtrafc.html.

VLAN Hopping

Tagging attacks are malicious schemes that allow a user on a VLAN to get unauthorized access to another VLAN. For example, if a switch port is configured as DTP auto and receives a fake DTP packet, it might become a trunk port and it might start accepting traffic destined for any VLAN. Therefore, a malicious user can start communicating with other VLANs through that compromised port.

Another version of this network attack is called double tagging, and involves tagging the transmitted frames with two 802.1q headers to forward the frames to the wrong VLAN (see Figure 5-2).

Figure 5-2 VLAN Hopping with Double-Encapsulated 802.1q Traffic



The first switch to encounter the double-tagged frame (1) strips the first tag off the frame and forwards the frame. The result is that the frame is forwarded with the inner 802.1q tag out all the switch ports (2), including trunk ports configured with the native VLAN of the network attacker. The second switch then forwards the packet to the destination based on the VLAN identifier in the second 802.1q header.

VLAN hopping attack can be prevented by setting DTP to "off" on all non-trusted ports:

- If you do not intend to trunk across those links, use the **switchport mode access interface** configuration command to disable trunking.

```
Switch(config)# interface type/slot port
Switch(config-if)# switchport mode access
```

- To enable trunking to a device that does not support DTP, use the **switchport mode trunk** and **switchport nonegotiate** interface configuration commands to cause the interface to become a trunk but to not generate DTP frames.

```
Switch(config)# interface type/slot port
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport nonegotiate
```

Sometimes, even when simply receiving regular packets, a switch port may behave like a full-fledged trunk port (for example, accepting packets for VLANs different from the native), even if it is not supposed to do so. This is commonly referred to as "VLAN leaking". Fortunately, the Catalyst switches have been designed in their hardware and software to always enforce proper traffic classification and isolation on all their ports.

ARP Spoofing Attack

Address Resolution Protocol (ARP) is used to map IP addressing to MAC addresses in a LAN segment where hosts of the same subnet reside. Normally, a host broadcasts an ARP request to find the MAC address of another host with a particular IP address, and an ARP response comes back from the host whose address matches the request. The requesting host then caches this ARP response. Within the ARP protocol, another provision is made for hosts to perform unsolicited ARP replies. The unsolicited ARP replies are called gratuitous ARPs (GARPs). GARPs can be exploited maliciously by an attacker to spoof the identity of an IP address on a LAN segment. Typically, this is used to spoof the identity between two hosts or all traffic to and from a default gateway in a man-in-the-middle attack.

By crafting an ARP reply, a network attacker can make their system appear to be the destination host sought by the sender. The ARP reply causes the sender to store the MAC address of the system of the network attacker in the ARP cache. This MAC address is also stored by the switch in its CAM table. In

this way, the network attacker has inserted the MAC address of their system into both the CAM table of the switch and the ARP cache of the sender. This allows the network attacker to intercept frames destined for the host being spoofed.

The use of DHCP snooping along with Dynamic ARP Inspection (DAI) mitigates various ARP-based network exploits. These Catalyst features validate ARP packets in a network and permit the interception, logging, and discarding of ARP packets with invalid MAC address to IP address bindings.

DHCP snooping provides security by filtering trusted DHCP messages and then using these messages to build and maintain a DHCP snooping binding table. DHCP snooping considers DHCP messages originating from any user-facing port that is not a DHCP server port or an uplink to a DHCP server as untrusted. From a DHCP snooping perspective, these untrusted, user-facing ports should not send DHCP server-type responses such as DHCPOffer, DHCPAck, or DHCPNak.

The DHCP snooping binding table contains the MAC address, IP address, lease time, binding type, VLAN number, and interface information corresponding to the local untrusted interfaces of a switch; it does not contain information regarding hosts interconnected with a trusted interface. An untrusted interface is an interface configured to receive messages from outside the network or firewall. A trusted interface is an interface that is configured to receive only messages from within the network. The DHCP snooping binding table can contain both dynamic as well as static MAC address to IP address bindings.

DAI determines the validity of an ARP packet based on the valid MAC address to IP address bindings stored in a DHCP snooping database. Additionally, DAI can validate ARP packets based on user-configurable ACLs. This allows for the inspection of ARP packets for hosts using statically configured IP addresses. DAI allows for the use of per-port access control lists (PACLs) and VLAN access control lists (VACLs) to limit ARP packets for specific IP addresses to specific MAC addresses.

```
Switch(config)# ip dhcp snooping
Switch(config)# ip dhcp snooping vlan vlan_id
Switch(config)# ip arp inspection vlan vlan_id
Switch(config)# ip arp inspection validates src-mac dst-mac ip
Switch(config)# interface type slot/port
Switch(config-if)# ip dhcp snooping trust
Switch(config-if)# ip dhcp snooping limit rate rate
Switch(config-if)# ip arp inspection trust
```

DHCP Attacks

There are two common types of DHCP attacks: DHCP starvation attack and rogue DHCP server attack.

A DHCP starvation attack works by broadcasting DHCP requests with spoofed MAC addresses. This is easily achieved with attack tools such as Gobbler. If enough requests are sent, the network attacker can exhaust the address space available to the DHCP servers for a period of time. The attack can be mitigated by configuring Port Security on the Catalyst switch as described in MAC Flooding Attack, page 5-16.

In a rogue DHCP server attack, the attacker sets up a rogue DHCP server on their system and responds to new DHCP requests from clients on the network. The network attacker can provide clients with addresses and other network information. Because DHCP responses typically include default gateway and DNS server information, the network attacker can supply their own system as the default gateway and DNS server, resulting in a man-in-the-middle attack.

Use the following commands to mitigate these attacks:

```
Switch(config)#ip dhcp snooping
Switch(config)#ip dhcp snooping vlan vlan number
Switch(config)#ip dhcp snooping information option
```

Security Design for the Manufacturing Zone

Because the security design strategy of the manufacturing zone is identical to that of the enterprise campus network, this section simply provides description of the required best practices. References are provided for their detailed implementation.

Security Design for the Catalyst 3750 Series Switch That Aggregates Cell/Area Zone Networks

Note the following:

- Device hardening (see Network Device Hardening, page 5-3)
- Layer 2 security for L2 ports (see Layer 2 Security Design, page 5-11)
- Ingress/egress filtering—RFC 1918 and RFC 2827 filtering should be implemented to protect against spoofed denial-of-service (DoS) attacks (http://www.cisco.com/en/US/tech/tk59/technologies_white_paper09186a0080174a5b.shtml).
- Routing protocol authentication—This is to prevent an attacker from sharing incorrect routing information between a rogue router and a valid one. The intent of the attack is to trick the router into not only sending data to the incorrect destination but also possibly to put it out of service. The recommended method is to check the integrity of routing updates by authentication using MD5-HMAC. See the following URLs:
  - Configuring EIGRP Authentication—http://www.cisco.com/en/US/tech/tk365/technologies_configuration_example09186a00807f5a63.shtml
  - Configuring IS-IS Authentication—http://www.cisco.com/en/US/tech/tk365/technologies_configuration_example09186a0080093f36.shtml
  - Configuring OSPF Authentication—http://www.cisco.com/en/US/tech/tk365/technologies_configuration_example09186a0080093f36.shtml

Security Design for the Catalyst 4500 Series Switch for the Core of the Control Network

The Catalyst 4500 is assumed to provide only L3 routing. Thus, only device hardening and L3 security best practices are needed. Note the following:

- Device hardening (see Network Device Hardening, page 5-3)
- Ingress/egress filtering—RFC 1918 and RFC 2827 filtering should be implemented to protect against spoofed DoS attacks (http://www.cisco.com/en/US/tech/tk59/technologies_white_paper09186a0080174a5b.shtml).
- Router with ACL—The Catalyst 4500 should be configured to block traffic flows at L3/L4 based on your trust domains and security policies.

- Routing protocol authentication—This is to prevent an attacker from sharing incorrect routing information between a rogue router and a valid one. The intent of the attack is to trick the router to not only send data to the incorrect destination but to also possibly put it out of service. The recommended method is to check the integrity of routing updates by authentication using MD5-HMAC.
  - Configuring EIGRP Authentication—http://www.cisco.com/en/US/tech/tk365/technologies_configuration_example09186a00807f5a63.shtml
  - Configuring IS-IS Authentication—http://www.cisco.com/en/US/tech/tk365/technologies_configuration_example09186a0080093f36.shtml
  - Configuring OSPF Authentication—http://www.cisco.com/en/US/tech/tk365/technologies_configuration_example09186a0080093f36.shtml
- Control Plane Policing (CoPP)—This feature protects the CPU from unnecessary or DoS traffic by giving priority to important control plane and management traffic. The idea is to protect most of the CPU bound traffic and ensure routing stability, reachability, and packet delivery. Most importantly, CoPP is often used to protect the CPU from the DoS attack. For more information, see the following URL: http://www.cisco.com/en/US/products/hw/switches/ps4324/products_configuration_guide_chapter09186a008062ce7b.html.

Security Design for the Catalyst 3750 Series Switch in the Server Farm

This Catalyst 3750 switch provides VLAN separation for different servers for network services, network management, and site manufacturing operations and control. Because the Catalyst 4500 performs routing for these VLANs, this Catalyst 3750 provides only Layer 2 switching. Thus, its security protection simply includes device hardening and L2 security (see Network Device Hardening, page 5-3 and Layer 2 Security Design, page 5-11).

Security Protection for Servers

The servers that provide network services, network management, or site manufacturing operations and control should be provided at least with the following security protection:

- Reusable passwords—Users likely authenticate to their systems with username and passwords.
- Session-application crypto—Any communication between a client to a server considered sensitive (based on your policy) should be cryptographically protected with session-application crypto.
- OS/application hardening—Harden the OS and any application. Do not simply deploy every patch as it is released. Use some mechanism to do testing on updates before applying to production systems. Also, make sure to follow hardening guides for popular applications, such as Microsoft Internet Information Server (IIS) and Apache web server, used on the servers.
- Partitioning disk space—In the event of a problem, you do not want one rogue process to consume the entire disk space of the server. In Unix, for example, it is good practice to set aside separate partitions for the following components: /, /var, /home, /usr, and /tmp.
- Turning off unneeded services —If the host is a standard desktop, it probably does not need to run any services for other users such as FTP. If it is a server, the running services should be limited to those that are required to perform the job of the server. For example, this means running HTTP but not Telnet on a web server.
- Deploying the Cisco Security Agent (CSA)—The CSA protects critical servers by being a host-based IDS to help mitigate local attacks. See Endpoint Protection with Cisco Security Agent, page 5-33.

Security Design for the Demilitarized Zone

In the design of the industrial Ethernet network, one of the critical elements is to ensure the separation between the control network and enterprise network. In terms of the Purdue Reference Model, this is the separation between levels 1–3 and levels 4–5. This separation is necessary because real-time availability and security are the critical elements for the traffic in the control network. You do not want enterprise traffic that has very different traffic characteristics to enter the control network and cause any disruption to the ongoing operations. Acting as a firewall, the Cisco ASA5500 provides this separation of the two networks.

Servers that users from both networks need to access are put in a separate demilitarized zone (DMZ) network that is connected to the same firewall. To provide more granular network access, the Cisco ASA provides authentication, authorization, and accounting (AAA) services by working in conjunction with the CiscoSecure Access Control Server (ACS). This provides a user database of which the Cisco ASA can inquire to identify and validate before permitting the transmission of traffic to the destination network.

In addition to controlling traffic access between the three networks, the Cisco ASA can optionally be installed with the Cisco Adaptive Inspection Prevention Security Services Module (AIP-SSM) to provide intrusion detection or intrusion protection to prevent network attacks to those destinations to which the firewall function of the Cisco ASA permits network access.

Finally, all the servers placed in the DMZ need to be secured. See Security Protection for Servers, page 5-21.

Security Levels on the Cisco ASA Interfaces

The Cisco ASA uses the concept of assigning security levels to its interfaces. The higher the security level, the more secure an interface is. The security level is thus used to reflect the level of trust of this interface with respect to the level of trust of another interface on the Cisco ASA.The security level can be between 0 and 100. The most secure network is placed behind the interface with a security level of 100. The security level is assigned by using the **security-level** command.

In the EttF 1.1 solution, Cisco recommends creating three networks in different security levels, as shown in Table 5-3.

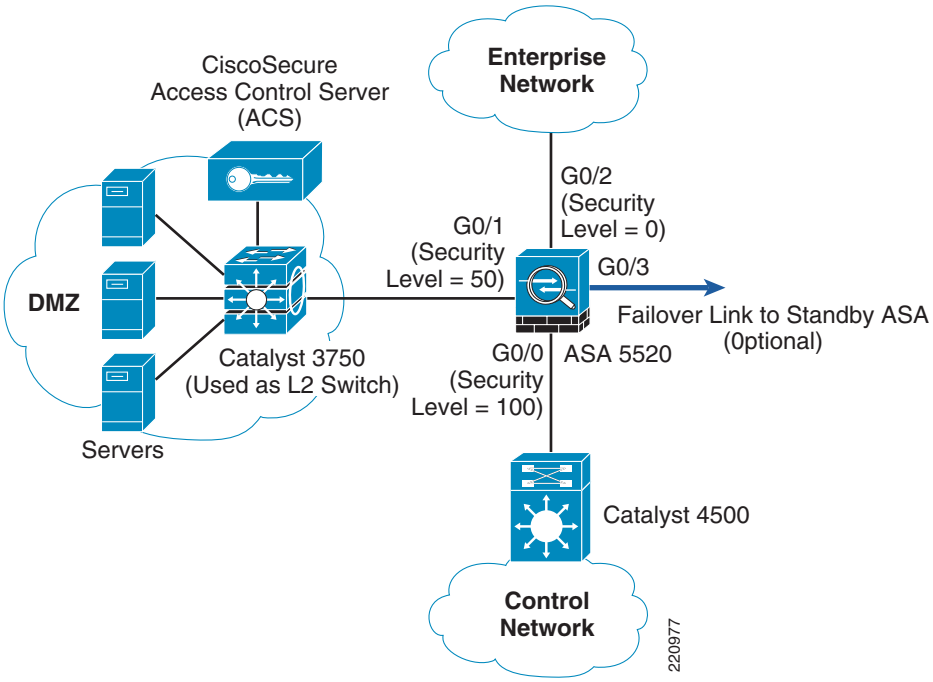
Table 5-3 Network Security Levels

Network	Security Level	Interface (see Figure 5-3)
Enterprise network	0	G0/2
DMZ	50	G0/1
Control network	100	G0/0

Configuration Example

Refer to Figure 5-3 for the subsequent configuration example.

Figure 5-3 Security Levels on the Interfaces of the Cisco ASA 5500



Based on the security level recommendations above, the following shows how to configure the levels on the interfaces of the Cisco ASA 5520 platform:

- GigabitEthernet 0/0 is the interface connected to the control network. It is named *inside*. Because it is at security level 100, it has the highest security level.

```
interface GigabitEthernet0/0
nameif inside
security-level 100
ip address 10.18.1.1 255.255.255.0
```

- GigabitEthernet 0/1 is the interface connected to the control network. It is named *outside* with security level set to 0.

```
interface GigabitEthernet0/1
nameif outside
security-level 0
ip address 10.13.2.1 255.255.255.248
```

- GigabitEthernet 0/2 is the interface connected to the DMZ. It is named *DMZ* with security level 50.

```
interface GigabitEthernet0/2
nameif dmz
security-level 50
ip address 10.19.2.9 255.255.255.248
```

The command **nameif** is used to assign a name to an interface. This interface name is used to set up any configuration feature associated to the given interface.

Note that the **ip address** configuration includes an optional parameter **standby**. It is used for configuring the standby Cisco ASA in the solution.

By default, the ASA 5500 implicitly permits traffic that enters the ASA via a high security level interface and leaves via a low security level interface, but the appliance implicitly denies traffic in the reverse direction. However, the EttF 1.1 solution recommends that traffic be denied going from the control network (security level 100) to the enterprise network (security level 0). An ACL needs to be explicitly configured to meet this access policy.

Stateful Packet Filtering

The Cisco ASA in the DMZ between the control network and enterprise network enables the operator to define policies and rules that identify what traffic should be permitted in or out of an interface. It uses ACLs to drop unwanted or unknown traffic when it attempts to enter the trusted networks.

An ACL, starting with a keyword **access-list**, is a list of security rules and policies grouped together that allows or denies packets after looking at the packet headers and other attributes. Each permit or deny statement can classify packets by inspecting up to Layer 4 headers for a number of parameters:

- Layer 2 protocol information such as EtherTypes
- Layer 3 protocol information such as ICMP, TCP, or UDP
- Source and destination IP addresses
- Source and destination TCP or UDP ports

After an ACL has been properly configured, it can be applied to an interface to filter traffic with the keyword **access-group**. The Cisco ASA can filter packets in both the inbound and outbound direction on an interface. When an inbound ACL is applied to an interface, the security appliance inspects against the ACL parameters after receiving or before transmitting them. An incoming packet is screened in the following sequence:

1. If this packet matches with an existing connection in the firewall connection table, it is allowed in. If it does not, go to Step 2.
2. The firewall tries to match the packet against the ACLs sequentially from the top to the bottom. After the first matched ACL is identified, the packet is allowed in or dropped according to the action (permit or deny). If there is no match, go to Step 3.
3. The security appliance drops all traffic that does not match any parameter defined in the ACL. There is an implicit deny at the end of all ACLs.

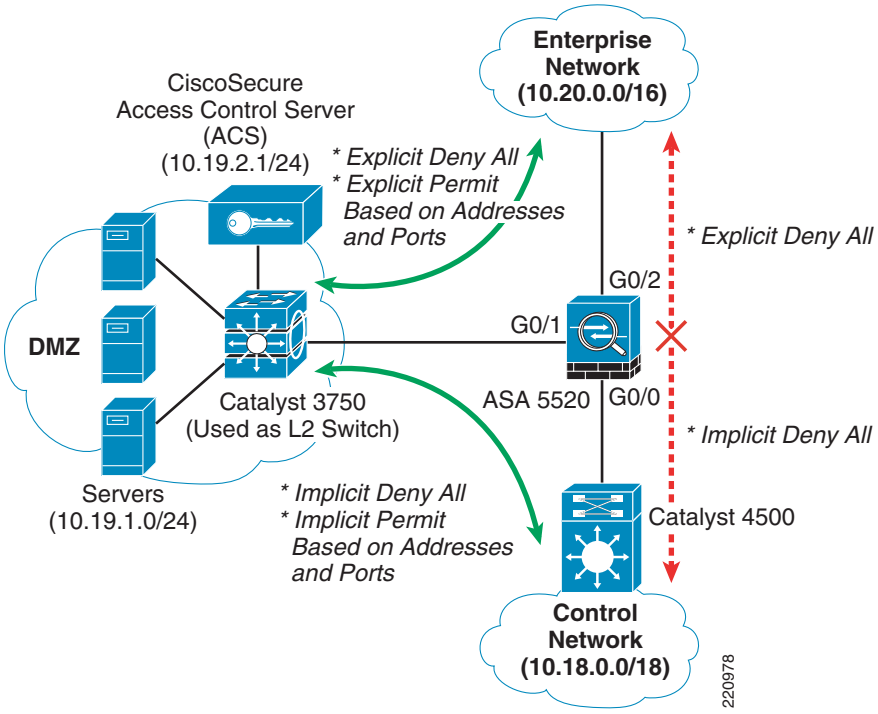
Note The interface ACL does not block packets destined for the IP addresses of the security appliance.

For the EttF 1.1 solution, general packet filtering recommendations are listed in Table 5-4 and shown in Figure 5-4.

Table 5-4 Packet Filtering Recommendations

		Traffic Source		
		Enterprise Network	**DMZ**	**Control Network**
Traffic Destination	**Enterprise Network**	N/A	Explicitly permitted by ACLs	Disallowed (explicitly denied by ACLs)
	DMZ	Explicitly permitted by ACLs	N/A	Explicitly permitted by ACLs
	Control Network	Disallowed (implicitly denied by ACLs)	Explicitly permitted by ACLs	N/A

Figure 5-4 High-Level Packet Filtering Recommendations for the DMZ between the Control and Enterprise Networks



Configuration Example

See Table 5-5 for an example for ingress ACLs applied to the control network-facing interface.

Table 5-5 Configuration Example for Ingress ACLs on the Control Networking-Facing Interface

Applied To Interface	Traffic Direction	Permitted Traffic Types (Source to Destination)

Interface connected to the control network (*inside*)	Inbound	• HTTP (servers in the control network to servers in DMZ)
		`access-list inside extended permit tcp 10.18.0.0 255.255.0.0 10.19.0.0 255.255.255.0 eq www`
		• HTTPS (any in the control network to servers in DMZ)
		`access-list inside extended permit tcp 10.18.0.0 255.255.0.0 10.19.0.0 255.255.255.0 eq https`
		• Telnet (any in the control network to host 10.19.1.10 in the DMZ)
		`access-list inside extended permit tcp 10.18.0.0 255.255.0.0 host 10.19.2.1 eq telnet`
		• ICMP (any in the control network to servers in the DMZ)
		`access-list inside extended permit icmp 10.18.0.0 255.255.0.0 10.19.2.0 255.255.255.0`
		• Explicitly deny other traffic types to anywhere (i.e. DMZ and enterprise networks)
		`access-list inside deny 10.18.0.0 255.255.0.0`
		• Apply the ACLs above to the ingress side of the control network-facing interface
		`access-group inside in interface inside`

See Table 5-6 for an example for ingress ACLs applied to the DMZ-facing interface.

Table 5-6 Configuration Example for Ingress ACLs on the DMZ -Facing Interface

Applied To Interface	Traffic Direction	Permitted Traffic Types (Source to Destination)

Interface connected to the DMZ (*dmz*)	Inbound	• Telnet (servers in the DMZ to the control and enterprise networks)
		`access-list dmz extended permit tcp 10.19.1.0 255.255.255.0 10.18.0.0 255.255.0.0 eq telnet`
		• HTTP (servers in the DMZ to the control and enterprise networks)
		`access-list dmz extended permit tcp 10.19.1.0 255.255.255.0 10.18.0.0 255.255.0.0 eq www`
		• HTTPS (servers in the DMZ to the control and enterprise networks)
		`access-list dmz extended permit tcp 10.19.1.0 255.255.255.0 10.18.0.0 255.255.0.0 eq https`
		• ICMP (servers in the DMZ to the control and enterprise networks)
		`access-list dmz extended permit icmp 10.19.1.0 255.255.255.0 10.18.0.0 255.255.0.0`
		`access-list dmz extended permit icmp 10.19.1.0 255.255.255.0 10.20.0.0 255.255.0.0`
		• Explicitly deny other traffic types to anywhere
		`access-list inside deny 10.19.0.0 255.255.0.0`
		• Apply the ACLs above to the ingress side of the DMZ-facing interface
		`access-group dmz in interface inside`

See Table 5-7 for the example for ingress ACLs applied to the enterprise network-facing interface.

Table 5-7 Configuration Example for Ingress ACLs on the Enterprise Networking-Facing Interface

Applied To Interface	Traffic Direction	Permitted Traffic Types (Source to Destination)

Interface connected to the enterprise network (*outside*)	Inbound	• Telnet (any in the enterprise network to the DMZ [10.19.0.0/16])
		`access-list outside extended permit tcp 10.20.0.0 255.255.0.0 10.19.1.0 255.255.255.0 eq telnet`
		• HTTP (any in the enterprise network to the DMZ [10.19.0.0/16])
		`access-list outside extended permit tcp 10.20.0.0 255.255.0.0 10.19.1.0 255.255.255.0 eq www`
		• HTTPS (any in the enterprise network to the DMZ [10.19.0.0/16])
		`access-list outside extended permit tcp 10.20.0.0 255.255.0.0 10.19.1.0 255.255.255.0 eq https`
		• Explicitly deny other traffic types to anywhere
		`access-list inside deny 10.20.0.0 255.255.0.0`
		• Apply the ACLs above to the ingress side of the enterprise network-facing interface
		`access-group outside in interface inside`

Authenticating Firewall Sessions for User Access to Servers in the DMZ

When users in the control network or enterprise network want to access servers in the DMZ, the best practice is to enable authentication on the Cisco ASA. This involves validating the users based on their identity and predetermined credentials, such as passwords. The Cisco ASA can be configured to maintain a local user database or to use an external server for authentication. To communicate with an external authentication server, the Cisco ASA supports various protocols such as RADIUS, TACACS+, RSA SecurID, Windows NT, Kerberos, and LDAP.

The following steps show how the Cisco ASA authenticates an HTTP session originated from the enterprise network before the Cisco ASA permits the session to access the web server in the DMZ:

1. The user on the outside of the Cisco ASA attempts to create an HTTP connection to the web server behind the ASA in the DMZ.
2. The Cisco ASA prompts the user for authentication.
3. The Cisco ASA receives the authentication information (userid and password) from the user and sends an AUTH Request to the CiscoSecure ACS.

4. The server authenticates the user and sends an AUTH Accept message to the Cisco ASA.
5. The Cisco ASA allows the user to access the web server.

Note For more details of the Cisco ACS, see the following URL: http://www.cisco.com/en/US/products/sw/secursw/ps2086/products_configuration_guide_book09186a0080721d25.html

Configuration Example

The following example illustrates how to use firewall session authentication in a plant floor network. Factory XYZ wants to define the following policies on the ASA to specify which source addresses have rights to access to a server at 10.18.1.2 in the DMZ:

- Any user in the enterprise network can access the server at 10.18.1.2. The permitted protocols are HTTP and HTTPS.
- Only users in the 10.17.0.0/16 subnets in the control floor can access the server. The permitted protocols are Telnet, HTTP, and HTTPS.

The users residing in these legitimate addresses are required for authentication before reaching out to the server.

Step 1 Define an AAA server group named *ETTF2* using TACACS+ as the protocol for authentication. This AAA server is at 10.19.2.11.

```
aaa-server ETTF2 protocol tacacs+
aaa-server ETTF2 host 10.19.2.11
key Cisco
```

Step 2 Add the Cisco ASA as an AAA client in the CiscoSecure ACS.

Step 3 Create an ACL named *INSAUTH* that requires authentication of HTTP and HTTPS traffic.

```
access-list INSAUTH extended permit tcp 10.17.0.0 255.0.0.0 host 10.18.1.2 eq telnet
access-list INSAUTH extended permit tcp 10.17.0.0 255.0.0.0 host 10.18.1.2 eq www
access-list INSAUTH extended permit tcp 10.17.0.0 255.0.0.0 host 10.18.1.2 eq 8080
```

Step 4 Define the AAA match command to match the source and destination addresses of the incoming Telnet, HTTP, and HTTPS traffic from the control network (*inside*) against the ACL group *INSAUTH*.

```
aaa authentication match INSAUTH inside ETTF2
```

Step 5 Create ACLs named *OUTAUTH* that require authentication of HTTP and HTTPS traffic.

```
access-list OUTAUTH extended permit tcp any host 10.18.1.2 eq www
access-list OUTAUTH extended permit tcp any host 10.18.1.2 eq 8080
```

Step 6 Define the AAA match command to match the source and destination addresses of the incoming HTTP and HTTPS traffic from the enterprise network (*outside*) against the ACL group *OUTAUTH*.

```
aaa authentication match OUTAUTH outside ETTF2
```

Step 7 Define the AAA match command to match the source and destination addresses of the incoming HTTP and HTTPS traffic.

If there is an ACL without authentication, the firewall session authentication can be customized in the following ways:

- Authentication exception based on users
- Authentication timeouts
- Customization of authentication prompts

Integrating the ASA 5500 Appliance with the Adaptive Inspection Prevention Security Services Module

The Cisco ASA supports the Adaptive Inspection Prevention Security Services Module (AIP-SSM) running the Cisco Intrusion Prevention System (CIPS) software. Although the Cisco ASA can also provide IPS support with the **ip audit** command if an AIP-SSM module is absent, it supports only a limited number of signatures compared to the module. Also, these built-in signatures are not upgradeable.

Note For details on how to upgrade the image or signatures of the module, see the following URL: http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/products_configuration_guide_chapter09186a00807517ba.html.

Note The Cisco ASA 5520, which is the ASA model recommended for the EttF 1.1 design, supports both the AIP-SSM10 and AIP-SSM20 modules.

Access to the AIP-SSM Module

An administrator can connect to the AIP-SSM module via the following:

- Telnet and SSH to the FastEthernet management interface port on the module
- Telnet and SSH to the FastEthernet management interface port on the ASA and then the **session** <*module-number*> command to the AIP-SSM module
- HTTPS to Adaptive Security Device Manager (ASDM) on the ASA

Note For the initialization and maintenance of the AIP-SSM module, see the ASA documentation at the following URL: http://www.cisco.com/en/US/products/ps6120/products_getting_started_guide_chapter09186a00806a8347.html.

Inline Versus Promiscuous Mode

The Cisco AIP-SSM supports both inline and promiscuous modes. In the inline mode, the module can be considered to be an intrusion protection system (IPS); in the promiscuous mode, it can be considered to be an intrusion detection system (IDS).

When configured as an inline IPS, the AIP-SSM module can drop malicious packets, generate alarms, or reset a connection, allowing the ASA to respond immediately to security threats and protect the network. Inline IPS configuration forces all traffic to be directed to the AIP-SSM. The ASA does not forward any traffic out to the network without the AIP-SSM first inspecting it.

Figure 5-5 shows the traffic flow when the Cisco ASA is configured in inline IPS mode.

Figure 5-5 Inline IPS Traffic Flow

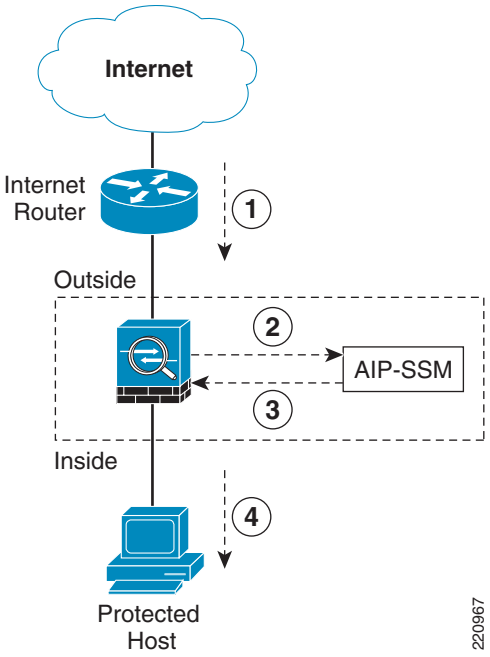


Figure 5-5 shows the following sequence of events:

1. The Cisco ASA receives an IP packet from the Internet.
2. Because the Cisco ASA is configured in inline IPS mode, it forwards the packet to the AIP-SSM for analysis.
3. The AIP-SSM analyzes the packet and, if it determines that the packet is not malicious, forwards the packet back to the Cisco ASA.
4. The Cisco ASA forwards the packet to its final destination (the protected host).

Note Inline IPS mode is the most secure configuration because every packet is inspected by the AIM-SSM. However, this may affect the overall throughput. The impact depends on the type of attack, signatures enabled on the system, and the amount of traffic passing through the application.

When the Cisco ASA is set up to use the AIP-SSM in promiscuous mode, the ASA sends a duplicate stream of traffic to the AIP-SSM. This mode has less impact on the overall throughput. Promiscuous mode is considered to be less secure than inline mode because the IPS module can only block traffic by forcing the ASA to shun the malicious traffic or send a TCP-RST (reset) message to terminate a TCP connection.

Note Promiscuous mode has less impact on performance because the AIP-SSM is not in the traffic path. A copy of the packet is sent to the AIM-SSM. If a packet is dropped, there is no effect on the ASA.

Figure 5-6 shows an example of how traffic flows when the AIP-SSM is configured in promiscuous mode.

Figure 5-6 Promiscuous Mode Traffic Flow

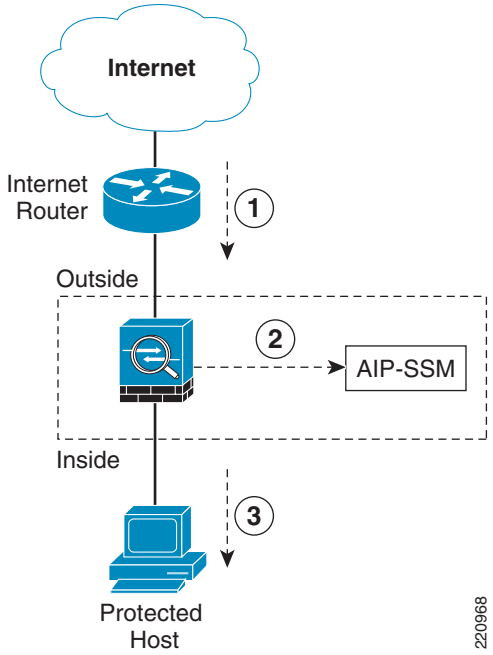


Figure 5-6 shows the following sequence of events:

1. The Cisco ASA receives an IP packet from the Internet.
2. Because the Cisco ASA is configured in promiscuous mode, the AIP-SSM silently snoops the packet.
3. The ASA forwards the packet to its final destination (the protected host) if the packet conforms to security policies; that is, if it does not match any of the configured signatures.

Note If the ASA firewall policies deny any inbound packet at the interface, the packet is not inspected by the AIM-SSM. This applies to both inline and promiscuous IPS modes.

Endpoint Protection with Cisco Security Agent

No security strategy can be effective if the servers and desktop computers (endpoints) are not protected. Endpoint attacks typically run in stages: probe, penetrate, persist, propagate, and paralyze. Most endpoint security technologies provide early stage protection (and then only when a signature is known).

The Cisco Security Agent (CSA) proactively defends against damage to a host throughout all stages of an intrusion, and is specifically designed to protect against new attacks where there is no known signature. The CSA goes beyond conventional endpoint security solutions by identifying and preventing malicious behavior before it can occur, thereby removing potential known and unknown security risks that threaten enterprise networks and applications.

When an application attempts an operation, the agent checks the operation against the security policy of the application. The agent makes a real-time “allow” or “deny” decision on its continuation and determines whether that request should be logged. Because protection is based on blocking malicious behavior, the default policies stop both known and unknown attacks without needing updates. Correlation is performed both at the agent and the management center console. Correlation at the agent results in dramatically increased accuracy, identifying actual attacks or misuse without blocking legitimate activity. Correlation at the management center identifies global attacks such as network worms or distributed scans.

Security Monitoring, Analysis, and Mitigation with CS-MARS

The Cisco Security Monitoring, Analysis, and Response System (CS-MARS) is an appliance-based, all-inclusive solution that allows network and security administrators to monitor, identify, isolate, and counter security threats. High-performance, scalable threat mitigation appliances fortify deployed network devices and security countermeasures by combining network intelligence with features such as ContextCorrelation, SureVector analysis, and AutoMitigate capability, empowering companies to readily identify, manage, and eliminate network attacks and maintain compliance.

Going beyond first- and second-generation security information management systems, CS-MARS more efficiently aggregates and reduces massive amounts of network and security data from popular network devices and security countermeasures. By gaining network intelligence, it effectively identifies network and application threats through sophisticated event correlation and threat validation. Verified attacks are visualized through an intuitive, detailed topology map to augment incident identification, investigation, and workflow. Upon attack discovery, the system allows the operator to prevent, contain, or stop an attack in real-time by pushing specific mitigation commands to network enforcement devices. The system supports customer-centric rule creation, threat notification, incident investigation, and a host of security posture and trend reports.

The entire solution is cost-effectively delivered in an appliance platform that affords low adoption costs and flexible use. CS-MARS appliances consist of standard Intel platforms with availability features accessible through a web-based user interface, hardened OS, embedded Oracle database, proprietary logic, and scalable architecture with various performance characteristics and price points to address a broad range of customer sizes and deployment scenarios.