Deploying the Cisco Catalyst 6500 Firewall Services Module in Transparent Mode

This chapter provides design and implementation recommendations for the use of firewall and load balancers in a data center to load balance and provide security services to web-based transactional applications (typically made of web servers, application servers, and data base servers), typical DMZ servers such as DNS servers and SMTP servers, and many more server types.

This chapter includes the following topics:

- Cisco Firewall Services Module Design Overview
- Configuration Details
- Configuring Redundancy
- Configuration Listings

Cisco Firewall Services Module Design Overview

This section includes the following topics:

- Transparent Firewalls
- Virtual Firewalls
- Routed Mode versus Bridge Mode
- Multicast Support
- Designs with FWSM and CSM
- Topology and Service Processing Sequence

The Cisco Firewall Services Module (FWSM) provides the following services for the server farm:

- ACL filtering—Inbound and outbound. In transparent mode in addition to standard and extended ACLs, the FWSM also supports Ethertype ACLs for non-IP traffic.
- Stateful inspection—The FWSM is a stateful device; it looks at the TCP connection establishment phase and does not let a segment pass until the TCP handshake occurs. The fixups complement this capability with specific application knowledge so that ports are opened dynamically based on the control protocol negotiation. Key fixups include the following: SMTP (also known as MailGuard), DNS (also known as DNSGuard), ICMP, RTSP, SQL*NET, SUN RPC, TFTP, UDP OraServ (Port 1525), and NetBIOS.

- Denial of service (DoS) protection—The FWSM protects against SYN floods with TCP Intercept (Release 1.1 to 2.2) and with SYN cookies (starting from Release 2.3).
- FragGuard—The FWSM protects against tiny fragment attacks. By default, the FWSM drops fragments, but many applications generate fragments. However, enabling fragment forwarding opens the door for fragment attacks such as those described in RFC1858. The FWSM provides protection against that type of attack by means of virtual fragment reassembly.
- TCP sequence randomization—Each TCP connection has two Initial Sequence Numbers (ISNs): one generated by the client and one generated by the server. The FWSM randomizes the ISN that is generated by the host/server on the higher security interface. At least one of the ISNs must be randomly generated so that attackers cannot predict the next ISN and potentially hijack the session. By using the FWSM, you can randomize the TCP sequence number of the servers, which prevents TCP session hijacking and also hides the OS fingerprint of the server.

The data center design with the Cisco Catalyst 6500 FWSM in transparent mode can be deployed with or without the Cisco Content Switching Module (CSM).

Transparent Firewalls

With FWSM Release 2.2 and later, the firewall can be configured as an OSI Layer 2 transparent bridge, which has the following important implications:

- Firewall interfaces are VLAN-based Layer 2 interfaces.
- The firewall bridges between the outside and inside VLANs. There is no other path between these two VLANs. With this design, the FWSM bridges traffic between the Multilayer Switch Feature Card (MSFC) interface (the gateway for the servers) and the servers. The MSFC has a VLAN interface with an IP address configured on the outside VLAN and the server ports are configured on the inside VLAN.
- Firewall protection is applied within the same subnet for traffic flowing from the outside interface to the inside interface.
- The gateway function for the servers is provided by the MSFC.

Note With FWSM 2.2 and 2.3, the transparent firewall supports only two interfaces. In transparent mode, the FWSM 2.2 or 2.3 does not support Network Address Translation (NAT) translations. NAT, if required, needs to be performed on an upstream device.

Rather than routing and Address Resolution Protocol (ARP) tables, a transparent firewall maintains a MAC address table, with each entry having an indicated interface. New command-line interface (CLI) commands have been introduced to manage this table: **mac-address-table** and **mac-learn**.

A transparent firewall typically has a single IP address that is not associated with an interface. The address is associated with the subnet and belongs to a specific firewall context (see the next section) and serves the following purposes:

- SSH or Telnet connection to the FWSM blade (or context if it belongs to a virtual context)
- Services such as authentication, authorization, and accounting (AAA), Websense, and Syslog that need to communicate with external servers using TCP/IP.
- Connectivity-related packets generated by the firewall, such as ping and ARP.

The servers are placed on the inside VLAN. The default gateway of the servers is the Hot Standby Router Protocol (HSRP) address of the MSFC. The MSFC forwards traffic to the servers by using ARP to find their IP address and rewriting the destination MAC address before sending the traffic to the FWSM, which is bridging traffic between outside and inside VLAN. FWSM bridges the traffic and while doing so applies the appropriate security policies.

As previously stated, a FWSM deployed in transparent mode bridges two interfaces. A typical data center uses the FWSM to protect multiple segments; a single FWSM can forward 5.5 Gbps of traffic, and considering the typical oversubscription levels of transactional applications, this is enough to aggregate traffic from multiple server farms.

Virtualization allows you to use a single FWSM as if you had multiple firewall appliances operating in bridge mode, with each one placed to protect a specific segment.

Virtual Firewalls

The virtual firewall feature available from FWSM 2.2 introduces the concept of multiple firewalls operating within the same hardware platform. Each virtual firewall exists in a separate *virtual context*, which includes the following:

- Collection of logical interfaces
- Security parameters for each interface
- Global data, state information, and statistics, which apply to the virtual firewall, accessed and referenced by a unique, virtual circuit identifier (VC ID)
- A configurable, enforced subset of the total system hardware resources

The virtual firewall feature also provides a global *system context* without firewall features, which exists outside of all other virtual firewall contexts. The global system context defines the settings for the entire FWSM blade. Specifically, it is used to define the individual virtual firewall contexts and all the physical interfaces for the platform. A Cisco IOS-like file system is used by the system context for storing information about each virtual firewall context. Failover is also defined in the system context.

The *administration context* has all the features of a regular virtual context but defines the interfaces used by the system context when loading a software image or configuration file from the TFTP server.

Virtual firewall contexts may define either Layer 2 transparent firewalls or Layer 3 routing firewalls. However, FWSM 2.2 and 2.3 require that *all* contexts on a single blade be one or the other: either all Layer 2 or all Layer 3.

Neither OSPF nor RIP is available on virtual Layer 3 firewalls. The FWSM does provide one routing table per context, however, but only with statically defined routes.

Failover between FWSM blades is supported using separate and identical active and standby service modules. Failover is not supported between virtual firewalls. Failover with FWSM 2.2 and 2.3 is strictly on a blade basis.

Up to 100 firewall contexts can be configured.

Routed Mode versus Bridge Mode

FWSM 2.2 Release and later supports two modes of operations: routed mode and transparent mode. In routed mode, the FWSM effectively routes traffic between interfaces. Each interface has its own IP address and the FWSM can perform static or dynamic routing (OSPF). In transparent mode, the FWSM bridges two VLANs: the outside and the inside VLANs. The FWSM in this case does not participate in any dynamic routing protocol but performs transparent bridging.

The design differences between the two modes of operations are as follows:

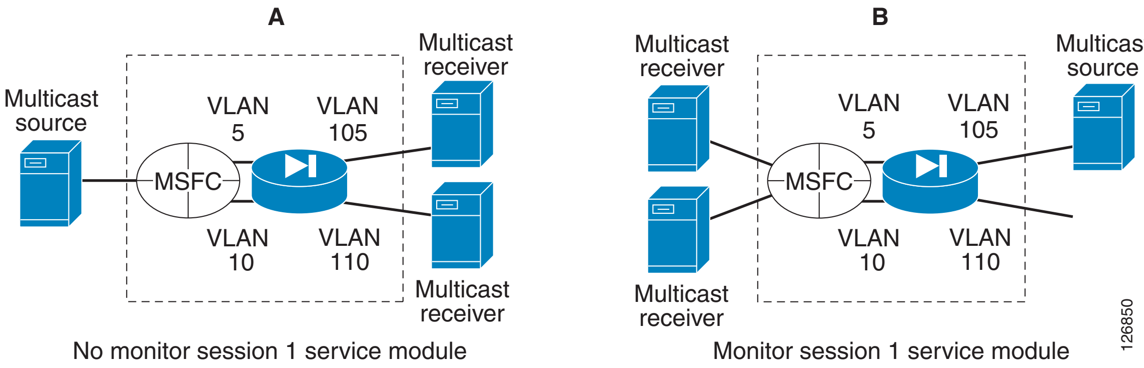
- Routed mode—On the MSFC, you configure static routes pushing traffic to the FWSM. These routes are typically redistributed into the dynamic routing protocol used by the enterprise and become external routes. The FWSM is configured to route traffic to the MSFC. In a server farm deployment, the FWSM in routed mode becomes the server default gateway. (Multicast traffic requires the multicast stub feature, which is not yet available). Each firewall context in routed mode supports 256 interfaces. When operating in routed mode, the FWSM can perform NAT.
- Transparent mode—The MSFC provides the Layer 3 interfaces (switched VLAN interface with an IP address) and the FWSM applies security functions in the path between the switch ports and the MSFC VLAN interface. No external route is injected in the routing protocol; the MSFC networks are advertised as part of the dynamic routing protocol operations. Each firewall context supports a maximum of two interfaces: outside and inside. When operating in transparent mode, the FWSM does not perform NAT. Any other Cisco IOS function supported by the MSFC is automatically supported in this design with the exception of multicast.

Multicast Support

Support for multicast sources protected by the FWSM requires using Cisco IOS as the software on the Catalyst 6500 with the SPAN reflector feature (monitor session servicemodule) enabled. Using CatOS on the Catalyst 6500 with the FWSM does not allow multicast switching in hardware for multicast sources protected by the FWSM.

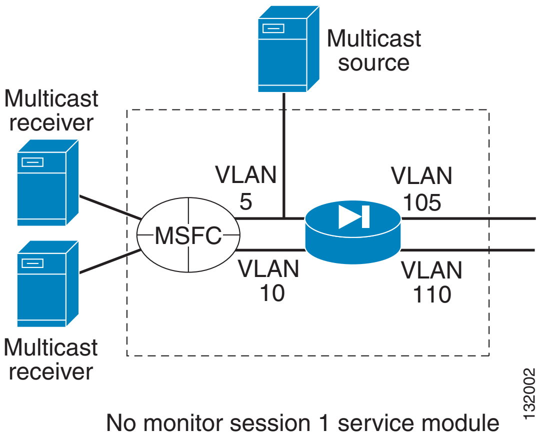
Figure 4-1 shows the placement of the multicast source, the placement of the receiver, and in which cases you need the SPAN reflector.

Figure 4-1 SPAN Reflector—When Needed and Not Needed



Using the SPAN reflector has some caveats in terms of performance, and it is also incompatible with bridging Bridge Protocol Data Units (BPDUs) through the FWSM. For these reasons, Cisco recommends, when possible, to place the multicast sources as close as possible to the MSFC, as indicated in Figure 4-2.

Figure 4-2 Alternative Design with Multicast Sources



Note An alternative and recommended solution is to force the FWSM to operate in bus mode using the **fabric switching-mode force bus** command. By using this command, the FWSM can support multicast sources on either the inside or the outside, BPDUs are bridged correctly, and cross-line card EtherChannels are supported. When the FWSM operates in bus mode, all traffic to and from the FWSM goes via the supervisor fabricand traffic from DFC-enabled line cards still uses the fabric connection.

Designs with FWSM and CSM

The Catalyst 6500 FWSM and CSM can be deployed in conjunction in several modes; the two main ones are the following:

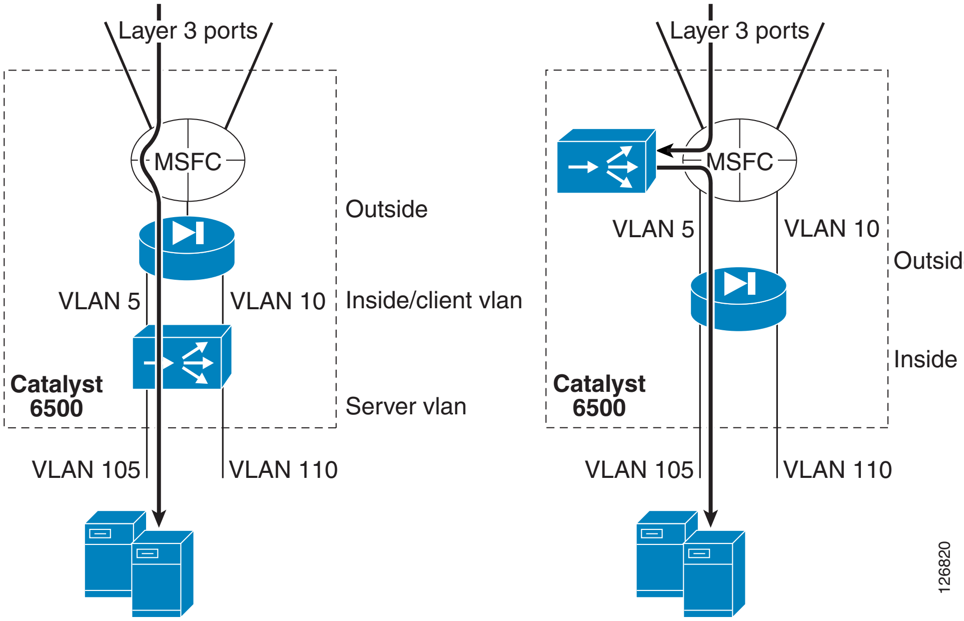
- FWSM in routed mode combined with the CSM in transparent mode—This design provides an easy-to-implement solution for multi-tier server farms.
- CSM in one-arm mode combined with the FWSM in transparent mode—This is the topic of this chapter for use with Supervisor 720.

Both designs can be implemented with the Catalyst 6500 Supervisor 2 or with the Catalyst 6500 Supervisor 720. The second design provides traffic optimization for connections that do not require any load balancing for increased performance at the price of a slightly more complex configuration using policy-based routing (PBR) or by using source NAT.

The load balancing and firewalling configuration with FWSM and CSM can have the following two main schemes:

- Inline CSM—MSFC–FWSM–CSM–servers (see Figure 4-3 to the left).
- One-arm CSM—MSFC–FWSM-servers + MSFC–CSM (see Figure 4-3 to the right).

Figure 4-3 Inline Design versus CSM One-Arm Mode with FWSM Transparent Mode



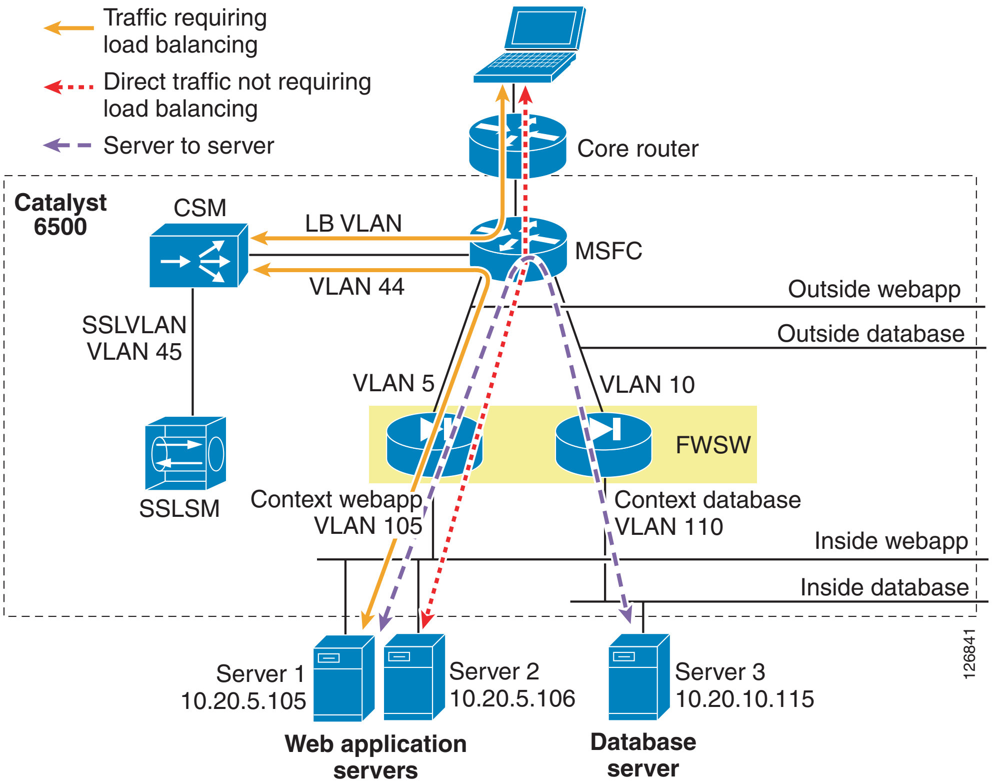
The benefits of this design include the fact that the denial of service (DoS) protection capabilities of the CSM and FWSM are combined, as follows:

- The CSM protects against DoS (SYN flood) attacks directed at the virtual IP (VIP).
- The FWSM protects against DoS (SYN flood) attacks directed at non-load balanced servers.

Topology and Service Processing Sequence

Figure 4-4 shows the logical topology of the design presented in this chapter, and the VLANs and IP addresses used in the configurations.

Figure 4-4 Logical Topology without Redundancy



The firewall is virtualized in multiple contexts; two in this example protect respectively the presentation/application tier and the database tier.

The Catalyst 6500 is the blue rectangle that includes the FWSM, the CSM, and the Cisco Secure Socket Layer Service Module (SSLSM).

Traffic that requires load balancing (represented with a continuous line in Figure 4-4) hits the MSFC first; then it is intercepted by the route health injection route (a route that the CSM installs dynamically when the VIP address is active); then it goes to the CSM for the load balancing decision.

The CSM performs the rewriting of the destination IP address to the server IP address and then sends the traffic to the MSFC in the Catalyst 6500 to be routed to the appropriate servers, wherever this server might be located; that is, the CSM can load balance across any application tier or firewall context (it is up to the firewall to prevent unwanted traffic from entering a given segment).

The traffic then enters the appropriate segment through a firewall instance. The firewall is operating in bridge mode; as such, the MSFC simply uses ARP to find the real IP address and then forwards the traffic through the firewall instance.

The return traffic takes the reverse path, and a PBR ACL is configured on the MSFC interface to push the traffic back to the CSM.

Traffic that does not require load balancing is forwarded directly to the servers. This traffic includes client-to-server traffic that is not subject to any load balancing rule on the CSM (dotted line in Figure 4-4) and server-to-server traffic (dashed line in Figure 4-4).

Note Whether the service modules are physically in the same Catalyst 6500 or have been placed in a "service switch" is not relevant for the topic of this chapter. This chapter assumes that the CSM and the FWSM are in the same chassis, but it is equally applicable if the FWSM is placed in an aggregation switch and the CSM is placed on a "service switch"; that is, an external Catalyst 6500 used to provide mostly content functions such as load balancing, SSL offloading, and providing connectivity to reverse proxy caches.

Using the FWSM to segregate server farms is useful for servers that belong to different organizations, for applications to which you want to apply different filtering policies, or to tier web/application/database servers to make it more difficult for a hacker to access confidential information.

To segregate servers with different security levels, assign them to different VLANs, with each VLAN trunked to the FWSM and assigned to a different firewall context.

Note Currently, in transparent mode, each firewall context provides one outside interface and one inside interface.

The correct placement of the MSFC is a key element for the performance of this design. The traffic hitting the aggregation switches from the core should go to the MSFC first and the FWSM afterwards. This enables the use of Layer 3 links to connect the aggregation switches with the core and the assignment of the MSFC as the default gateway of the servers.

You can use FWSM Release 2.2 and above in either routed mode or bridge mode. This chapter uses the FWSM in transparent mode.

Configuration Details

This section includes the following topics:

- Configuring Inside and Outside Interfaces
- Basic ACL Template
- DoS Protection and Identity NAT
- Using Timeouts
- Using Virtual Fragment Reassembly

Configuring Inside and Outside Interfaces

Each FWSM interface is assigned a numeric security level. The term *inside* is typically used to identify the interface with the higher security level, and *outside* is used to identify the interface with the lower security level.

Deciding which interface of the firewall should be the outside and which one should be the inside requires understanding the specific differences in the way the firewall handles traffic coming from either interface. In most deployments, the intuitive configuration uses the inside facing the servers and the outside facing the core network.

Because servers can be both targets and agents of an attack, the choice of where the inside or outside should be placed is often debated. Following are the differences between the outside and inside:

- TCP Intercept (with SYN cookies starting from FWSM 2.3) applies only for connections from outside clients to hosts or servers in the higher security level interfaces.
- Maximum connection limits are applicable for hosts or servers in higher security level interfaces.
- Established command allows connections from a lower security level host to a higher security level host if there is already an established connection from a higher security level host to a lower security level host.
- TCP sequence randomization applies only for hosts or servers in the higher security level interfaces.
- The SMTP Fixup is applied only for inbound connections to protect SMTP servers in the inside network.

The above list indicates that in general it makes sense to place the servers on the inside of the firewall to minimize the chance of a server being compromised.

Still, it is important to mitigate the effects of an attack from a server that has been compromised. For this reason, Cisco recommends configuring outbound ACL filtering by applying an inbound ACL on the inside interface, and to make sure to configure anti-spoofing ACLs very close to the server farm to prevent a compromised server from saturating the connection table of any stateful device in the path, including the firewall. Alternately, if you are deploying the firewall in routed mode, you can also use unicast RPF check on the firewall itself.

Basic ACL Template

Cisco recommends that each context on the FWSM be configured for both inbound (access-group <name> in interface outside) and outbound filtering (access-group <name> in interface inside). The ACLs are tuned according to the specific security policies: ACLs are different for the presentation tier of a business-to-customer (B2C) environment or for the DMZ services in general than they are for the intranet applications or even the database tier of the B2C environment.

The following ACLs are just an example of access list entries for anti-spoofing and for allowing the control traffic to enter the appropriate segment. For example, you want to allow the traffic from the MSFC to enter the server VLANs, so you need to allow traffic from 10.20.5.2, 10.20.5.3, and 10.20.5.1 (the IP addresses of the MSFC VLAN interfaces) to enter the outside interface of the webapp context. However, you want to deny any other 10.20.5.x address from entering the webapp context because it would be a spoofed address.

You also want the CSM to be able to monitor the servers, so you need to allow the traffic originating from the CSM into the server VLANs (in this example the CSM belongs to the subnet 10.20.44.x).

The ACL should also prevent attacks that exploit the directed broadcast, which might be needed for messaging software. For this purpose, you need to permit UDP traffic for the directed broadcast address for the port used by the specific application.

If the management traffic is carried inband, be sure to open the necessary ports for NTP, syslog, SNMP, SSH, and so on.

Remember to take advantage of the logging feature. Quoting the product documentation: “When you enable logging for message 106100, if a packet matches an ACE, the FWSM generates a system message at the first hit and at the end of each interval, identifying the total number of hits during the interval. At the end of each interval, the FWSM resets the hit count to 0. If no packets match the ACL during an interval, the FWSM deletes the flow entry.”

By default, when traffic is denied by an extended ACE, the FWSM generates system message 106023. The log option allows you to enable message 106100 instead of message 106023. By default, if you enter the log option without any arguments, you enable system log message 106100 at the default level (6) and for the default interval (300 seconds). The following configuration uses level 4 instead of 6. By default, the default **deny any any** does not generate a log, so be sure to explicitly define a log.

For the other entries of the ACL you should follow the best practices in ACL tuning, which is not the purpose of this document.

Note WEB Server VLAN=10.20.5.0/24, DATABASE VLAN = 10.20.10.0/24, MFSC = 10.20.5.1, CSM network 10.20.44.0. Notice that the first entry in this access list permits the traffic from the MSFC interface and is followed by entries that protect against source IP spoofing (deny 10.20.5.0/24 any) followed by permit entries to allow the management traffic, and traffic originated by the CSM or the SSLSM.

```
! INBOUND FILTERING
!
access-list portal-in remark  >> MSFC IP addresses allowed <<
access-list portal-in extended permit ip 10.20.5.1 255.255.255.255 10.20.5.0 255.255.255.0
access-list portal-in extended permit ip 10.20.5.2 255.255.255.255 10.20.5.0 255.255.255.0
access-list portal-in extended permit ip 10.20.5.3 255.255.255.255 10.20.5.0 255.255.255.0
access-list portal-in remark .
access-list portal-in remark >> allow CSM probes to monitor the servers <<
access-list portal-in extended permit ip 10.20.44.0 255.255.255.0 10.20.5.0 255.255.255.0
access-list portal-in remark .
access-list portal-in remark >> antispoofing <<
!
! By default, when traffic is denied by an extended ACE,
! the FWSM generates system message 106023. The log option
! allows you to enable message 106100 instead of message 106023
!
access-list portal-in extended deny ip 10.20.5.0 255.255.255.0 any log 4
access-list portal-in remark .
access-list portal-in remark >> prevent exploitation of directed broadcast <<
access-list portal-in extended deny icmp any 10.20.5.255 255.255.255.255
access-list portal-in extended deny tcp any 10.20.5.255 255.255.255.255
!
! For connectionless protocols such as ICMP you either need
! ACLs to allow ICMP in both directions or you need to enable the ICMP
! inspection engine with the 2.3 code
!
access-list portal-in remark .
access-list portal-in remark >> allow ICMP to function
access-list portal-in extended permit icmp any 10.20.5.0 255.255.255.0 echo
access-list portal-in extended permit icmp any 10.20.5.0 255.255.255.0 echo-reply
access-list portal-in extended permit icmp any 10.20.5.0 255.255.255.0 time-exceeded
access-list portal-in extended permit icmp any 10.20.5.0 255.255.255.0 unreachable
access-list portal-in remark .
access-list portal-in remark >> allow access to web-based applications
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq 80
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq 8080
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq 443
!
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq ftp-data
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq ftp
!
access-list portal-in remark .
access-list portal-in remark >>  messaging applications (add the port number information)
<<
access-list portal-in extended permit udp any 10.20.5.255 255.255.255.255
```

```
access-list portal-in remark .
access-list portal-in remark >>  allow SSH, SNMP traffic (if carried inband) <<
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq 22
access-list portal-in extended permit udp any 10.20.5.0 255.255.255.0 eq 22
access-list portal-in extended permit udp any 10.20.5.0 255.255.255.0 eq 161
access-list portal-in extended permit udp any 10.20.5.0 255.255.255.0 eq 162
access-list portal-in remark the implicit deny doesn't generate a log
access-list portal-in extended deny ip any any log 4
access-group portal-in in interface outside
```

Note Remember to configure an entry on the ACLs to allow the CSM probes to enter the server farm segments.

Note If this firewall is placed at the Internet edge, you should complete the inbound ACL with entries to deny the following source IP addresses: RFC 1918, the loopback address 127.0.0.0/8 range, multicast, and RFC 3330 addresses. The ACL template provided in this example assumes that there is a first layer of security already deployed on the border routers.

Outbound filtering should be configured to make it more difficult for a compromised server to open a connection to the attacker PC (imagine that a hacker managed to install netcat on a server and uses port 80 to open a command shell back to its PC).

Outbound filtering should prevent servers from initiating TFTP transfers from an outside host (for example, the hacker PC).

Outbound filtering should also prevent source IP spoofing. This is the right place to prevent a server from originating traffic from a subnet to which it does not belong; a compromised server can saturate the translation table of a firewall or a load balancer by cycling multiple source IP addresses.

You can also configure anti-spoofing by using Unicast Reverse Path Forwarding (uRPF) on the Sup720, but anti-spoofing is best done as close as possible to the source, which is why Cisco recommends configuring it on an ACL applied to the inside interface of each context.

The outbound ACL can be made more granular by specifying to which device the servers are allowed to connect and to which port.

For example, the ACL for the webapp context in this document could be the following:

```
access-list portal-out extended permit udp 10.20.5.0 255.255.255.0 host <DNS server> eq 53
access-list portal-out extended permit tcp 10.20.5.0 255.255.255.0 host <DNS server> eq 53
access-list portal-out extended permit udp 10.20.5.0 255.255.255.0 eq 123 host <NTP
server> eq 123
access-list portal-out extended permit tcp 10.20.5.0 255.255.255.0 host <database server>
eq 1434
access-list portal-out extended permit tcp 10.20.5.0 255.255.255.0 host <database server>
eq 1433
access-list portal-out extended permit tcp 10.20.5.0 255.255.255.0 host <database server>
eq 153
access-list portal-out extended permit udp 10.20.5.0 255.255.255.0 host <database server>
eq 153
access-list portal-out remark the implicit deny doesn't generate a log
access-list portal-out extended deny ip any any log 4
```

Note Some of the listed protocols, such as Distributed Component Object Model (DCOM) for example, negotiate dynamic ports that the FWSM might not necessarily be able to open. In this case, the ACL needs to be simplified by specifying the hosts that are allowed to talk without specifying the ports.

Apply the ACL to the inside interface as follows:

```
access-group portal-out in interface inside
```

Follow a similar configuration for the other contexts.

DoS Protection and Identity NAT

NAT is mainly deployed at the Internet edge of enterprise campus networks to conserve public IP addresses and to hide the IP addresses used on the intranet. Intranet data centers often do not require NAT. Firewalls, by default, require NAT to be configured either for meaningful translation between public and private addresses or with identity NAT, which disables address translation.

NAT is typically performed between an “inside global” and an “inside local” address. You can configure NAT on the FWSM using two different commands: **static** and **nat 0**. The **static** configuration lets you specify the maximum number of embryonic connections, which affects how the FWSM responds to a DoS attack. The **nat 0 access-list** configuration does not let you specify the number of embryonic connections.

In the case of the FWSM deployed in transparent mode, NAT is disabled, but you still need to use the NAT syntax to configure TCP Intercept against DoS attacks.

The following is the syntax for the **static** command:

```
static (inside, outside) global-IP-address local-IP-address netmask netmask
max-connection-count embryonic-count
```

With the **static** command, you need to list the interface pairs:

```
FWSM/webapp(config)# static (inside,outside) 10.20.5.0 10.20.5.0 netmask 255.255.255.0 tcp
0 1000
```

The static command lets you specify the number of embryonic connections (1000 in the example). This allows the TCP Intercept feature to operate if too many half-opened connections are present. TCP Intercept protects the servers from SYN flooding.

When the number of embryonic connections passes a certain limit (configured by the user), the FWSM intervenes with the normal connection process, validating incoming connection requests by replying to the client SYN with an acknowledgement (SYN-ACK) on behalf of the destination device (the server).

If the client responds with the appropriate acknowledgement (ACK), the FWSM establishes a connection with the destination device, usually a server, on behalf of the client and then weaves the two connections together. This process prevents illegitimate connection requests from consuming the limited resources of enterprise endpoints, thus thwarting the DoS attack.

The FWSM TCP Intercept feature employs an embryonic limit, which is a threshold that defines the number of “incomplete” connections the FWSM permits before intercepting further connection requests (SYN packets). The definition of an incomplete connection is a client which has not responded to the SYN-ACK sent by the destination device protected by the FWSM. When the embryonic limit is surpassed, the FWSM begins intercepting incoming connection requests.

You can monitor the FWSM TCP Intercept operation as follows:

```
FWSM/webapp# sho np 3 int stats
********************************************************************************
                     TCP Intercept Statistics Counters                         *
********************************************************************************
      Total Number of Leaves Allocated: 335000
           Total Number of Free Leaves: 335000
                     Timer Wheel Index: 4948
Total Number of Retransmitted SYN/ACKs: 0
    Total Number of Retransmitted SYNs: 0
```

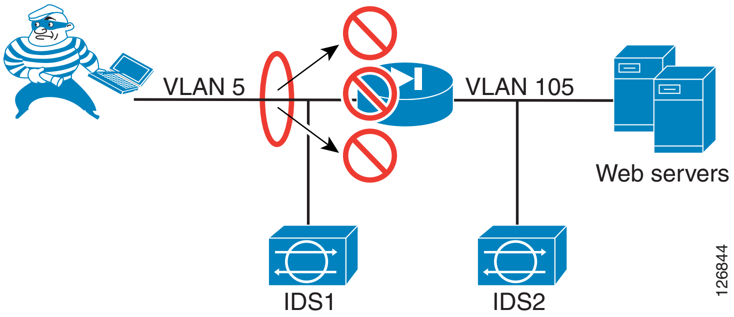
```
      Total Number of Aborted Sessions: 0
Total Number of SYN/ACK Timeout Aborts: 0
    Total Number of SYN Timeout Aborts: 0
Total Number of Xmit SYN with Diff Seq: 0
    Total Number of ACKs with Diff ACK: 0
     Total Number of Client RST Aborts: 0
 Total Number of SYN/ACK with Diff ACK: 0
     Total Number of Server RST Aborts: 0
         Total Number of Normal Aborts: 0
      Total Number of Timer TLV Frames: 0
    Number of Garbage Collected Leaves: 0
    Number of Intercept Address Errors: 0

*******************************************************************************
```

Note the following:

- Total Number of SYN/ACK Timeout Aborts—Number of intercepted sessions that timed out waiting for an ACK to arrive in response to the SYN/ACK sent to the client.
- Total number of SYN Timeout Aborts—Number of intercepted sessions that were timed out waiting for a SYN/ACK to arrive from the server in response to the SYN the FWSM sent to the server.
- Total Number of Normal Aborts—Number of sessions successfully intercepted resulting in a complete handshake between client and server.

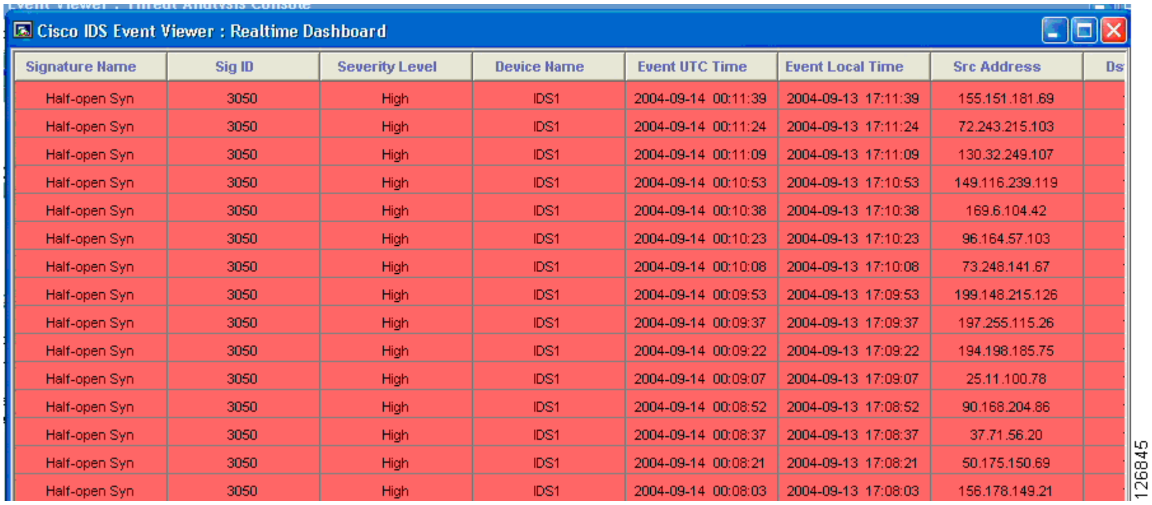
An intrusion detection device placed in front and behind the FWSM indicates that the DoS protection mechanism is functioning on the FWSM and also brings up the topic of whether an IDS sensor is better placed outside or inside the FWSM. Figure 4-5 shows a design in which IDS1 is placed outside the FWSM and IDS2 is placed inside the FWSM.

Figure 4-5 DoS Protection with FWSM and IDS Detection



IDS1 detects the DoS attack, as shown in Figure 4-6.

Figure 4-6 DoS Detection with IDS1



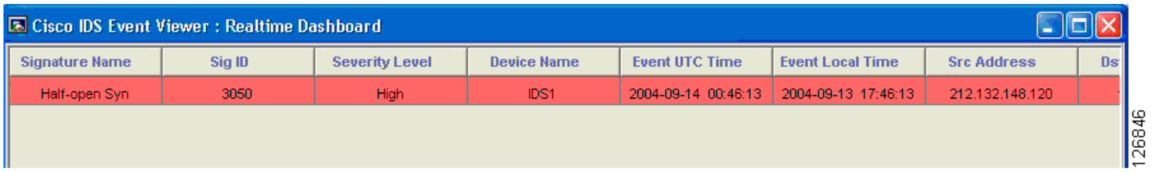
Cisco IDS Event Viewer : Realtime Dashboard

Signature Name	Sig ID	Severity Level	Device Name	Event UTC Time	Event Local Time	Src Address	Ds
Half-open Syn	3050	High	IDS1	2004-09-14 00:11:39	2004-09-13 17:11:39	155.151.181.69	
Half-open Syn	3050	High	IDS1	2004-09-14 00:11:24	2004-09-13 17:11:24	72.243.215.103	
Half-open Syn	3050	High	IDS1	2004-09-14 00:11:09	2004-09-13 17:11:09	130.32.249.107	
Half-open Syn	3050	High	IDS1	2004-09-14 00:10:53	2004-09-13 17:10:53	149.116.239.119	
Half-open Syn	3050	High	IDS1	2004-09-14 00:10:38	2004-09-13 17:10:38	169.6.104.42	
Half-open Syn	3050	High	IDS1	2004-09-14 00:10:23	2004-09-13 17:10:23	96.164.57.103	
Half-open Syn	3050	High	IDS1	2004-09-14 00:10:08	2004-09-13 17:10:08	73.248.141.67	
Half-open Syn	3050	High	IDS1	2004-09-14 00:09:53	2004-09-13 17:09:53	199.148.215.126	
Half-open Syn	3050	High	IDS1	2004-09-14 00:09:37	2004-09-13 17:09:37	197.255.115.26	
Half-open Syn	3050	High	IDS1	2004-09-14 00:09:22	2004-09-13 17:09:22	194.198.185.75	
Half-open Syn	3050	High	IDS1	2004-09-14 00:09:07	2004-09-13 17:09:07	25.11.100.78	
Half-open Syn	3050	High	IDS1	2004-09-14 00:08:52	2004-09-13 17:08:52	90.168.204.86	
Half-open Syn	3050	High	IDS1	2004-09-14 00:08:37	2004-09-13 17:08:37	37.71.56.20	
Half-open Syn	3050	High	IDS1	2004-09-14 00:08:21	2004-09-13 17:08:21	50.175.150.69	
Half-open Syn	3050	High	IDS1	2004-09-14 00:08:03	2004-09-13 17:08:03	156.178.149.21	

126845

If the FWSM has been configured with an embryonic connection limit, IDS2 does not show alarms for the DoS attack, as shown in Figure 4-7.

Figure 4-7 DoS Detection on IDS2



Cisco IDS Event Viewer : Realtime Dashboard

Signature Name	Sig ID	Severity Level	Device Name	Event UTC Time	Event Local Time	Src Address	Ds
Half-open Syn	3050	High	IDS1	2004-09-14 00:46:13	2004-09-13 17:46:13	212.132.148.120	

126846

You can verify that the DoS is being stopped on the FWSM by looking at the statistics:

```
FWSM/webapp# sho np 3 int stats
********************************************************************************
                     TCP Intercept Statistics Counters                         *
********************************************************************************
      Total Number of Leaves Allocated: 335000
           Total Number of Free Leaves: 298612
                     Timer Wheel Index: 4088
Total Number of Retransmitted SYN/ACKs: 0
    Total Number of Retransmitted SYNs: 0
      Total Number of Aborted Sessions: 2246200
Total Number of SYN/ACK Timeout Aborts: 2246200
    Total Number of SYN Timeout Aborts: 0
Total Number of Xmit SYN with Diff Seq: 0
    Total Number of ACKs with Diff ACK: 0
     Total Number of Client RST Aborts: 0
 Total Number of SYN/ACK with Diff ACK: 0
     Total Number of Server RST Aborts: 0
         Total Number of Normal Aborts: 0
      Total Number of Timer TLV Frames: 50543
    Number of Garbage Collected Leaves: 0
    Number of Intercept Address Errors: 0

********************************************************************************
```

Note Chapter 9, "Deployment of Network-Based IDS Sensors and Integration with Service Modules," recommends "placing" the IDS "outside" the FWSM. "Placing" in the context of the design of the Catalyst 6500 simply specifies the VLAN from which to span. This recommendation relates to the FWSM and Catalyst 6500 architecture, the need to generate only one copy of each frame, and to capture both directions of the traffic in a fully redundant topology.

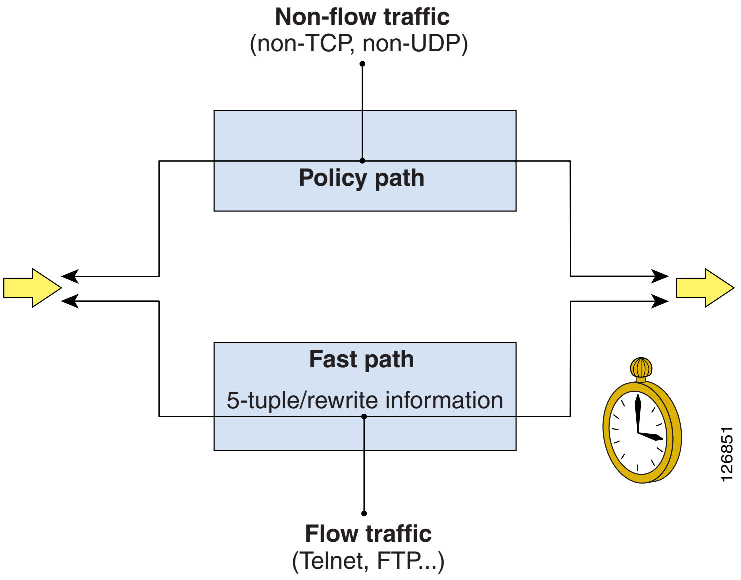
Using Timeouts

It is important to remember that, unlike a router, a firewall is a Layer 4/7 device. This means that a firewall maintains state information for TCP connections and caches the flows in a table. Flows are cleared when TCP connections are closed, or in the case of UDP traffic, they are typically aged out.

Figure 4-8 is a high-level illustration of how the FWSM handles traffic. The policy path is the one that is normally used by the first packet of a given flow. The FWSM handles the first packet similarly to how Layer 3 forwarding works on a router. The firewall performs a route table lookup, finds the Layer 2 rewrite information, and uses ARP to find the destination IP if it is a directly-connected subnet. At the same time, the firewall applies the configured security policy, including ACLs. The packet is eventually sent to the destination.

The FWSM creates flows for TCP and UDP traffic and treats the remaining traffic as non-flow traffic. For the subsequent packets in a flow, the FWSM creates a fast path by consulting a connection table that provides rewrite information for the given flow. Entries for the fast path must be aged out because traffic that is taking this path uses memory space, with an entry created for each flow. In contrast, the traffic using the policy path is identified in a routing table that does not change in size.

Figure 4-8 Flow and Non-Flow Traffic



To control the aging process, the FWSM uses an xlate timer and a connection timer. Xlate entries are associated with a static, dynamic, or identity address translation. Even if a connection is not completed, an embryonic entry is created in the xlate table for the specific local and global address. If the connection setup is completed, the FWSM creates an entry in the connection table that is aged out independently of the xlate table. The xlate aging mechanism intervenes if the connection is not cleared. If the connection remains half open, the SYN is no longer sent to the servers but is terminated at the FWSM, once the connection limit specified in the static configuration is reached.

For connections initiated from a higher security level to a lower security level, the default xlate timeout is three hours on the FWSM. The timeout for xlates created from a lower security level interface to a higher security level interface is one minute. Half-opened connections are aged out much faster. By default, connection entries are aged out in one hour.

It is very important to change the connection timeout to the longest timeout required by the applications used in the data center. For example, TN3270 applications can have connections idle for many hours. For this reason, you should modify the timeout to approximately eight hours, using the command **timeout conn 8**.

Using Virtual Fragment Reassembly

By default, the FWSM drops fragments, but many applications generate fragments. Enabling fragment forwarding opens the door for fragment attacks, however, such as those described in RFC 1858. The FWSM provides protection against that type of attack by means of virtual fragment reassembly. The following configuration example enables the virtual fragment reassembly on the interfaces “web”, “app”, and “windows” with a temporary database buffer size of 200 packets.

```
fragment size 200 web
fragment size 200 app
fragment size 200 windows
```

The **fragment** command also lets you define the maximum number of fragments that can be chained together and how long the FWSM waits for the fragments to arrive before discarding them. The syntax is as follows: **fragment** {**size | chain | timeout**} *interface*.

Configuring Redundancy

This section includes the following topics:

- Using Spanning Tree
- Using SPAN Reflector
- Configuring the FWSM to Bridge BPDUs
- Assigning Spanning-Tree Priorities
- Loopguard
- Verifying FWSM Failover Time

The FWSM supports stateful failover, which means that TCP connection and UDP flows are replicated from the active to the standby firewall module so that when a failure occurs, the standby becoming active keeps forwarding traffic for existing connections. The failover time for the FWSM Release 2.2 and 2.3 is approximately three seconds.

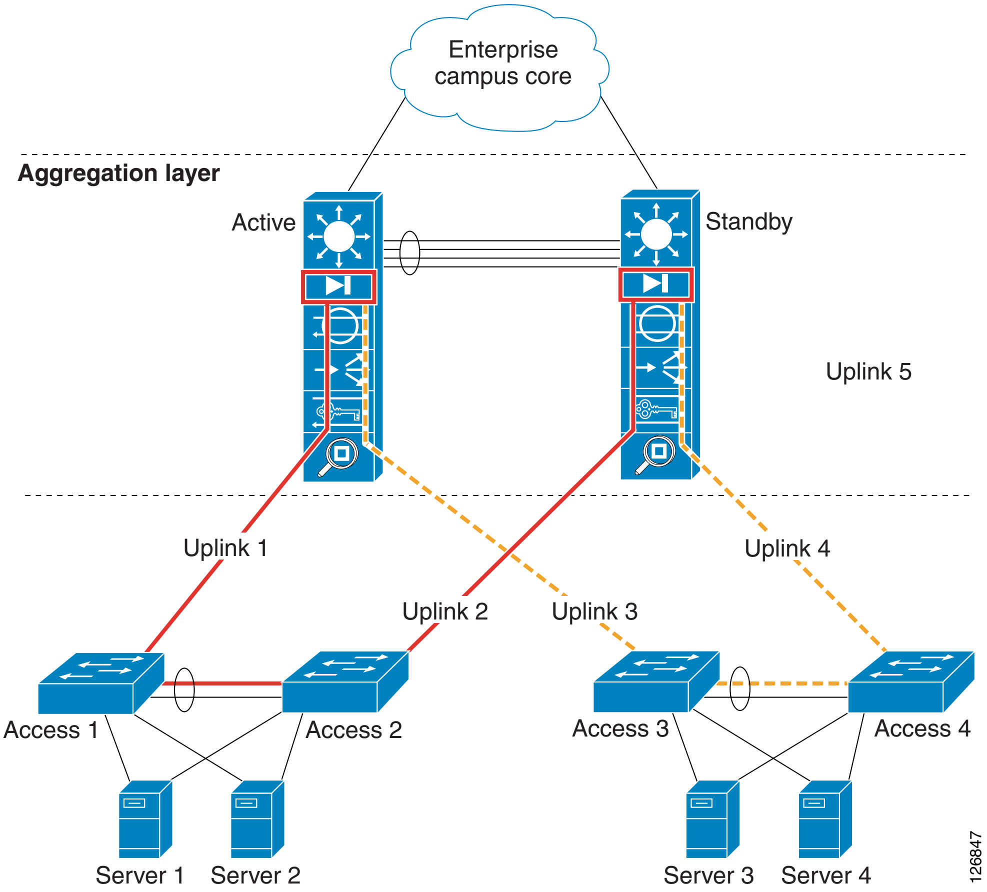
In a redundant configuration, the firewall modules exchange information over a failover VLAN, which allows detection of any failure of the peering device and selection of the active and standby devices. The FWSM also uses a VLAN to replicate the state information of the traffic that the firewall is forwarding.

However, losing a link on a given interface VLAN does not bring down the active firewall or activate FWSM failover. For this reason it is important to provide redundant Layer 2 paths in the design.

Note This loop-free design becomes possible with the latest FWSM releases by using Rapid Link Failure Detection.

Figure 4-9 shows an incorrect configuration that fails to recognize this.

Figure 4-9 Incorrect FWSM Design (if Rapid Link Failure Detection is not Available)



In Figure 4-9, a pair of redundant firewall modules is operating in active/standby mode. The Layer 2 configuration is incorrect because all the access switches are connected with a loop-free topology. For example, the VLAN illustrated in red (on the left side) is only present on one access switch and it is not trunked between the aggregation switches. If Uplink 1 fails, traffic cannot reach Server 1 or Server 2. The active firewall does not failover just because one VLAN loses connectivity to the access switches.

The correct configuration requires redundant Layer 2 paths, using a looped topology, as shown in Figure 4-10. With this configuration, the failure of Uplink 1 does not isolate Server 1 and Server 2.

Figure 4-10 Correct FWSM Design

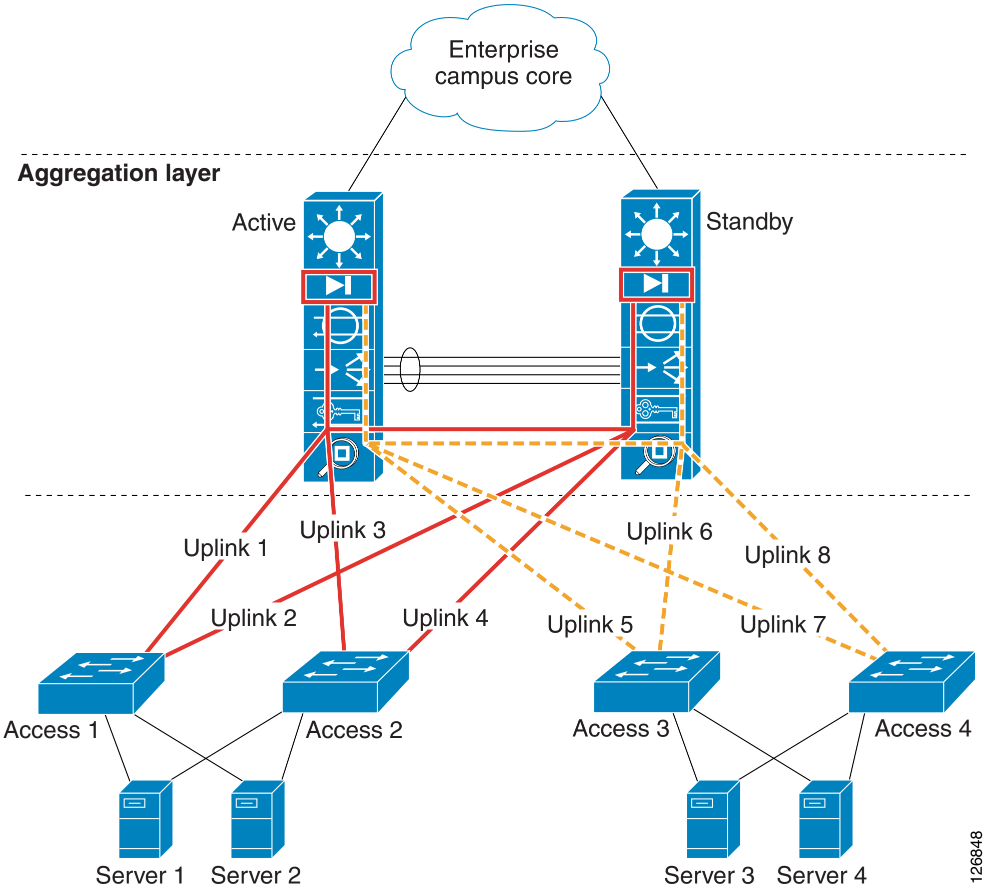
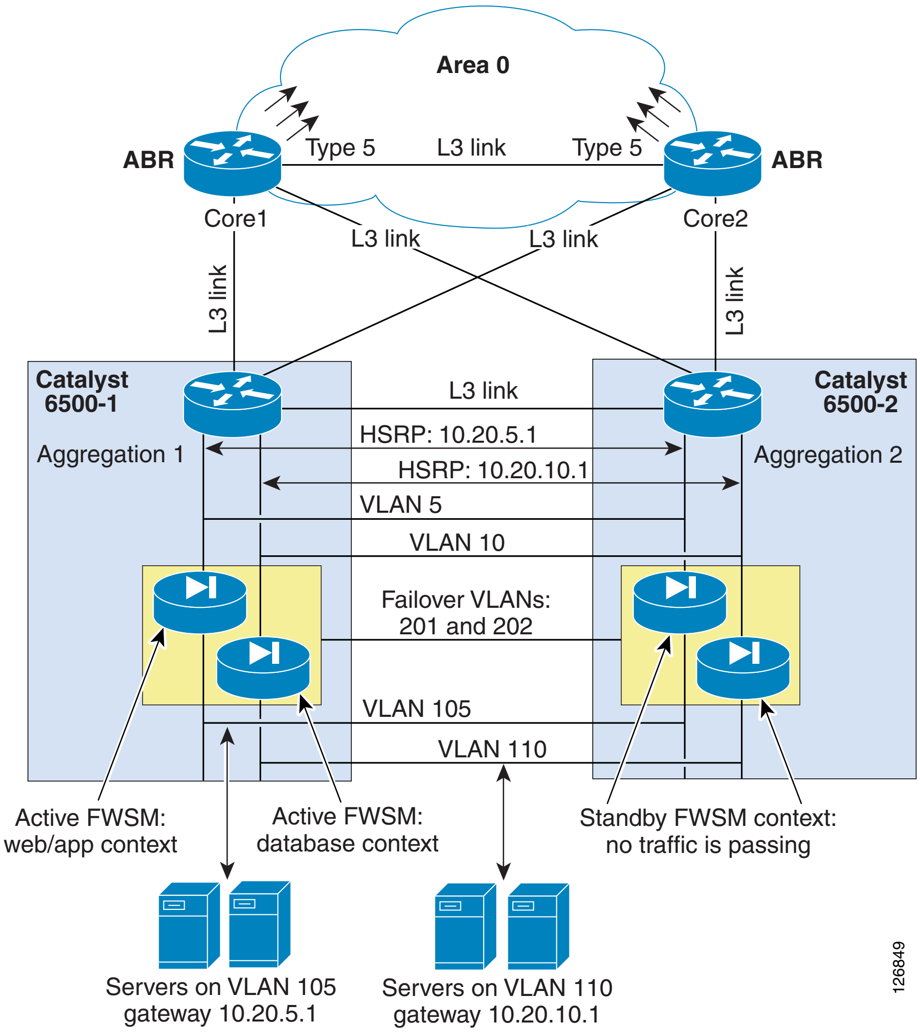


Figure 4-11 shows redundant Catalyst 6500s, each with a FWSM configured for transparent operation. The FWSM is configured in “multiple mode”, which provides multiple logical contexts. An administration context is defined as well as two production contexts. FWSM context *webapp* bridges the 10.20.5.0 subnet and context *database* bridges the 10.20.10.0 subnet.

This topology provides a redundant active/standby solution with Spanning Tree, HSRP, and FWSM priorities combining to make the 6500-1 on the left the primary device. The FWSM in the 6500-1 is explicitly configured as the primary firewall and the FWSM in 6500-2 as the secondary firewall. As a result, 6500-1 with its service modules provides the active path for all traffic. The network as configured protects against all single failures, including complete failure of switch 6500-1.

Figure 4-11 Redundant Design with the FWSM in Transparent Mode


Area 0
ABR
Type 5
L3 link
Type 5
ABR
Core1
Core2
L3 link
L3 link
L3 link
L3 link
Catalyst 6500-1
Aggregation 1
L3 link
HSRP: 10.20.5.1
HSRP: 10.20.10.1
VLAN 5
VLAN 10
Catalyst 6500-2
Aggregation 2
Failover VLANs: 201 and 202
VLAN 105
VLAN 110
Active FWSM: web/app context
Active FWSM: database context
Standby FWSM context: no traffic is passing
Servers on VLAN 105 gateway 10.20.5.1
Servers on VLAN 110 gateway 10.20.10.1
126849

6500-1 and 6500-2 are configured for HSRP on VLAN 5 and 10 with the HSRP address providing the IP default gateway for servers on VLAN 105 and 110 respectively.

The redundant, active/standby FWSMs are configured for multiple, transparent mode. They are bridging multiple subnets. Each subnet bridges between a pair of VLANs: VLANs 5 and 105, and VLANs 10 and 110. The FWSM in 6500-1 is configured as the primary of the pair.

VLAN 201 and 202 are used for the failover firewall pair to arbitrate the active/standby role (VLAN 201) and to exchange state information (VLAN 202).

Using Spanning Tree

Spanning Tree is necessary for VLANs 5, 10, 105, and 110 in Figure 4-11, because dual-connected access switches complete the picture. Spanning Tree is unnecessary but equally configured for VLAN 201 and 202.

At steady state, a loop-free topology for the solution is guaranteed because the FWSMs are configured in active/standby pairs and the standby unit does not pass traffic. Also, with proper configuration the FWSM (Release 2.2) forwards Bridge Protocol Data Units (BPDUs) when in active state. This is highly desirable in a situation where both FWSMs mistakenly become active.

If the FWSM were not capable of bridging BPDUs, a loop is caused by the bridging of VLAN 5 and 105 (and 10 and 110) on both devices when both FWSMs become active. If the FWSM is configured to bridge BPDUs, the failure scenario when both FWSMs end up being active does not cause a loop because Spanning Tree blocks one link in the topology.

In Figure 4-11, VLANs connected by horizontal lines are trunked between the two Catalyst 6500s. Cisco recommends that this trunk be a Gigabit EtherChannel comprised of links from more than one Catalyst 6500 line card. This practice is referred to as multi-module channeling and protects the EtherChannel trunk from a single point of failure. Table 4-1 lists the function of each of the trunked VLANs in the topology. The full configuration listings for all solution components are provided in the following section,Configuration Listings, page 4-26.

Table 4-1 Trunked VLANs in Redundant Data Center Topology with FWSM at Layer 2

VLAN ID	Description
VLAN 30	Point-to-point Layer 3 VLAN between MSFCs. Its purpose is OSPF route distribution. It is the only trunked VLAN defined to OSPF.
VLAN 5	Outside VLAN for the webapp security zone or context (subnet 10.20.5.0). VLAN 5 also carries HSRP control traffic for 10.20.5.1.
VLAN 105	Inside VLAN for the webapp security zone or context (subnet 10.20.105.0).
VLAN 10	Outside VLAN for the database security zone or context (subnet 10.20.10.0). VLAN 10 also carries HSRP control traffic for 10.20.10.1.
VLAN 110	Inside VLAN for the database security zone or context (subnet 10.20.110.0).
VLAN 201	FWSM failover VLAN used to arbitrate the FWSM active/secondary roles to detect when the active device has failed and for configuration synchronization.
VLAN 202	FWSM VLAN used to replicate the state of connections.

Note Cisco also recommends using multiple EtherChannels; one (EtherChannel 1) for the data traffic (VLAN 5, 105, 10, and 110) and one (EtherChannel 2) for the failover VLANs (VLAN 201 and 202). This design reduces the chances for failures where both modules become active as a result of the loss of the control protocol communication between the firewalls. As an example, if EtherChannel 2 is lost, the secondary firewall probes the active firewall on the data VLANs before initiating a failover. If the primary firewall is still active, the secondary firewall does not become active.

When the firewall is configured in transparent mode, care needs to be taken in assigning spanning-tree root and secondary root roles to the VLANs connecting the firewall to the routing engine (VLAN 5 and 10 in Figure 4-11) and the VLANs connecting the firewall to the servers (vlan 105 and 110 in Figure 4-11). Root and secondary root assignments can make topologies converge in a deterministic way, are easier to troubleshoot, and optimize traffic paths. This topic is explained in further detail in Assigning Spanning-Tree Priorities, page 4-22.

Before entering the details of the Spanning Tree configuration, keep in mind these best practices:

- Make 6500-1 the root switch
- Make 6500-2 the secondary root

- Use Rapid PVST+ if you decide to bridge BPDUs on the FWSM. Failure to do this means that a failover reconfiguration could converge in ~30 seconds.
- Use Loop Guard, although when bridging BPDUs with the firewall (or any transparent device) you should not enable Loop Guard globally. For more details, see Loopguard, page 4-23.

Note If you are using regular PVST+, *do not bridge BPDUs on the FWSM*.

Using SPAN Reflector

When using Sup720 with an FWSM in the chassis running Cisco Native IOS, by default a SPAN session is used. If you check for unused sessions with **show monitor**, you see that "session 1" is in use:

```
agg#show monitor
Session 1
---------
Type                   : Service Module Session
```

This session is automatically installed for the support of hardware multicast replication when a firewall blade is in the Catalyst 6500 chassis. This is because an FWSM cannot replicate multicast streams, so if multicast streams sourced behind the FWSM must be replicated at Layer 3 to multiple line cards, the automatic session copies the traffic to the supervisor through a fabric channel.

If you have a multicast source that generates a multicast stream from behind the FWSM you need the SPAN reflector. If you place the multicast source on the outside VLAN, the SPAN reflector is not necessary. The SPAN reflector is incompatible with bridging BPDUs through the FWSM.

You can disable the SPAN reflector by using the **no monitor session service module** command.

Configuring the FWSM to Bridge BPDUs

The recommended configuration for loop avoidance for misconfigurations in the presence of redundant FWSMs operating in transparent mode is as follows:

- On the Catalyst 6500, configure Rapid PVST+.
- Configure the FWSM to bridge BPDUs (if there is no multicast source behind the FWSM).

Note An alternative and recommended solution is to force the FWSM to operate in bus mode using the **fabric switching-mode force bus** command. By using this command, the FWSM can support multicast sources on either the inside or the outside, BPDUs are bridged correctly, and cross-line card EtherChannels are supported. When the FWSM operates in bus mode, all traffic to and from the FWSM goes via the supervisor fabricand traffic from DFC-enabled line cards still uses the fabric connection.

Each FWSM context needs to be explicitly configured to bridge BPDUs by performing the following steps:

- Make sure that the Catalyst 6500 Sup720 is running Spanning Tree in Rapid PVST+ mode.
- Make sure that the Catalyst 6500-1 is the root for the VLANs connecting the firewall to the routing engine (VLAN 5 and 10).
- Make sure that the Catalyst 6500-2 is the secondary root for the VLANs connecting the firewall to the routing engine (VLAN 5 and 10).

- On the Catalyst Sup720, disable the default SPAN session for service modules if you do not need it (that is, if there is no multicast source on the inside of the firewall) with the **no monitor session servicemodule** command.
- Log into each FWSM context and configure the Ethertype ACL to bridge BPDUs, and apply the ACLs to the outside and the inside interfaces:

```
access-list BPDU ethertype permit bpdu
access-group BPDU in interface vlan5
access-group BPDU in interface vlan10
```

- To make sure that BPDUs are bridged correctly, check the server-side VLAN (for example, VLAN 105) and verify that VLAN 5 is the root bridge for VLAN 105 on the Catalyst 6500-1 (the Catalyst where the FWSM is active):

```
AGG1#show spanning-tree vlan 105
VLAN0105
  Spanning tree enabled protocol rstp
  Root ID    Priority    24581
             Address     0009.12ec.9f00
             Cost        3
             Port        1666 (Port-channel271)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    24681  (priority 24576 sys-id-ext 105)
             Address     0009.12ec.9f00
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300
```

Assigning Spanning-Tree Priorities

When using a transparent firewall, there are the following design choices regarding the assignment of spanning-tree priorities:

- Making Catalyst 6500-1 the root for all VLANs (5, 105, 10, and 110 in this example) and Catalyst 6500-2 the secondary root for all VLANs (5, 105, 10,and 110). Bridging VLANs 5 and 105 together means that the VLAN with the lower ID (VLAN 5 in this case) becomes the root for VLAN 105. Bridging VLANs 10 and 110 together means that VLAN 10 becomes the root for VLAN 110. Depending on the design goals, you might choose to assign lower IDs (VLAN numbers) to all the outside VLANs (5 and 10 in this example) and higher IDs to the inside VLANs (105 and 110 in this example). This choice forces the spanning-tree forwarding path through the active firewall, so if you want to allocate a new subnet by bridging VLAN 6 and 106, make sure that VLAN 6 is the outside and VLAN 106 is the inside.
- Making the Catalyst 6500-1 root only for the outside VLANs (VLAN 5 and 10) and the Catalyst 6500-2 the secondary root. This design choice is equivalent to the previous one, except that the inside VLANs have the default priority (32768). This choice forces the spanning-tree forwarding path through the active firewall regardless of the VLAN ID of the inside VLANs. For example, if the bridge pair is VLAN 6 and VLAN 106, you could very well decide to use VLAN 106 for the outside bridged with VLAN 6 on the inside because you would keep the default priority for VLAN 6.
- With the previous design choice, if both firewalls stop forwarding traffic, the Layer 2 topology reconverges in a non-deterministic way because no priority is assigned to the inside VLANs. To address this problem, you could decide to create four tiers of priorities (instead of the normal approach of root and secondary root). You could assign the following priorities:
  - Priority 4096 to the Catalyst 6500-1 outside VLANs
  - Priority 8192 for the Catalyst 6500-2 outside VLANs
  - Priority 24576 to the Catalyst 6500-1 inside VLANs

- Priority 28672 to the Catalyst 6500-2 inside VLANs

With this configuration, when either firewall is active, the root path is through the active firewall. If both firewalls fail, Catalyst 6500-1 becomes the root.

In Figure 4-10, the link from Access 1 to Aggregation 1 (uplink1) is forwarding for the inside VLANs, and the link from Access 1 to Aggregation 2 (uplink2) is blocking for the inside VLANs:

```
access#show spanning-tree vlan 105
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Po1              Root FWD 5000      128.1665 P2p
Po2              Altn BLK 5000      128.1666 P2p
```

As displayed in Figure 4-11, the firewall in Aggregation 1 is active, and it provides the path to the root (that is, to VLAN 5 on Catalyst 6500-1, which is the root). If the firewall fails over, the firewall in Aggregation 2 becomes active and it becomes the path towards the root (that is, to VLAN 5 on Catalyst 6500-1). This means that uplink2 on Access 1 becomes forwarding, thus ensuring the most direct path between the servers and the active firewall:

```
access#show spanning-tree vlan 105
Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Po1              Altn BLK 5000      128.1665 P2p
Po2              Root FWD 5000      128.1666 P2p
```

This behavior is the result of giving the outside VLANs a lower priority and/or ID than the inside VLANs.

With the first design described above, you can configure the VLANs as follows:

```
Catalyst 6500-1: spanning-tree vlan 1-1000 root primary
Catalyst 6500-2: spanning-tree vlan 1-1000 root secondary
```

Choose the outside and inside VLANs such that the VLAN ID for outside is a smaller number than the inside VLAN (for example, use 5-10 as the outside VLANs and 105-110 as the inside VLANs).

The second approach assigns the root role only to the outside VLANs:

```
Catalyst 6500-1: spanning-tree vlan 5-10 root primary
Catalyst 6500-2: spanning-tree vlan 5-10 root secondary
```

The third approach assigns specific priorities to the VLANs as follows:

```
Outside VLANs on Catalyst 6500-1: spanning-tree vlan 5-10 priority 4096
Outside VLANs on Catalyst 6500-2: spanning-tree vlan 5-10 priority 8192
Inside VLANs on Catalyst 6500-1: spanning-tree vlan 105-110 priority 24576
Inside VLANs on Catalyst 6500-2: spanning-tree vlan 105-110 priority 28672
```

Loopguard

If using loopguard globally, when you manually force a failback, the port channel connecting to the secondary firewall goes into loop inconsistent state:

```
Po273            Desg BKN*3330      128.1672 P2p *LOOP_Inc
```

The reason is that the secondary firewall that was active before the manual failback is not now forwarding any traffic, and loopguard is not receiving any BPDUs on a VLAN that was previously sending them. This is an expected behavior, because the firewall operates as active/standby, and this happens only when you type “failover active” on the primary after the secondary is active.

For this reason, when using a firewall operating in transparent mode, or when using a transparent device at the aggregation layer, it is good practice to disable loopguard globally and to enable it only on the interfaces that require it. At the aggregation layer, these are the EtherChannels connecting the two aggregation switches.

To summarize, these are the required configuration steps:

```
no spanning-tree loopguard default
interface Port-channel1
(config-if)# spanning-tree guard loop
```

Verifying FWSM Failover Time

The FWSM failover time with transparent mode and Rapid PVST+ is ~3 seconds; other configurations might have slightly higher convergence times. The FWSM in 6500-2 becoming active starts bridging traffic, including BPDUs. Spanning Tree (Rapid PVST+) converges immediately. You can verify this by simply looking at the Spanning Tree topology on VLAN 105 and VLAN 5 on 6500-2:

```
AGG2# show spanning-tree vlan 105
VLAN0105
  Spanning tree enabled protocol rstp
  Root ID    Priority    24681
             Address     0009.12ec.9f00
             Cost        1
             Port        1667 (Port-channel1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Bridge ID  Priority    28777  (priority 28672 sys-id-ext 105)
             Address     00d0.ff88.6000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Po271            Root FWD 3         128.1665 P2p
Po1              Desg FWD 1         128.1667 P2p

AGG2# show spanning-tree vlan 5

VLAN0005
  Spanning tree enabled protocol rstp
  Root ID    Priority    24581
             Address     0009.12ec.9f00
             Cost        1
             Port        1667 (Port-channel1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    28677  (priority 28672 sys-id-ext 5)
             Address     00d0.ff88.6000
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time 300

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- --------------------------------
Po271            Desg FWD 3         128.1665 P2p
Po1              Root FWD 1         128.1667 P2p
```

A typical concern with bridging devices after a failover is the presence of stale MAC entries in the Layer 2 table on the Catalyst 6500. The FWSM addresses this problem by storing a table with all the MAC addresses that were learned on the active FWSM and by flooding dummy Layer 2 frames with source MAC addresses of the devices that are present on the opposite interface. This quickly updates the Layer 2 tables so the traffic is not impacted.

Connections are also replicated between the active and standby device so that existing connections are forwarded by the secondary device when it becomes active.

Configuration Listings

This section includes the following topics:

- FWSM1 Configuration
- MSFC-AGG1 Configuration
- MSFC-AGG2 Configuration

FWSM1 Configuration

This section lists various configurations for the FWSM1.

System Context

```
firewall transparent
mode multiple
!
! LOGIN AND ENABLE PASSWORD
! ========================
!
enable password P1%3N@8l3
username pixadmin password P1%C1sC0!
!
hostname FWSM
domain-name example.com
class default
  limit-resource Mac-addresses 65535
  limit-resource All 0
  limit-resource IPSec 5
  limit-resource SSH 5
  limit-resource Telnet 5
!
! failover
!
failover lan unit primary
failover lan interface ft vlan 201
failover polltime unit 1 holdtime 3
failover polltime interface 3
failover interface-policy 50%
failover replication http
failover interface ip ft 10.20.201.1 255.255.255.0 standby 10.20.201.2
failover link state vlan 202
failover interface ip state 10.20.202.1 255.255.255.0 standby 10.20.202.2
!
arp timeout 14400
!
admin-context admin
context admin
  allocate-interface vlan82
  config-url disk:/admin.cfg
!
context webapp
  allocate-interface vlan5
  allocate-interface vlan105
  config-url disk:/webapp.cfg
!
context database
  member database
  allocate-interface vlan10
  allocate-interface vlan110
  config-url disk:/database.cfg
!
```

Admin Context

```
enable password P1%3N@8l3
username pixadmin password P1%C1sC0!
!
nameif vlan82 inside security100
!
ip address  10.20.26.10 255.255.255.0
route inside 0.0.0.0 0.0.0.0 10.20.26.1
!
! SSH configuration for OOB mgmt
! ------------------------------
!
domain-name example.com
crypto ca generate rsa key 1024
crypto ca save all
ssh 10.20.26.0 255.255.255.0 inside
ssh timeout 60
!
icmp permit any inside
!
```

Web and Application Context

```
firewall transparent
!
enable password P1%3N@8l3
username pixadmin password P1%C1sC0
!
hostname webapp
!
!
nameif vlan5 outside security0
nameif vlan105 inside security100
!
ip address  10.20.5.4 255.255.255.0
route outside 0.0.0.0 0.0.0.0 10.20.5.1
!
! SSH configuration for inband mgmt
! (e.g. from IDS sensors)
!
domain-name example.com
crypto ca generate rsa key 1024
crypto ca save all
ssh 10.20.26.0 255.255.255.0 outside
ssh timeout 60
!
! NTP
!
! No need to configure NTP, the FWSM syncs its clock from the 6500's
!
! LOGGING
!
logging on
logging timestamp
no logging console
no logging monitor
logging buffered informational
logging queue 32768
logging trap informational
logging host outside <syslog server>
logging device-id hostname
!
fixup protocol icmp
!
! Enable only for troubleshooting
! -------------------------------
! icmp permit any outside
! icmp permit any inside
!
! INBOUND FILTERING
!
access-list portal-in remark  >> MSFC IP addresses allowed <<
access-list portal-in extended permit ip 10.20.5.1 255.255.255.255 10.20.5.0 255.255.255.0
access-list portal-in extended permit ip 10.20.5.2 255.255.255.255 10.20.5.0 255.255.255.0
access-list portal-in extended permit ip 10.20.5.3 255.255.255.255 10.20.5.0 255.255.255.0
access-list portal-in remark .
access-list portal-in remark >> allow CSM probes to monitor the servers <<
access-list portal-in extended permit ip 10.20.44.0 255.255.255.0 10.20.5.0 255.255.255.0
access-list portal-in remark .
access-list portal-in remark >> antispoofing <<
!
! By default, when traffic is denied by an extended ACE,
! the FWSM generates system message 106023. The log option
! allows you to enable message 106100 instead of message 106023
!
access-list portal-in extended deny ip 10.20.5.0 255.255.255.0 any log 4
access-list portal-in remark .
access-list portal-in remark >> prevent exploitation of directed broadcast <<
access-list portal-in extended deny icmp any 10.20.5.255 255.255.255.255
access-list portal-in extended deny tcp any 10.20.5.255 255.255.255.255
!
! For connectionless protocols such as ICMP you either need
! ACLs to allow ICMP in both directions or you need to enable the ICMP
! inspection engine with the 2.3 code
!
access-list portal-in remark .
access-list portal-in remark >> allow ICMP to function
access-list portal-in extended permit icmp any 10.20.5.0 255.255.255.0 echo
access-list portal-in extended permit icmp any 10.20.5.0 255.255.255.0 echo-reply
access-list portal-in extended permit icmp any 10.20.5.0 255.255.255.0 time-exceeded
access-list portal-in extended permit icmp any 10.20.5.0 255.255.255.0 unreachable
access-list portal-in remark .
access-list portal-in remark >> allow access to web-based applications
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq 80
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq 8080
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq 443
!
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq ftp-data
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq ftp
!
access-list portal-in remark .
access-list portal-in remark >>  messaging applications (add the port number information)
<<
access-list portal-in extended permit udp any 10.20.5.255 255.255.255.255
access-list portal-in remark .
access-list portal-in remark >>  allow SSH, SNMP traffic (if carried inband) <<
access-list portal-in extended permit tcp any 10.20.5.0 255.255.255.0 eq 22
access-list portal-in extended permit udp any 10.20.5.0 255.255.255.0 eq 22
access-list portal-in extended permit udp any 10.20.5.0 255.255.255.0 eq 161
```

```
access-list portal-in extended permit udp any 10.20.5.0 255.255.255.0 eq 162
access-list portal-in remark the implicit deny doesn't generate a log
access-list portal-in extended deny ip any any log 4
!
access-list portal-out extended permit udp 10.20.5.0 255.255.255.0 host <DNS server> eq 53
access-list portal-out extended permit tcp 10.20.5.0 255.255.255.0 host <DNS server> eq 53
access-list portal-out extended permit udp 10.20.5.0 eq 123 host <NTP server> eq 123
access-list portal-out extended permit tcp 10.20.5.0 255.255.255.0 host <database server> 
eq 1434
access-list portal-out extended permit tcp 10.20.5.0 255.255.255.0 host <database server> 
eq 1433
access-list portal-out extended permit tcp 10.20.5.0 255.255.255.0 host <database server> 
eq 153
access-list portal-out extended permit udp 10.20.5.0 255.255.255.0 host <database server> 
eq 153
access-list portal-out remark the implicit deny doesn't generate a log
access-list portal-out extended deny ip any any log 4
!
access-group portal-in in interface outside
access-group portal-out in interface inside
!
static (inside,outside) 10.20.5.0 10.20.5.0 netmask 255.255.255.0 tcp 0 1000
!
fragment size 200 inside
!
! IF NO MONITOR SESSION SERVICE MODULE
!
access-list BPDU ethertype permit bpdu
access-group BPDU in interface outside
access-group BPDU in interface inside
!
```

Database Context

```
firewall transparent
!
enable password P1%3N@8l3
username pixadmin password P1%C1sC0!
!
hostname database
!
nameif vlan10 outside security0
nameif vlan110 inside security100
!
ip address  10.20.10.4 255.255.255.0
route outside 0.0.0.0 0.0.0.0 10.20.10.1
!
! SSH configuration for inband mgmt
! (e.g. from IDS sensors)
!
domain-name example.com
crypto ca generate rsa key 1024
crypto ca save all
ssh 10.20.26.0 255.255.255.0 outside
ssh timeout 60
!
! NTP
!
! No need to configure NTP, the FWSM syncs its clock from the 6500's
!
! LOGGING
!
```

```
logging on
logging timestamp
no logging console
no logging monitor
logging buffered informational
logging queue 32768
logging trap informational
logging host outside <syslog server>
logging device-id hostname
!
fixup protocol icmp
!
!
! Enable only for troubleshooting
! ------------------------------
! icmp permit any outside
! icmp permit any inside
!
!
access-list database-in remark  >> MSFC IP addresses allowed <<
access-list database-in extended permit ip 10.20.10.1 255.255.255.255 10.20.10.0
255.255.255.0
access-list database-in extended permit ip 10.20.10.2 255.255.255.255 10.20.10.0
255.255.255.0
access-list database-in extended permit ip 10.20.10.3 255.255.255.255 10.20.10.0
255.255.255.0
access-list database-in remark .
access-list database-in remark >> allow CSM probes to monitor the servers <<
access-list database-in extended permit ip 10.20.44.0 255.255.255.0 10.20.10.0
255.255.255.0
access-list database-in remark .
access-list database-in remark >> antispoofing <<
access-list database-in extended deny ip 10.20.10.0 255.255.255.0 any
access-list database-in remark .
access-list database-in remark >> prevent exploitation of directed broadcast <<
access-list database-in extended deny icmp any 10.20.10.255 255.255.255.255
access-list database-in extended deny tcp any 10.20.10.255 255.255.255.255 log 4
access-list database-in remark .
access-list database-in remark >> allow ICMP to function
access-list database-in extended permit icmp any 10.20.5.0 255.255.255.0 echo
access-list database-in extended permit icmp any 10.20.5.0 255.255.255.0 echo-reply
access-list database-in extended permit icmp any 10.20.5.0 255.255.255.0 time-exceeded
access-list database-in extended permit icmp any 10.20.5.0 255.255.255.0 unreachable
access-list database-in remark .
access-list database-in remark >> allow access to the database <<
access-list database-in extended permit tcp 10.20.5.0 255.255.255.0 host 10.20.10.115 eq
1433
access-list database-in extended permit tcp 10.20.5.0 255.255.255.0 host 10.20.10.115 eq
1434
access-list database-in extended permit tcp 10.20.5.0 255.255.255.0 host 10.20.10.115 eq
153
access-list database-in extended permit udp 10.20.5.0 255.255.255.0 host 10.20.10.115 eq
153
access-list database-in remark .
access-list database-in remark >>  messaging applications (add the port number
information) <<
access-list database-in extended permit udp any 10.20.10.255 255.255.255.255
access-list database-in remark .
access-list database-in remark >>  allow SSH, SNMP traffic (if carried inband) <<
access-list database-in extended permit tcp any 10.20.10.0 255.255.255.0 eq 22
access-list database-in extended permit udp any 10.20.10.0 255.255.255.0 eq 22
access-list database-in extended permit udp any 10.20.10.0 255.255.255.0 eq 161
access-list database-in extended permit udp any 10.20.10.0 255.255.255.0 eq 162
access-list database-in remark the implicit deny doesn't generate a log
```

```
access-list database-in extended deny ip any any log 4
!
! OUTBOUND FILTERING
!
access-list database-out extended permit udp 10.20.10.0 255.255.255.0 host <DNS server> eq 
53
access-list database-out extended permit tcp 10.20.10.0 255.255.255.0 host <DNS server> eq 
53
access-list database-out extended permit udp 10.20.10.0 255.255.255.0 eq 123 host <NTP 
server> eq 123
access-list database-out remark the implicit deny doesn't generate a log
access-list database-out extended deny ip any any log 4
!
! IF NO MONITOR SESSION SERVICE MODULE
!
access-list BPDU ethertype permit bpdu
access-group BPDU in interface outside
access-group BPDU in interface inside
!
```

MSFC-AGG1 Configuration

```
!
hostname agg1
!
firewall multiple-vlan-interfaces
firewall module 3 vlan-group 3
firewall vlan-group 3  5,10,82,105,110,201,202
!
! VTP and Spanning-Tree
! ====================
!
vtp domain mydomain
vtp mode transparent
!
spanning-tree mode rapid-pvst
no spanning-tree loopguard default
spanning-tree extend system-id
spanning-tree vlan 1-1000 root primary
spanning-tree pathcost method long
!
! VLAN CONFIGURATION
!
vlan internal allocation policy descending
!
vlan 5
 name webappoutside
!
vlan 10
 name databaseoutside
!
vlan 82
 name networkmgmt
!
vlan 105
 name webappinside
!
vlan 110
 name databaseinside
!
vlan 201
 name fwsm_failover_vlan
```

```
!
vlan 202
 name fwsm_flink
!
interface Port-channel1
 no ip address
 switchport
 switchport trunk encapsulation dot1q
 spanning-tree guard loop
 switchport mode trunk
 switchport nonegotiate
! >> use a != native VLANs on trunks than on access ports <<
 switchport trunk native vlan 2
! >> do not trunk VLAN 13, 14, 82<<
 switchport trunk allowed vlan 5,10,30,44,45,100,105,110,201,202,300
 no shut
!
! SVI CONFIGURATION
! =================
!
interface Vlan5
 description webapp
 ip address 10.20.5.2 255.255.255.0
 standby 1 ip 10.20.5.1
 standby 1 timers 1 3
 standby 1 priority 120
 standby 1 preempt delay minimum 180
 ! If need directed broadcast:
 ! ip directed-broadcast
 ! mls ip directed-broadcast exclude-router
 no ip unreachables
 no ip redirects
 no ip proxy-arp
 ! >> Disable NTP services <<
 ntp disable
 no shut
!
interface Vlan10
 description database
 ip address 10.20.10.2 255.255.255.0
 standby 1 ip 10.20.10.1
 standby 1 timers 1 3
 standby 1 priority 120
 standby 1 preempt delay minimum 180
 ! If need directed broadcast:
 ! ip directed-broadcast
 ! mls ip directed-broadcast exclude-router
 no ip unreachables
 no ip redirects
 no ip proxy-arp
 ! >> Disable NTP services <<
 ntp disable
 no shut
!
! If no multicast source protected by FWSM
!
no monitor session servicemodule
```

MSFC-AGG2 Configuration

```
!
hostname agg2
```

```
firewall multiple-vlan-interfaces
firewall module 3 vlan-group 3
firewall vlan-group 3  5,10,82,105,110,201,202
firewall module 3 vlan-group 3
!
! VTP and Spanning-Tree
! ====================
!
vtp domain mydomain
vtp mode transparent
!
spanning-tree mode rapid-pvst
no spanning-tree loopguard default
spanning-tree extend system-id
spanning-tree vlan 1-1000 root secondary
!
! VLAN CONFIGURATION
!
vlan internal allocation policy descending
!
vlan 5
 name webappoutside
!
vlan 10
 name databaseoutside
!
vlan 82
 name networkmgmt
!
vlan 105
 name webappinside
!
vlan 110
 name databaseinside
!
vlan 201
 name fwsm_failover_vlan
!
vlan 202
 name fwsm_flink
!
interface Port-channel1
 no ip address
 switchport
 switchport trunk encapsulation dot1q
 spanning-tree guard loop
 switchport mode trunk
 switchport nonegotiate
! >> use a != native VLANs on trunks than on access ports <<
 switchport trunk native vlan 2
! >> do not trunk VLAN 13 , 14 , 82 <<
 switchport trunk allowed vlan 5,10,30,44,45,100,105,110,201,202,300
 no shut
!
! SVI CONFIGURATION
! ================
!
! ip directed-broadcast often needed
! in serverfarms disable it if possible
!
interface Vlan5
 description webapp
 ip address 10.20.5.3 255.255.255.0
 standby 1 ip 10.20.5.1
 standby 1 timers 1 3
 standby 1 priority 110
 standby 1 preempt delay minimum 180
 ! If need directed broadcast:
 ! ip directed-broadcast
 ! mls ip directed-broadcast exclude-router
 no ip unreachables
 no ip redirects
 no ip proxy-arp
! >> Disable NTP services <<
 ntp disable
 no shut
!
interface Vlan10
 description database
 ip address 10.20.10.3 255.255.255.0
 standby 1 ip 10.20.10.1
 standby 1 timers 1 3
 standby 1 priority 110
 standby 1 preempt delay minimum 180
 ! If need directed broadcast:
 ! ip directed-broadcast
 ! mls ip directed-broadcast exclude-router
 no ip unreachables
 no ip redirects
 no ip proxy-arp
! >> Disable NTP services <<
 ntp disable
 no shut
! If no multicast source protected by FWSM
!
no monitor session servicemodule
```