13 March 2017

ClusterXL

R80.10

Administration Guide

Check Point
SOFTWARE TECHNOLOGIES LTD.

Important Information

Latest Software

We recommend that you install the most recent software release to stay up-to-date with the latest functional improvements, stability fixes, security enhancements and protection against new and evolving attacks.

Check Point R80.10

For more about this release, see the R80.10 home page http://supportcontent.checkpoint.com/solutions?id=sk111841.

More Information

Visit the Check Point Support Center http://supportcenter.checkpoint.com.

Latest Version of this Document

Download the latest version of this document http://supportcontent.checkpoint.com/documentation_download?ID=TBD.

To learn more, visit the Check Point Support Center http://supportcenter.checkpoint.com.

Feedback

Check Point is engaged in a continuous effort to improve its documentation.

Please help us by sending your comments mailto:cp_techpub_feedback@checkpoint.com?subject=Feedback on ClusterXL R80.10 Administration Guide.

Searching in Multiple PDFs

To search for text in all the R80.10 PDF documents, download and extract the complete R80.10 documentation package http://supportcontent.checkpoint.com/documentation_download?ID=TBD.
Use **Shift-Control-F** in Adobe Reader or Foxit reader.

Revision History

Date	Description
13 March 2017	First release of this document

Contents

Terms

Active Member

A cluster member that handles network connections. In a High Availability deployment only one member can handle connections. In a Load Sharing deployment, all members are active and can handle connections.

Active Up

ClusterXL High Availability mode that is configured as **Maintain current active Cluster Member**. If the current **Active** member fails or reboots, failover occurs and a Standby member becomes the **Active** member. When the failed member is restored, it becomes the **Standby** member.

Active/Standby

A High Availability cluster where only one member handles connections.

Bond

A virtual interface that contains ("enslaves") two or more physical interfaces for redundancy and load sharing. The physical interfaces share one IP address and one MAC address.

Bridge Mode

A Security Gateway or Virtual System that works as a Layer-2 bridge device for easy deployment in an existing topology.

Cluster

1. Two or more Security Gateways or servers synchronized for High Availability or Load Sharing.

2. In a virtualized environment - a set of ESX/i hosts used for High Availability or Load Sharing.

Cluster Member

A Security Gateway that is part of a cluster.

ClusterXL

Check Point software-based cluster solution for Security Gateway redundancy and Load Sharing.

Failback

The act of restoring a system, component, or service in a state of failover back to its original state.

Failed Member

A cluster member that cannot send or accept traffic because of a hardware or software problem.

Failover

A redundancy operation where one cluster member automatically takes over for a failed member.

High Availability Cluster

A redundant cluster where a failed active member automatically fails over to a standby member for continuous operation.

Link Aggregation

A technology that joins multiple physical interfaces together into one virtual interface, known as a bond interface. Also known as interface bonding.

Load Sharing Cluster

A cluster with two or more members that share the traffic load between them. Load Sharing clusters typically have High Availability redundancy as well.

Management Server

A Security Management Server or Multi-Domain Server that manages one or more Security Gateways and security policies.

Multicast Load Sharing

All cluster members get packets set to the cluster at the same time.

Non-Sticky Connection

Connection packets for Load Sharing that are handled by different cluster members, typically based on the traffic direction.

Pivot Member

A member that is currently associated with the cluster virtual IP address in a unicast Load Sharing cluster. This is the only member that can get packets sent to the cluster. The pivot member then sends the packets to other members for Load Sharing.

Primary Up

ClusterXL High Availability member that is configured as **Switch to higher priority Cluster Member**. The member with highest priority automatically becomes the **Active** member. If the **Active** member fails or reboots, failover occurs and the member with the next highest priority becomes the **Active** member. When the failed member is restored, it again becomes **Active** only if it has a higher priority. In this case, the replaced member becomes a **Standby** member.

Security Gateway

A computer or an appliance that inspects traffic and enforces Security Policies for connected network resources.

Security Management Server

The server that manages, creates, stores, and distributes the security policy to Security Gateways.

SmartDashboard

A legacy Check Point client used to create and manage the security policy.

Standby Member

A cluster member that is not active and does not handle connections.

Sticky Connection

A connection where all packets (in both directions) are handled by one cluster member.

Traffic

The flow of data between network resources.

Unicast Load Sharing

One cluster member (pivot member) gets all packets and sends them to the other members.

VLAN

Virtual Local Area Network. Open servers or appliances connected to a virtual network which are not physically connected to the same network.

VLAN Trunk

A connection between two switches that contains multiple VLANs.

Introduction to ClusterXL

In This Section:

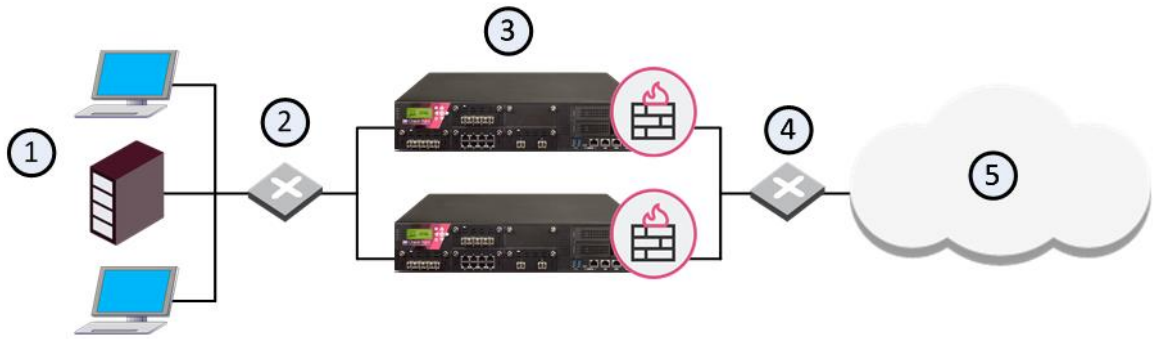
The Need for Clusters

Security Gateways and VPN connections are business critical devices. The failure of a Security Gateway or VPN connection can result in the loss of active connections and access to critical data. The Security Gateway between the organization and the world must remain open under all circumstances.

ClusterXL Solution

ClusterXL is a Check Point software-based cluster solution for Security Gateway redundancy and Load Sharing. A ClusterXL Security Cluster contains identical Check Point Security Gateways.

- A High Availability Security Cluster ensures Security Gateway and VPN connection redundancy by providing transparent failover to a backup Security Gateway in the event of failure.
- A Load Sharing Security Cluster provides reliability and also increases performance, as all members are active



Item	Description
1	Internal network
2	Switch for internal network
3	Security Gateways with ClusterXL Software Blade
4	Switch for external networks

Item	Description
5	Internet

IPv6 Support for ClusterXL

R80.10 ClusterXL supports High Availability clusters for IPv6. IPv6 status information is synchronized and the IPv6 clustering mechanism is activated during failover.

You can define IPv6 addresses for:

- Cluster virtual interfaces
- Member physical interfaces

Limitations

- IPv6 is not supported for Load Sharing clusters.
- You cannot define IPv6 address for synchronization interfaces.

ClusterXL High Availability

During failover, a cluster sends gratuitous ARP request packets to update hosts and routers connected to cluster interfaces. It does this by advertising the new MAC address for the virtual cluster IPv4 addresses.

ClusterXL updates the IPv6 network during failovers. ClusterXL sends Neighbor Advertisement messages to update the neighbor cache (which is equivalent to the ARP cache in IPv4) by advertising the new MAC address for the virtual cluster IPv6 address. In addition, ClusterXL will reply to any Neighbor Solicitation with a target address equal to the Virtual Cluster IPv6 address.

Note - ClusterXL failover event detection is based on IPv4 probing. During state transition the IPv4 driver instructs the IPv6 driver to reestablish IPv6 network connectivity to the HA cluster.

How ClusterXL Works

ClusterXL uses *State Synchronization* to keep active connections alive and prevent data loss when a member fails. With State Synchronization, each member "knows" about connections that go through other members.

ClusterXL uses virtual IP addresses for the cluster itself and unique physical IP and MAC addresses for the members. Virtual IP addresses do not belong to physical interfaces.

ClusterXL can work with OPSEC certified High Availability and Load Sharing products, which use the same State Synchronization infrastructure as Check Point ClusterXL.

Note - The *ClusterXL Administration Guide* contains information only for Security Gateway clusters. For information about the use of ClusterXL with VSX, see the *R80.10 VSX Administration Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD.

The Cluster Control Protocol

The Cluster Control Protocol (CCP) is the glue that links together the members in the Security Cluster. CCP traffic is distinct from ordinary network traffic and can be viewed using any network sniffer.

CCP runs on UDP port 8116, and has the following roles:

- It allows cluster members to report their own states and learn about the states of other members by sending keep-alive packets (this only applies to ClusterXL clusters).
- State Synchronization.

The Check Point CCP is used by all ClusterXL modes as well as by OPSEC clusters. However, the tasks performed by this protocol and the manner in which they are implemented may differ between cluster types.

Note - There is no need to add a rule to the Security Policy Rule Base that accepts CCP

Installation and Platform Support

ClusterXL must be installed in a distributed configuration in which the Security Management Server and the Security Cluster members are on different computers. ClusterXL is part of the standard Security Gateway installation.

For installation instructions, see the *R80.10 Installation and Upgrade Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD.

To see the ClusterXL supported platforms, see the *R80.10 Release Notes* http://supportcontent.checkpoint.com/documentation_download?ID=TBD .

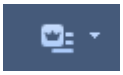
Clock Synchronization in ClusterXL

When using ClusterXL, make sure to synchronize the clocks of all of the cluster members. You can synchronize the clocks manually or using a protocol such as NTP. Features such as VPN only function properly when the clocks of all of the members are synchronized.

SmartConsole Toolbars

For a guided tour of SmartConsole, click **What's New** in the left bottom corner of SmartConsole.

Global Toolbar (top left of SmartConsole)

	Description and Keyboard Shortcut
	The main SmartConsole Menu
	The **Objects** menu. Also leads to the Object Explorer **Ctrl+E**

	Description and Keyboard Shortcut
	Install policy on managed gateways **Ctrl+Shift+Enter**

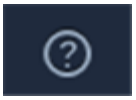
Navigation Toolbar (left side of SmartConsole)

	Description and Keyboard Shortcut
	Gateway configuration view **Ctrl+1**
	Security Policies Access Control view Security Policies Threat Prevention view **Ctrl+2**
	Logs & Monitor view **Ctrl+3**
	Manage & Settings view - review and configure the Security Management Server settings **Ctrl+4**

Command Line Interface Button (left bottom corner of SmartConsole)

	Description and Keyboard Shortcut
	Open a command line interface for management scripting and API **F9**

What's New Button (left bottom corner of SmartConsole)

	Description and Keyboard Shortcut
	Open a tour of the SmartConsole

Objects and Validations Tabs (right side of SmartConsole)

	Description
Objects	Manage security and network objects
Validations	Validation warnings and errors

System Information Area (bottom of SmartConsole)

	Description
Task List	Management activities, such as policy installation tasks

	Description
Server Details	The IP address of the Security Management Server
Connected Users	The administrators that are connected to the Security Management Server

Synchronizing Connections in the Cluster

In This Section:

The Check Point State Synchronization Solution

A failure of a firewall results in an immediate loss of active connections in and out of the organization. Many of these connections, such as financial transactions, may be mission critical, and losing them will result in the loss of critical data. ClusterXL supplies an infrastructure that ensures that no data is lost in case of a failure, by making sure each cluster member is aware of the connections going through the other members. Passing information about connections and other Security Gateway states between the cluster members is called State Synchronization.

Every IP based service (including TCP and UDP) recognized by the Security Gateway is synchronized.

State Synchronization is used both by ClusterXL and by third-party OPSEC-certified clustering products.

Members in a ClusterXL Load Sharing configuration must be synchronized. Members in a ClusterXL High Availability configuration do not have to be synchronized, though if they are not, connections will be lost upon failover.

The Synchronization Network

The Synchronization Network is used to transfer synchronization information about connections and other Security Gateway states between cluster members.

Since the synchronization network carries the most sensitive Security Policy information in the organization, it is critical that you protect it against both malicious and unintentional threats. We recommend that you secure the synchronization interfaces using one of the following strategies:

- Use a dedicated synchronization network
- Connecting the physical network interfaces of the cluster members directly using a cross-cable. In a cluster with three or more members, use a dedicated hub or switch

Note -You can synchronize members across a WAN. To do this, do the steps in Synchronizing Clusters on a WAN (see "Synchronizing Clusters on a Wide Area Network" on page 24).

These recommendations make the synchronization network more secure because no other networks carry synchronization information.

In ClusterXL, the synchronization network is supported on the lowest VLAN tag of a VLAN interface. For example, if three VLANs with tags *10*, *20* and *30* are configured on interface *eth1*, interface *eth1.10* may be used for synchronization.

How State Synchronization Works

Synchronization works in two modes:

- *Full sync* transfers all Security Gateway kernel table information from one cluster member to another. It is handled by the **fwd** daemon using an encrypted TCP connection.
- *Delta sync* transfers *changes* in the kernel tables between cluster members. Delta sync is handled by the Security Gateway kernel using UDP multicast or broadcast on port 8116.

Full sync is used for initial transfers of state information, for many thousands of connections. If a cluster member is brought up after being down, it will perform full sync. After all members are synchronized, only updates are transferred via delta sync. Delta sync is quicker than full sync.

State Synchronization traffic typically makes up around 90% of all Cluster Control Protocol (CCP) traffic. State Synchronization packets are distinguished from the rest of CCP traffic via an opcode in the UDP data header.

Note - The source MAC address for CCP packets can be changed (see "Synchronizing Clusters on a Wide Area Network" on page 24).

Non-Synchronized Services

In a cluster, all connections on all cluster members are normally synchronized across the cluster. Not all services that go through a cluster must be synchronized.

- You can decide not to synchronize TCP, UDP and other service types. By default, all these services are synchronized.
- The VRRP and IP Clustering control protocols, and the IGMP protocol, are not synchronized by default (but you can choose to turn on synchronization for these protocols). Protocols that run solely between cluster members need not be synchronized. Although you can synchronize them, no benefit will be gained. This synchronization information will not help a failover. These protocols are not synchronized by default: IGMP, VRRP, IP clustering and some other OPSEC cluster control protocols.
- Broadcasts and multicasts are not synchronized, and cannot be synchronized.

You can have a synchronized service and a non-synchronized definition of a service, and use them selectively in the Rule Base.

Configuring Services not to Synchronize

Synchronization incurs a performance cost. You may choose not to synchronize a service if these conditions are true:

- A significant amount of traffic goes through the cluster for a service. Not synchronizing the service reduces the amount of synchronization traffic and so enhances cluster performance.
- The service typically opens short connections, whose loss may not be noticed. DNS (over UDP) and HTTP are typically responsible for most connections and frequently have short life and inherent recoverability in the application level. Services that open long connections, such as FTP, should always be synchronized.
- Configurations that ensure bi-directional stickiness for all connections do not require synchronization to operate (only to maintain High Availability). Such configurations include:
  - Any cluster in High Availability mode (for example, ClusterXL New HA or IPSO VRRP).

- ClusterXL in a Load Sharing mode with clear connections (no VPN or static NAT).
- OPSEC clusters that guarantee full stickiness (refer to the OPSEC cluster documentation).
- VPN and Static NAT connections passing through a ClusterXL cluster in a Load Sharing mode (either multicast or unicast) may not maintain bi-directional stickiness. State Synchronization must be turned on for such environments.

Duration Limited Synchronization

Some TCP services (HTTP for example) are characterized by connections with a very short duration. There is no point in synchronizing these connections because every synchronized connection consumes Security Gateway resources, and the connection is likely to have finished by the time a failover occurs.

For all TCP services whose Protocol Type (that is defined in the GUI) is HTTP or None, you can use this option to delay telling the Security Gateway about a connection, so that the connection will only be synchronized if it still exists x seconds after the connection is initiated. This feature requires a SecureXL device that supports "Delayed Notifications" and the current cluster configuration (such as Performance Pack with ClusterXL LS Multicast).

This capability is only available if a SecureXL-enabled device is installed on the Security Gateway through which the connection passes.

The setting is ignored if connection templates are not offloaded from the ClusterXL-enabled device. See the SecureXL documentation for additional information.

Sticky Connections

In This Section:

Introduction to Sticky Connections

A connection is considered **sticky** when all of its packets are handled, in either direction, by a single cluster member. This is the case in High Availability mode, where all connections are routed through the same cluster member, and hence, sticky. This is also the case in Load Sharing mode when there are no VPN peers, static NAT rules or SIP.

In Load Sharing mode, however, there are cases where it is necessary to ensure that a connection that starts on a specific cluster member will continue to be processed by the same cluster member in both directions. To that end, certain connections can be made sticky by enabling the Sticky Decision Function.

The Sticky Decision Function

The Sticky Decision Function enables certain services to operate in a Load Sharing deployment. For example, it is required for L2TP traffic, or when the cluster is a participant in a site to site VPN tunnel with a third party peer.

The following services and connection types are now supported by enabling the Sticky Decision Function:

- VPN deployments with third-party VPN peers
- SecureClient /SecuRemote/ SSL Network Extender encrypted connections, including SecureClient visitor mode

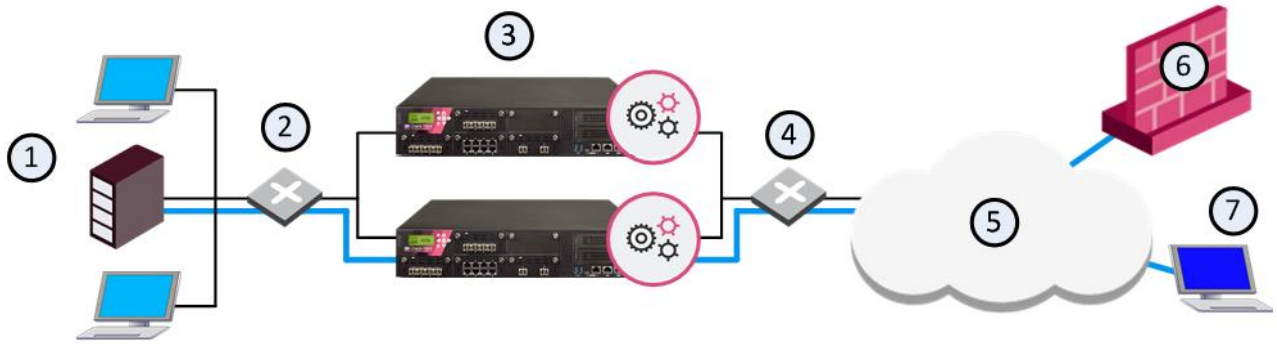
The Sticky Decision Function has the following limitations:

- Sticky Decision Function is not supported when employing either Performance Pack or a hardware-based accelerator card. Enabling the Sticky Decision Function disables these acceleration products.
- When the Sticky Decision Function is used in conjunction with VPN, cluster members are prevented from opening more than one connection to a specific peer. Opening another connection would cause another SA to be generated, which a third-party peer, in many cases, would not be able to process.

VPN Tunnels with 3rd Party Peers and Load Sharing

Check Point provides interoperability with third-party vendor gateways by enabling them to peer with Check Point gateways. A special case occurs when certain third-party peers (Microsoft LT2P, IPSO Symbian, and Cisco gateways and clients) attempt to establish VPN tunnels with ClusterXL Security Gateways in the Load Sharing mode. These peers are limited in their ability to store SAs,

which means that a VPN session that begins on one cluster member and, due to Load Sharing, is routed on the return trip through another, is unrecognized and dropped.



Item	Description
1	Internal network
2	Switch for internal network
3	Security Gateways with ClusterXL Software Blade
4	Switch for external networks
5	Internet
6	3rd party peer VPN gateway
7	3rd party peer laptop with VPN client

In this scenario:

- A third-party peer (gateway or client) attempts to create a VPN tunnel.
- Cluster Members A and B belong to a ClusterXL Security Gateway in Load Sharing mode.

The third-party peers, lacking the ability to store more than one set of SAs, cannot negotiate a VPN tunnel with multiple cluster members, and therefore the cluster member cannot complete the routing transaction.

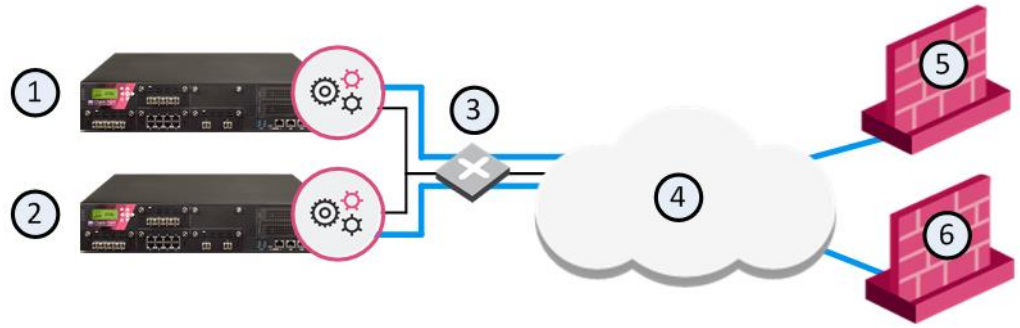
This issue is resolved for certain third-party peers or gateways that can save only one set of SAs by making the connection sticky. Enabling the Sticky Decision Function sets all VPN sessions initiated by the same third-party gateway to be processed by a single cluster member.

To enable the Sticky Decision Function:

1. In SmartConsole edit the cluster object **> ClusterXL** page **> Advanced.**
2. Enable the property **Use Sticky Decision Function**.

Third-Party Gateways in Hub and Spoke Deployments

Another case where Load Sharing mode requires the Sticky Decision Function is when integrating certain third-party gateways into a hub and spoke deployment. Without the ability to store more than one set of SAs, a third-party gateway must maintain its VPN tunnels on a single cluster member in order to avoid duplicate SAs.



Item	Description
1	Security Gateway - Cluster member A
2	Security Gateway - Cluster member B
3	Switch for external networks
4	Internet
5	Gateway - Spoke A
6	Gateway - Spoke B

In this sample deployment:

- The intent of this deployment is to enable hosts that reside behind Spoke A to communicate with hosts behind Spoke B.
- The ClusterXL Security Gateway is in Load Sharing mode, is composed of Cluster Members A and B, and serves as a VPN Hub.
- Spoke A is a third-party gateway, and is connected by a VPN tunnel that passes through the Hub to Spoke B.
- Spoke B can be either another third-party gateway or a Check Point Security Gateway.

Spokes A and B must be set to always communicate using the same cluster member. Enabling the Sticky Decision Function solves half of this problem, in that all VPN sessions initiated by either third-party gateway are processed by a single cluster member.

To make sure that all communications between Spokes A and B are always using the same cluster member, you must make some changes to the **user.def** file. This second step ensures that both third-party gateways always connect to the same cluster member (see "Establishing a Third-Party Gateway in a Hub and Spoke Deployment" on page 22).

Configuring the Sticky Decision Function

To configure the Sticky Decision Function:

1. Select a cluster object from the **Network Object** tree.

2. Select **ClusterXL** from the tree and click **Advanced**.
3. In the **Advanced Load Sharing** window, select one of the following options:
   - **IPs, Ports, SPIs** (default) provides the best sharing distribution, and is recommended for use. It is the least "sticky" sharing configuration.
   - **IPs, Ports** should be used only if problems arise when distributing IPsec packets to a few members although they have the same source and destination IP addresses.
   - **IPs** should be used only if problems arise when distributing IPsec packets or different port packets to a few members although they have the same source and destination IP addresses. It is the most "sticky" sharing configuration.
4. Enable the **Sticky Decision Function** option to enable its functionality or clear to disable. By default the Sticky Decision Function is disabled.

   If enabled, the connection will pass through a single cluster member on both inbound and outbound directions.

Establishing a Third-Party Gateway in a Hub and Spoke Deployment

To establish a third-party gateway as a spoke in a hub and spoke deployment, perform the following on the Security Management Server:

1. Enable the Sticky Decision Function if not already enabled. In SmartConsole, edit the cluster object **> ClusterXL** page **> Advanced**, and enable the property **Use Sticky Decision Function**.
2. Create a Tunnel Group to handle traffic from specific peers. Use a text editor to edit the file **$FWDIR/lib/user.def**, and add a line similar to the following:

```
all@{member1,member2} vpn_sticky_gws = {<10.10.10.1;1>,
<20.20.20.1;1>};
```

   The elements of this configuration are as follows:
   - **all** - All cluster interfaces
   - **member1,member2** - Names of cluster members in SmartConsole
   - **vpn_sticky_gws** - Table name
   - **10.10.10.1** - IP address of Spoke A
   - **20.20.20.1** - IP address of Spoke B
   - **;1** - Tunnel Group Identifier, which indicates that the traffic from these IP addresses should be handled by the same cluster member
3. Other peers can be added to the Tunnel Group by including their IP addresses in the same format as shown above. To continue with the example above, adding Spoke C would look like this:

```
all@{member1,member2} vpn_sticky_gws = {<10.10.10.1;1>,
<20.20.20.1;1>,<30.30.30.1;1>};
```

   Note that the Tunnel Group Identifier **;1** stays the same, which means that the listed peers will always connect through the same cluster member.

   Note - More tunnel groups than cluster members may be defined.

   This procedure in essence turns off Load Sharing for the connections affected. If the implementation is to connect multiple sets of third-party Security Gateways one to another, a form of Load Sharing can be accomplished by setting Security Gateway pairs to work in tandem with specific cluster members. For instance, to set up a connection between two other spokes (C and D), simply add their IP addresses to the line and replace the Tunnel Group Identifier **;1** with **;2**. The line would then look something like this:

```
all@{member1,member2} vpn_sticky_gws = {<10.10.10.1;1>,
<20.20.20.1;1>,<192.168.15.5;2>,<192.168.1.4;2>,};
```

Note that there are now two peer identifiers: **;1** and **;2**. Spokes A and B will now connect through one cluster member, and Spokes C and D through another.

Note - The tunnel groups are shared between active cluster members. In case of a change in cluster state (e.g., failover or member attach/detach), the reassignment is performed according to the new state.

Non-Sticky Connections

A connection is called **sticky** if all packets are handled by a single cluster member. In a **non-sticky** connection, the reply packet returns via a different Security Gateway than the original packet.

The synchronization mechanism knows how to properly handle non-sticky connections. In a non-sticky connection, a cluster member can receive an out-of-state packet, which Security Gateway normally drops because it poses a security risk.

In Load Sharing configurations, all cluster members are active, and in Static NAT and encrypted connections, the source and destination IP addresses change. Therefore, Static NAT and encrypted connections through a Load Sharing cluster may be non-sticky. Non-stickiness may also occur with Hide NAT, but ClusterXL has a mechanism to make it sticky.

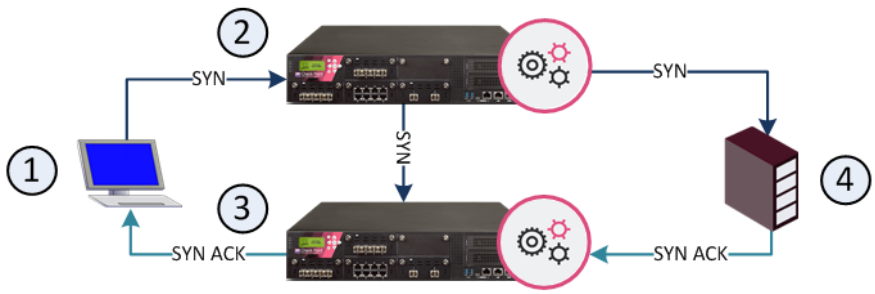
In High Availability configurations, all packets reach the Active member, so all connections are sticky. If failover occurs during connection establishment, the connection is lost, but synchronization can be performed later.

If the other members do not know about a non-sticky connection, the packet will be out-of-state, and the connection will be dropped for security reasons. However, the Synchronization mechanism knows how to inform other members of the connection. The Synchronization mechanism thereby prevents out-of-state packets in valid, but non-sticky connections, so that these non-sticky connections are allowed.

Non-sticky connections will also occur if the network Administrator has configured asymmetric routing, where a reply packet returns through a different Security Gateway than the original packet.

Non-Sticky Connection Example: TCP 3-Way Handshake

The 3-way handshake that initiates all TCP connections can very commonly lead to a non-sticky (often called asymmetric routing) connection. This diagram shows a sample scenario:



Item	Description
1	Client
2	Security Gateway - Cluster member A

Item	Description
3	Security Gateway - Cluster member B
4	Server

The client initiates a connection by sending a SYN packet to the server. The SYN passes through cluster member A, but the SYN/ACK reply returns through cluster member B. This is a non-sticky connection, because the reply packet returns through a different Security Gateway than the original packet.

The synchronization network notifies cluster member B. If cluster member B is updated before the SYN/ACK packet sent by the server reaches it, the connection is handled normally. If, however, synchronization is delayed, and the SYN/ACK packet is received on cluster member B before the SYN flag has been updated, then the Security Gateway treats the SYN/ACK packet as out-of-state, and drops the connection.

You can configure enhanced 3-Way TCP Handshake (see "Enhanced 3-Way TCP Handshake Enforcement" on page 114) enforcement to address this issue.

Synchronizing Non-Sticky Connections

The synchronization mechanism prevents out-of-state packets in valid, but non-sticky connections. The way it does this is best illustrated with reference to the 3-way handshake that initiates all TCP data connections. The 3-way handshake proceeds as follows:

1. SYN (client to server)
2. SYN/ACK (server to client)
3. ACK (client to server)
4. Data (client to server)

To prevent out-of-state packets, the following sequence (called "Flush and Ack") occurs

1. Cluster member receives first packet (SYN) of a connection.
2. Suspects that it is non-sticky.
3. Hold the SYN packet.
4. Send the pending synchronization updates to all cluster members (including all changes relating to this packet).
5. Wait for all the other cluster members to acknowledge the information in the sync packet.
6. Release held SYN packet.
7. All cluster members are ready for the SYN-ACK.

Synchronizing Clusters on a Wide Area Network

Organizations are sometimes faced with the need to locate cluster members in geographical locations that are distant from each other. A typical example is a replicated data center whose locations are widely separated for disaster recovery purposes. In such a configuration it is clearly impractical to use a cross cable as the synchronization network.

The synchronization network can be spread over remote sites, which makes it easier to deploy geographically distributed clustering. There are two limitations to this capability:

1. The synchronization network must guarantee no more than 100ms latency and no more than 5% packet loss.

2. The synchronization network may only include switches and hubs. No routers are allowed on the synchronization network, because routers drop Cluster Control Protocol packets.

You can monitor and troubleshoot (see "Troubleshooting Synchronization" on page 75) geographically distributed clusters using the command line interface.

Synchronized Cluster Restrictions

The following restrictions apply when you synchronize cluster members:

- The use of more than one synchronization interface for redundancy is not supported. You can use Link Aggregation ("Sync Redundancy" on page 102) for synchronization interface redundancy. Synchronization interface redundancy is not supported for VRRP clusters.
- All cluster members must run on identically configured platforms.
- All cluster members must use the same Check Point software version.
- If a cluster member goes down, user-authenticated connections through that member are lost. Other cluster members cannot restore the connection. Client-authenticated or session-authenticated connections are maintained.

  The reason for these restrictions is that the user authentication state is maintained by a process on the Security Gateway. It cannot be synchronized on members the same way that kernel data is synchronized. However, the states of session authentication and client authentication are saved in kernel tables, and can be synchronized.
- The connection statues that use system resources cannot be synchronized for the same reason that user-authenticated connections cannot be synchronized.
- Accounting information is accumulated on each cluster member and sent to the Security Management Server and aggregated. In the event of a failover, accounting information not yet sent to the Security Management Server is lost. To minimize this risk, you can reduce the time interval when accounting information is sent. To do this, on the cluster object **Logs and Masters > Additional Logging** page, set a lower value for the **Update Account Log every** attribute.

Configuring State Synchronization

Configure State synchronization as part of the process of configuring ClusterXL and OPSEC certified clustering products. Configuring State synchronization involves the following steps:

1. Setting up a synchronization network for the cluster
2. Installing a Security Gateway and enabling synchronization during the configuration process
3. Enabling State Synchronization on the **ClusterXL** page for the cluster object

For configuration procedures, refer to the sections for configuring ClusterXL (see "Configuring ClusterXL" on page 45) and OPSEC certified cluster products (see "Configuring OPSEC Certified Clustering Products" on page 56).

Configuring a Service Not to Synchronize

To set a service not to synchronize:

1. In the **Services** branch of the objects tree, double click the TCP, UDP or Other type service that you do not wish to synchronize.

2. In the **Service Properties** window, click **Advanced** to display the **Advanced Services Properties** window.
3. Clear **Synchronize connections on the cluster.**

Creating Synchronized and Non-Synchronized Versions

It is possible to have both a synchronized and a non-synchronized definition of the service, and to use them selectively in the Security Rule Base.

1. Define a new TCP, UDP and Other type service. Give it a name that distinguishes it from the existing service.
2. Copy all the definitions from the existing service into the **Service Properties** window of the new service.
3. In the new service, click **Advanced** to display the **Advanced Services Properties** window.
4. Copy all the definitions from the existing service into the **Advanced Service Properties** window of the new service.
5. Set **Synchronize connections on the cluster** in the new service, so that it is different from the setting in the existing service.

Configuring Duration Limited Synchronization

Before you start this procedure, become familiar with the concept (see "Duration Limited Synchronization" on page 18).

Note - This feature is limited to HTTP-based services. The **Start synchronizing** option is not displayed for other services.

To configure duration limited synchronization:

1. In the **Services** branch of the objects tree, double click the TCP, UDP or Other type service that you wish to synchronize.
2. In the **Service Properties** window, click **Advanced** to display the **Advanced Services Properties** window.
3. Select **Start synchronizing x seconds after connection initiation.**
4. In the **Seconds** field, enter the number of seconds or select the number of seconds from the list, for which you want synchronization to be delayed after connection initiation.

High Availability and Load Sharing in ClusterXL

In This Section:

Introduction to High Availability and Load Sharing

ClusterXL is a software-based Load Sharing and High Availability solution that distributes network traffic between clusters of redundant Security Gateways.

ClusterXL has these High Availability features:

- Transparent failover in case of member failures
- Zero downtime for mission-critical environments (when using State Synchronization)
- Enhanced throughput (in Load Sharing modes)
- Transparent upgrades

All members in the cluster are aware of the connections passing through each of the other members. The cluster members synchronize their connection and status information across a secure synchronization network.

The glue that binds the members in a ClusterXL cluster is the Cluster Control Protocol (CCP), which is used to pass synchronization and other information between the cluster members.

High Availability

In a High Availability cluster, only one member is active (Active/Standby operation). In the event that the active cluster member becomes unavailable, all connections are re-directed to a designated standby without interruption. In a synchronized cluster, the standby cluster members are updated with the state of the connections of the active cluster member.

In a High Availability cluster, each member is assigned a priority. The highest priority member serves as the Security Gateway in normal circumstances. If this member fails, control is passed to the next highest priority member. If that member fails, control is passed to the next member, and so on.

Upon Security Gateway recovery, you can maintain the current active Security Gateway (Active Up), or to change to the highest priority Security Gateway (Primary Up).

ClusterXL High Availability supports IPv4 and IPv6.

Load Sharing

ClusterXL Load Sharing distributes traffic within a cluster so that the total throughput of multiple members is increased. In Load Sharing configurations, all functioning members in the cluster are active, and handle network traffic (Active/Active operation).

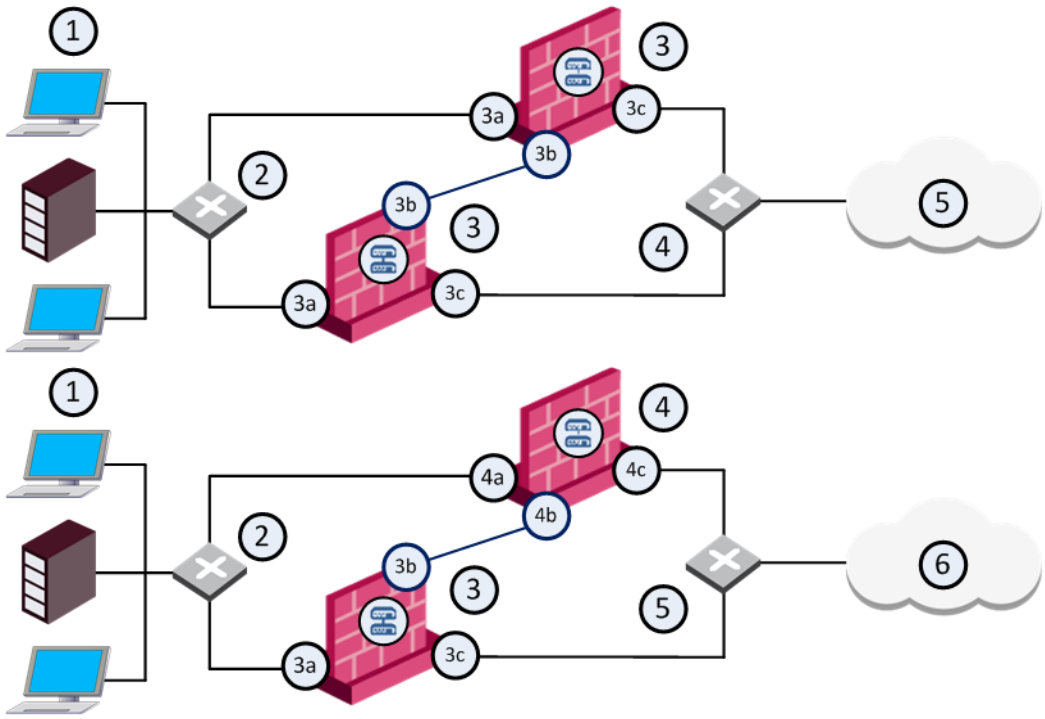
If any member in a cluster becomes unreachable, transparent failover occurs to the remaining operational members in the cluster, thus providing High Availability. All connections are shared between the remaining Security Gateways without interruption.

IPv6 is not supported for Load Sharing clusters.

Example ClusterXL Topology

ClusterXL uses unique physical IP and MAC addresses for each **cluster member**, and a virtual IP addresses for the **cluster** itself. Cluster interface addresses do not belong to any real member interface.

The following diagram illustrates a two-member ClusterXL cluster, showing the cluster virtual IP addresses and member physical IP addresses. This sample deployment is used in many of the examples presented in this chapter.



Item	Description
1	Internal network
2	Internal switch (internal cluster IP address 10.10.0.100)
3	Security Gateway - Cluster member A
3a	Virtual interface to the internal network (10.10.0.1)
3b	Interface to the Cluster Sync network (10.0.10.1)

Item	Description
3c	Virtual interface to the external network (192.168.10.1)
4	Security Gateway - Cluster member B
4a	Virtual interface to the internal network (10.10.0.2)
4b	Interface to the Cluster Sync network (10.0.10.2)
4c	Virtual interface to the external network (192.168.10.2)
5	External switch (external routable cluster IP address 192.168.10.100)
6	Internet

Each cluster member has three interfaces: one external interface, one internal interface, and one for synchronization. Cluster member interfaces facing in each direction are connected via a switch, router, or VLAN switch.

All cluster member interfaces facing the same direction must be in the same network. For example, there must not be a router between cluster members.

The Security Management Server can be located anywhere, and should be routable to either the internal or external cluster addresses.

These sections present ClusterXL configuration concepts shown in the example.

Note

1. High Availability Legacy Mode uses a different topology (see "High Availability Legacy Mode" on page 134).

2. In these examples, RFC 1918 private addresses in the range `192.168.0.0` to `192.168.255.255` are routable (public) IP addresses.

Defining the Cluster Member IP Addresses

The guidelines for configuring each cluster member are as follows:

All members within the cluster must have at least three interfaces:

- An interface facing the external cluster interface, which in turn faces the internet.
- An interface facing the internal cluster interface, which in turn faces the internal network.
- An interface to use for synchronization.

All interfaces pointing in a certain direction must be on the same network.

For example, in the previous illustration, there are two cluster members, Member_A and Member_B. Each has an interface with an IP address facing the Internet through a hub or a switch. This is the external interface with IP address 192.168.10.1 on Member_A and 192.168.10.2 on Member_B, and is the interface that the cluster external interface sees.

Note - This release presents an option to use only two interfaces per member, one external and one internal and to run synchronization over the internal interface. This configuration is not recommended and should be used for backup only (see "Synchronizing Connections in the Cluster" on page 16).

Defining the Cluster Virtual IP Addresses

In the previous illustration, the IP address of the cluster is **192.168.10.100**.

The cluster has one external virtual IP address and one internal virtual IP address. The external IP address is **192.168.10.100**, and the internal IP address is **10.10.0.100**.

The Synchronization Network

State Synchronization between cluster members ensures that if there is a failover, connections that were handled by the failed member will be maintained. The synchronization network is used to pass connection synchronization and other state information between cluster members. This network therefore carries all the most sensitive security policy information in the organization, and so it is important to make sure the network is secure. It is possible to define more than one synchronization network for backup purposes.

To secure the synchronization interfaces, they should be directly connected by a cross cable, or in a cluster with three or more members, use a dedicated hub or switch.

Members in a Load Sharing cluster must be synchronized because synchronization is used in normal traffic flow. Members in a High Availability cluster do not have to be synchronized, though if they are not, connections may be lost upon failover.

The previous illustration shows a synchronization interface with a unique IP address on each member. **10.0.10.1** on Member_A and **10.0.10.2** on Member_B.

Configuring Cluster Addresses on Different Subnets

Only one routable IP address is required in a ClusterXL cluster, for the virtual cluster interface that faces the Internet. All cluster member physical IP addresses can be non-routable.

Configuring different subnets for the cluster IP addresses and the member addresses is useful in order to:

- Configure a cluster to replace one Security Gateway in a pre-configured network, without the need to allocate new addresses to the cluster members.
- Allow organizations to use only one routable address for the ClusterXL Cluster. This saves routable addresses.

ClusterXL Modes

ClusterXL has four working modes. This section briefly describes each mode and its relative advantages and disadvantages.

- **Load Sharing Multicast Mode**
- **Load Sharing Unicast Mode**
- **New High Availability Mode**
- **High Availability Legacy Mode**

Refer to the High Availability Legacy appendix (see "High Availability Legacy Mode" on page 134) for a detailed discussion of legacy High Availability functionality. It is recommended that you use the High Availability New Mode to avoid problems with backward compatibility.

Note - Many examples in the section refer to the sample deployment shown in the ClusterXL example ("Example ClusterXL Topology" on page 28).

Load Sharing Multicast Mode

Load Sharing enables you to distribute network traffic between cluster members. In contrast to High Availability, where only a single member is active at any given time, all cluster members in a Load Sharing solution are active, and the cluster is responsible for assigning a portion of the traffic to each member. This assignment is the task of a decision function, which examines each packet going through the cluster, and determines which member should handle it. Thus, a Load Sharing cluster utilizes all cluster members, which usually leads to an increase in its total throughput.

It is important to understand that ClusterXL Load Sharing, when combined with State Synchronization, provides a full High Availability solution as well. When all cluster members are active, traffic is evenly distributed between the members. In case of a failover event, caused by a problem in one of the members, the processing of all connections handled by the faulty member is immediately taken over by the other members.

ClusterXL offers two separate Load Sharing solutions: Multicast and Unicast. The two modes differ in the way members receive the packets sent to the cluster. This section describes the Multicast mode.

The Multicast mechanism, which is provided by the Ethernet network layer, allows several interfaces to be associated with a single physical (MAC) address. Unlike Broadcast, which binds all interfaces in the same subnet to a single address, Multicast enables grouping within networks. This means that it is possible to select the interfaces within a single subnet that will receive packets sent to a given MAC address.

ClusterXL uses the Multicast mechanism to associate the virtual cluster IP addresses with all cluster members. By binding these IP addresses to a Multicast MAC address, it ensures that all packets sent to the cluster, acting as a Security Gateway, will reach all members in the cluster. Each member then decides whether it should process the packets or not. This decision is the core of the Load Sharing mechanism: it has to assure that at least one member will process each packet (so that traffic is not blocked), and that no two members will handle the same packets (so that traffic is not duplicated).

An additional requirement of the decision function is to route each connection through a Security Gateway, to ensure that packets that belong to a single connection will be processed by the same member. Unfortunately, this requirement cannot always be enforced, and in some cases, packets of the same connection will be handled by different members. ClusterXL handles these situations using its State Synchronization mechanism, which mirrors connections on all cluster members.

Example

This scenario describes a user logging from the Internet to a Web server behind the firewall cluster that is configured in Load Sharing Multicast mode.

1. The user requests a connection from **192.168.10.78** (his member) to **10.10.0.34** (the Web server).
2. A router on the **192.168.10.x** network recognizes **192.168.10.100** (the cluster virtual IP address) as the Security Gateway to the **10.10.0.x** network.
3. The router issues an ARP request to **192.168.10.100**.

4. One of the active members intercepts the ARP request, and responds with the Multicast MAC assigned to the cluster IP address of **192.168.10.100**.
5. When the Web server responds to the user requests, it recognizes **10.10.0.100** as its Security Gateway to the Internet.
6. The Web server issues an ARP request to **10.10.0.100**.
7. One of the active members intercepts the ARP request, and responds with the Multicast MAC address assigned to the cluster IP address of **10.10.0.100**.
8. All packets sent between the user and the Web server reach every cluster member, which decides whether to handle or drop each packet.
9. When a failover occurs, one of the cluster members goes down. However, traffic still reaches all of the active cluster members, and hence there is no need to make changes in the network ARP routing. All that changes is the cluster decision function, which takes into account the new state of the members.

Load Sharing Unicast Mode

Load Sharing Unicast mode provides a Load Sharing solution adapted to environments where Multicast Ethernet cannot operate. In this mode a single cluster member, referred to as *Pivot*, is associated with the cluster virtual IP addresses, and is thus the only member to receive packets sent to the cluster. The pivot is then responsible for propagating the packets to other cluster members, creating a Load Sharing mechanism. Distribution is performed by applying a decision function on each packet, the same way it is done in Load Sharing Multicast mode. The difference is that only one member performs this selection: any non-pivot member that receives a forwarded packet will handle it, without applying the decision function. Note that non-pivot members are still considered as "active", since they perform routing and firewall tasks on a share of the traffic (although they do not perform decisions.).

Even though the pivot member is responsible for the decision process, it still acts as a Security Gateway that processes packets (for example, the decision it makes can be to handle a packet on the local computer). However, since its additional tasks can be time consuming, it is usually assigned a smaller share of the total load.

When a failover event occurs in a non-pivot member, its handled connections are redistributed between active cluster members, providing the same High Availability capabilities of New High Availability and Load Sharing Multicast. When the pivot member encounters a problem, a regular failover event occurs, and, in addition, another member assumes the role of the new pivot. The pivot member is always the active member with the highest priority. This means that when a former pivot recuperates, it will retain its previous role.

Example

In this scenario, we use a Load Sharing Unicast cluster as the Security Gateway between the end user computer and the Web server.

1. The user requests a connection from **192.168.10.78** (his computer) to **10.10.0.34** (the Web server).
2. A router on the **192.168.10.x** network recognizes **192.168.10.100** (the cluster virtual IP address) as the Security Gateway to the **10.10.0.x** network.
3. The router issues an ARP request to **192.168.10.100**.
4. The pivot member intercepts the ARP request, and responds with the MAC address that corresponds to its own unique IP address of **192.168.10.1**.

5. When the Web server responds to the user requests, it recognizes **10.10.0.100** as its Security Gateway to the Internet.
6. The Web server issues an ARP request to **10.10.0.100**.
7. The pivot member intercepts the ARP request, and responds with the MAC address that corresponds to its own unique IP address of **10.10.0.1**.
8. The user request packet reaches the pivot member on interface **192.168.10.1**.
9. The pivot decides that the second member should handle this packet, and forwards it to **192.168.10.2**.
10. The second member recognizes the packet as a forwarded one, and processes it.
11. Further packets are processed by either the pivot member, or forwarded and processed by the non-pivot member.
12. When a failover occurs on the pivot, the second member assumes the role of pivot.
13. The new pivot member sends gratuitous ARP requests to both the **192.168.10.x** and the **10.10.0.x** networks. These requests associate the virtual IP address of **192.168.10.100** with the MAC address that corresponds to the unique IP address of **192.168.10.2**, and the virtual IP address of **10.10.0.100** with the MAC address that correspond to the unique IP address of **10.10.0.2**.
14. Traffic sent to the cluster is now received by the new pivot, and processed by the local computer (as it is currently the only active computer in the cluster).
15. When the first computer recovers, it re-assumes the role of pivot, by associating the cluster IP addresses with its own unique MAC addresses.

High Availability Mode

The High Availability Mode provides basic High Availability capabilities in a cluster environment. This means that the cluster can provide firewall services even when it encounters a problem, which on a stand-alone Security Gateway would have resulted in a complete loss of connectivity. When combined with Check Point State Synchronization, ClusterXL High Availability can maintain connections through failover events, in a user-transparent manner, allowing a flawless connectivity experience. Thus, High Availability provides a backup mechanism, which organizations can use to reduce the risk of unexpected downtime, especially in a mission-critical environment (such as one involving money transactions over the Internet.)

To achieve this purpose, ClusterXL High Availability mode designates one of the cluster members as the active member, while the other members remain in stand-by mode. The cluster virtual IP addresses are associated with the physical network interfaces of the active member (by matching the virtual IP address with the unique MAC address of the appropriate interface). Thus, all traffic directed at the cluster is actually routed (and filtered) by the active member. The role of each cluster member is chosen according to its priority, with the active member being the one with the highest ranking. Member priorities correspond to the order in which they appear in the **Cluster Members** page of the **Cluster Properties** window. The top-most member has the highest priority. You can modify this ranking at any time.

In addition to its role as a firewall, the active member is also responsible for informing the stand-by members of any changes to its connection and state tables, keeping these members up-to-date with the current traffic passing through the cluster.

Whenever the cluster detects a problem in the active member that is severe enough to cause a failover event, it passes the role of the active member to one of the standby members (the member with the currently highest priority). If State Synchronization is applied, any open connections are recognized by the new active member, and are handled according to their last

known state. Upon the recovery of a member with a higher priority, the role of the active member may or may not be switched back to that member, depending on the user configuration.

It is important to note that the cluster may encounter problems in standby members as well. In this case, these members are not considered for the role of active members, in the event of a failover.

Example

This scenario describes a user logging from the Internet to a Web server behind the firewall cluster.

1. The user requests a connection from **192.168.10.78** (his computer) to **10.10.0.34** (the Web server).
2. A router on the **192.168.10.x** network recognizes `192.168.10.100` (the cluster virtual IP address) as the Security Gateway to the **10.10.0.x** network.
3. The router issues an ARP request to **192.168.10.100**.
4. The active member intercepts the ARP request, and responds with the MAC address that corresponds to its own unique IP address of **192.168.10.1**.
5. When the Web server responds to the user requests, it recognizes **10.10.0.100** as its Security Gateway to the Internet.
6. The Web server issues an ARP request to **10.10.0.100**.
7. The active member intercepts the ARP request, and responds with the MAC address that corresponds to its own unique IP address of **10.10.0.1**.
8. All traffic between the user and the Web server is now routed through the active member.
9. When a failover occurs, the standby member concludes that it should now replace the faulty active member.
10. The stand-by member sends gratuitous ARP requests to both the **192.168.10.x** and the **10.10.0.x** networks. These requests associate the virtual IP address of **192.168.10.100** with the MAC address that corresponds to the unique IP address of **192.168.10.2**, and the virtual IP address of **10.10.0.100** with the MAC address that correspond to the unique IP address of **10.10.0.2**.
11. The stand-by member has now switched to the role of the active member, and all traffic directed through the cluster is routed through this computer
12. The former active member is now considered to be "down", waiting to recover from whatever problem that had caused the failover event

Mode Comparison Table

This table summarizes the similarities and differences between the ClusterXL modes.

	Legacy High Availability	**New High Availability**	**Load Sharing Multicast**	**Load Sharing Unicast**
High Availability	Yes	Yes	Yes	Yes
Load Sharing	No	No	Yes	Yes
Performance	Good	Good	Excellent	Very Good

	Legacy High Availability	New High Availability	Load Sharing Multicast	Load Sharing Unicast
Hardware Support	All	All	Not all routers are supported	All
SecureXL Support	Yes	Yes	Yes, with Performance Pack or SecureXL Turbocard.	Yes
State Synchronization Mandatory	No	No	Yes	Yes
VLAN Tagging Support	Yes	Yes	Yes	Yes

Note - For further details regarding VLAN Tagging Support, see *sk40107* http://supportcontent.checkpoint.com/documentation_download?ID=TBD.

Failover

Failover is a redundancy operation that automatically occurs of a member is not functional. When this happens, another member takes over for the failed member.

In a High Availability configuration, if one member in a synchronized cluster goes down, another member becomes active and "takes over" the connections of the failed member. If you do not use State Synchronization, existing connections are closed when failover occurs, although new connections can be opened.

In a Load Sharing configuration, if one member in a cluster is unavailable, its connections are distributed among the remaining members. All members in a Load Sharing configuration are synchronized, so no connections are interrupted.

To tell each member that the other members are alive and functioning, the ClusterXL Cluster Control Protocol maintains a heartbeat between cluster members. If after a predefined time, no message is received from a member, it is assumed that the cluster member is down and failover occurs. At this point, another member automatically assumes the functionality of the failed member.

It should be noted that a cluster member may still be operational, but if any of the above tests fail, then the faulty member starts the failover because it has determined that it can no longer function as a member.

Note that more than one cluster member may encounter a problem that will result in a failover event. In cases where all cluster members encounter such problems, ClusterXL will try to choose a single member to continue operating. The state of the chosen member will be reported *as Active Attention*. This situation lasts until another member fully recovers. For example, if a cross cable connecting the cluster members malfunctions, both members will detect an interface problem. One of them will change to the *Down* state, and the other to *Active Attention*.

When Does a Failover Occur?

A failover takes place when one of the following occurs on the active cluster member:

- Any critical device (such as **fwd**) fails. A critical device is a process running on a cluster member that enables the member to notify other cluster members that it can no longer function as a member. The device reports to the ClusterXL mechanism regarding its current state or it may fail to report, in which case ClusterXL decides that a failover has occurred and another cluster member takes over.
- An interface or cable fails.
- The member fails or becomes unstable.
- The Security Policy is uninstalled. When the Security Policy is uninstalled the Security Gateway can no longer function as a firewall. If it cannot function as a firewall, it can no longer function as a cluster member and a failover occurs. Normally a policy is not uninstalled by itself but would be initiated by a user. For more on failovers, see sk62570 http://supportcontent.checkpoint.com/solutions?id=sk62570.

What Happens When a Security Gateway Recovers?

In a Load Sharing configuration, when the failed Security Gateway in a cluster recovers, all connections are redistributed among all active members.

In a High Availability configuration, when the failed Security Gateway in a cluster recovers, the recovery method depends on the configured cluster setting. The options are:

- **Maintain Current Active Security Gateway** means that if one member passes on control to a lower priority member, control will be returned to the higher priority member only if the lower priority member fails. This mode is recommended if all members are equally capable of processing traffic, in order to minimize the number of failover events.
- **Switch to Higher Priority Security Gateway** means that if the lower priority member has control and the higher priority member is restored, then control will be returned to the higher priority member. This mode is recommended if one member is better equipped for handling connections, so it will be the default Security Gateway.

How a Recovered Cluster Member Obtains the Security Policy

The Administrator installs the security policy on the cluster rather than separately on individual cluster members. The policy is automatically installed on all cluster members. The policy is sent to the IP address defined in the **General Properties** page of the cluster member object.

When a failed cluster member recovers, it will first try to take a policy from one of the other cluster members. The assumption is that the other cluster members have a more up to date policy. If this does not succeed, it compares its own local policy to the policy on the Security Management Server. If the policy on the Security Management Server is more up to date than the one on the cluster member, the policy on the Security Management Server will be retrieved. If the cluster member does not have a local policy, it retrieves one from the Security Management Server. This ensures that all cluster members use the same policy at any given moment.

Implementation Planning Considerations

High Availability or Load Sharing

Whether to choose a Load Sharing (Active/Active) or a High Availability (Active/Standby) configuration depends on the need and requirements of the organization. A High Availability cluster ensures fail-safe connectivity for the organization. Load Sharing provides the additional benefit of increasing performance.

Note - When working on a sync network, it is recommended to use a NIC with the same bandwidth as the NICs that are used for general traffic.

Choosing the Load Sharing Mode

Load Sharing Multicast mode is an efficient way to handle a high load because the load is distributed optimally between all cluster members. However, not all routers can be used for Load Sharing Multicast mode. Load Sharing Multicast mode associates a multicast MAC with each unicast cluster IP address. This ensures that traffic destined for the cluster is received by all members. The ARP replies sent by a cluster member will therefore indicate that the cluster IP address is reachable via a multicast MAC address.

Some routing devices will not accept such ARP replies. For some routers, adding a static ARP entry for the cluster IP address on the routing device will solve the issue. Other routers will not accept this type of static ARP entry.

Another consideration is whether your deployment includes routing devices with interfaces operating in promiscuous mode. If on the same network segment there exists two such routers and a ClusterXL Security Gateway in Load Sharing Multicast mode, traffic destined for the cluster that is generated by one of the routers could also be processed by the other router.

For these cases, use Load Sharing Unicast mode, which does not require the use of multicast for the cluster addresses.

IP Address Migration

If you wish to provide High Availability or Load Sharing to an existing Security Gateway configuration, it is recommended to take the existing IP addresses from the active Security Gateway, and make these the cluster addresses (cluster virtual addresses), when feasible. Doing so will avoid altering current IPSec endpoint identities, as well keep Hide NAT configurations the same in many cases.

Hardware Requirements, Compatibility and Cisco Example

ClusterXL Hardware Requirements

The Cluster is usually located in an environment having other networking devices such as switches and routers. These devices and the Security Gateways must interact to assure network connectivity. This section outlines the requirements imposed by ClusterXL on surrounding networking equipment.

HA New and Load Sharing Unicast Modes

Multicast mode is the default Cluster Control Protocol (CCP) mode in High Availability New Mode and Load Sharing Unicast Mode (and also Load Sharing Multicast Mode).

When using CCP in multicast mode, configure the following settings on the switch.

Switch Setting	Explanation
IGMP and Static CAMs	IGMP registration (also known as IGMP Snooping) is enabled by default. You can disable IGMP registration ("Disabling IGMP Snooping" on page 41). In scenarios where disabling IGMP registration is problematic, you can configure static CAMs to allow multicast traffic on specified ports.
Disabling multicast limits	Certain switches have an upper limit on the number of broadcasts and multicasts that they can pass, in order to prevent broadcast storms. This limit is usually a percentage of the total interface bandwidth. It is possible to either turn off broadcast storm control, or to allow a higher level of broadcasts or multicasts through the switch. If the connecting switch is incapable of having any of these settings configured, it is possible, though less efficient, for the switch to use broadcast to forward traffic, and to configure the cluster members to use broadcast CCP (see "Choosing the CCP Transport Mode on the Cluster Members" on page 46).

Configure the following settings on the router:

Router Setting	Explanation
Unicast MAC	When working in High Availability Legacy mode, High Availability New mode and Load Sharing Unicast mode, the Cluster IP address is mapped to a regular MAC address, which is the MAC address of the active member. The router needs to be able to learn this MAC through regular ARP messages.

VMAC Mode

When ClusterXL is configured in HA mode or Load Sharing unicast mode (not multicast) a single cluster member is associated with the Cluster Virtual IP address. In a High Availability environment, the single member is the active member. In a Load Sharing environment, the single member is the pivot.

After fail-over, the new active member (or pivot member) broadcasts a series of Gratuitous ARP Requests (G-ARPs). The G-ARPS associate the Virtual IP address of the cluster with the physical MAC address of the new active member or the new pivot. When this happens:

- **A member with a large number of Static NAT entries can transmit too many GARPS**
  Switches may not integrate these GARP updates quickly enough into their ARP tables. Switches continue to send traffic to the physical MAC address of the member that failed. This results in traffic outage until the switches have fully updated ARP cache tables.
- **Network components, such as VoIP phones, ignore GARPs**
  These components continue to send traffic to the MAC address of the failed member.

To minimize possible traffic outage during a fail-over, configure the cluster to use a virtual MAC address (VMAC).

By enabling Virtual MAC in ClusterXL High Availability mode, or Load Sharing Unicast mode, all cluster members associate the same Virtual MAC address with all Cluster Virtual Interfaces and the Virtual IP address. In Virtual MAC mode, the VMAC that is advertised by the cluster members (through G-ARP Requests) keeps the real MAC address of each member and adds a Virtual MAC address on top of it.

(For local connections and sync connections, the real MAC address of each member is still associated with its real IP address.)

Note - VMAC mode is supported only on SecurePlatform and Gaia.

- In SecurePlatform, you can enable VMAC with the command line only
- In Gaia, you can enable VMAC with the command line or SmartConsole

VMAC failover time is shorter than a failover that involves a physical MAC address.

To configure VMAC Mode using SmartConsole:

1. Double-click the Cluster object to open its **Properties** window.
2. On the **ClusterXL and VRRP** page, select **Use Virtual MAC**.
3. Install a Policy.

To configure VMAC Mode using the command line:

Set the value of global kernel parameter `fwha_vmac_global_param_enabled`.

1. First get the current value of global kernel parameter by running this command on a cluster member:
   ```
   fw ctl get int fwha_vmac_global_param_enabled
   ```
2. Set the new value by running:
   ```
   fw ctl set int fwha_vmac_global_param_enabled VALUE
   ```
   Where:

VALUE	Description
1	VMAC enabled
0	VMAC disabled

3. Make sure VMAC mode is enabled by running: `cphaprob -a if`
   This command shows the VMAC address of each virtual cluster interface.

Note -

- On SecurePlatform run this command in the Expert mode.
- On Gaia run this command can be run in Clish or the Expert mode.

For more on VMAC mode, see: sk50840
http://supportcontent.checkpoint.com/solutions?id=sk50840

To set the VMAC mode value *permanently*, see sk26202
http://supportcontent.checkpoint.com/solutions?id=sk26202

Load Sharing Multicast Mode

When working in Load Sharing Multicast mode, the switch settings are as follows:

Switch Configuration for Load Sharing Multicast Mode

Switch Setting	Explanation
CCP in Multicast mode	Multicast mode is the default Cluster Control Protocol mode in Load Sharing Multicast.
Port Mirroring	ClusterXL does not support the use of unicast MAC addresses with Port Mirroring for Multicast Load Sharing solutions.

When working in Load Sharing Multicast mode, the router must support sending unicast IP packets with Multicast MAC addresses. This is required so that all cluster members will receive the data packets.

The following settings may need to be configured in order to support this mode, depending on the model of the router:

Router Configuration for Load Sharing Multicast Mode

Router Setting	Explanation
Static MAC	Most routers can learn ARP entries with a unicast IP and a multicast MAC automatically using the ARP mechanism. If you have a router that is not able to learn this type of mapping dynamically, you'll have to configure static MAC entries.
IGMP and static cams	Some routers require disabling of IGMP snooping or configuration of static cams in order to support sending unicast IP packets with Multicast MAC addresses.
Disabling multicast limits	Certain routers have an upper limit on the number of broadcasts and multicasts that they can pass, in order to prevent broadcast storms. This limit is usually a percentage of the total interface bandwidth. It is possible to either turn off broadcast storm control, or to allow a higher level of broadcasts or multicasts through the router.
Disabling forwarding multicast traffic to the router	Some routers will send multicast traffic to the router itself. This may cause a packet storm through the network and should be disabled.

ClusterXL Hardware Compatibility

The following routers and switches are known to be compatible for all ClusterXL modes:

Routers

- Cisco 7200 Series
- Cisco 1600, 2600, 3600 Series

Routing Switch

- Extreme Networks Blackdiamond (Disable IGMP snooping)
- Extreme Networks Alpine 3800 Series (Disable IGMP snooping)
- Foundry Network Bigiron 4000 Series
- Nortel Networks Passport 8600 Series
- Cisco Catalyst 6500 Series (Disable IGMP snooping, Configure Multicast MAC manually)

Switches

- Cisco Catalyst 2900, 3500 Series
- Nortel BayStack 450
- Alteon 180e
- Dell PowerConnect 3248 and PowerConnect 5224

Example Configuration of a Cisco Catalyst Routing Switch

The following example shows how to perform the configuration commands needed to support ClusterXL on a Cisco Catalyst 6500 Series routing switch. For more details, or instructions for other networking devices, please refer to the device vendor documentation.

Disabling IGMP Snooping

To disable IGMP snooping run:

```
no ip igmp snooping
```

Defining Static Cam Entries

To add a permanent multicast entry to the table for module 1, port 1, and module 2, ports 1, 3, and 8 through 12:

1. Console> (enable) set cam permanent 01-40-5e-28-0a-64 1/1,2/1,2/3,2/8-12
2. Permanent multicast entry added to CAM table.
3. Console> (enable)

To determine the MAC addresses that must be set:

1. On a network that has a cluster IP address of x.y.z.w :
   - If y<=127, the multicast MAC address would be **01:00:5e:y:z:w**. For example: **01:00:5e:5A:0A:64** for 192.90.10.100
   - If y>127, the multicast MAC address would be **01:00:5e:(y-128):z:**w. For example: **01:00:5e:28:0A:64** for 192.168.10.100 (168-128=40 = 28 in hex).
2. For a network x.y.z.0 that does not have a cluster IP address, such as the sync, you would use the same procedure, and substitute **fa** instead of **0** for the last octet of the MAC address.
   - For example: **01:00:5e:00:00:fa** for the 10.0.0.X network.

Disabling Multicast Limits

To disable multicast limits run:

```
no storm-control multicast level
```

ClusterXL Compatibility (Excluding IPS)

The following table and accompanying notes present ClusterXL Load Sharing and High Availability compatibility for OPSEC Certified cluster products (see "Working with OPSEC Certified Clustering Products" on page 56). Some Check Point products and features are not supported or are only partially supported (as detailed in the footnotes) for use with ClusterXL.

Feature or Product	Feature	LS	HA
Security Management		No	No
firewall	Authentication/Security Servers	Yes (**1**)	Yes (**1**)
firewall	ACE servers and SecurID	Yes	Yes
firewall	Application Intelligence protocol inspection (**2**)	Yes (**3**)	Yes
firewall	Sequence Verifier	Yes (**4**)	Yes (**1**)
firewall	UDP encapsulation	Yes	Yes
firewall	SAM	Yes	Yes
firewall	ISP Redundancy	Yes	Yes
VPN	Third party VPN peers	Yes	Yes
Endpoint Security Client	Software Distribution Server (SDS)	No	No
Endpoint Security Client	IP per user in Office Mode	Yes	Yes
SecureXL or Performance Pack		Yes	Yes
Check Point QoS		Yes (**4, 5**)	Yes
SmartProvisioning	SmartLSM Security Gateway	No	No
Check Point Security Gateway		Yes	Yes

Configuring a Static ARP Entry on the Router

To define a static ARP entry:

1. Determine the MAC address (see "Defining Static Cam Entries" on page 41).
2. Run `arp <MAC address> arpa`

Disabling Multicast Packets from Reaching the Router

To prevent multicast packets from reaching the router:

1. Determine the MAC address (see "Defining Static Cam Entries" on page 41).
2. Run `set cam static <MAC address> module/port`.

Check Point Software Compatibility

ClusterXL Compatibility with IPS

The following IPS features are supported by ClusterXL, with the limitations listed in the notes.

Feature	Load Sharing	High Availability
Fragment Sanity Check	Yes (1, 3)	Yes (1)
Pattern Matching	Yes (2, 3)	Yes (2)
Sequence Verifier	Yes (2, 4)	Yes (2)
FTP, HTTP and SMTP Security Servers	Yes (2, 5)	Yes (2)

Footnotes

1. If there is a failover when fragments are being received, the packet will be lost.
2. Does not survive failover.
3. Requires unidirectional stickiness. This means that the same member must receive all external packets, and the same member must receive all internal packets, but the same member does not have to receive both internal and external packets.
4. Requires bidirectional connection stickiness.
5. Uses the forwarding layer, described in the next section.

Forwarding Layer

The Forwarding Layer is a ClusterXL mechanism that allows a cluster member to pass packets to other members, after they have been locally inspected by the firewall. This feature allows connections to be opened from a cluster member to an external host.

Packets originated by cluster members are hidden behind the cluster virtual IP. Thus, a reply from an external host is sent to the cluster, and not directly to the source member. This can pose problems in the following situations:

- The cluster is working in New High Availability mode, and the connection is opened from the stand-by member. All packets from the external host are handled by the active member, instead.
- The cluster is working in a Load Sharing mode, and the decision function has selected another member to handle this connection. This can happen since packets directed at a cluster IP are distributed among cluster members as with any other connection.

If a member decides, upon the completion of the firewall inspection process, that a packet is intended for another cluster member, it can use the Forwarding Layer to hand the packet over to

that destination. This is done by sending the packet over a secured network (any subnet designated as a Synchronization network) directly to that member. It is important to use secured networks only, as encrypted packets are decrypted during the inspection process, and are forwarded as clear-text (unencrypted) data.

Packets sent on the Forwarding Layer use a special source MAC address to inform the receiving member that they have already been inspected by another Security Gateway. Thus, the receiving member can safely hand over these packets to the local Operating System, without further inspection. This process is secure, as Synchronization Networks should always be isolated from any other network (using a dedicated network).

Configuring ClusterXL

In This Section:

This procedure describes how to configure the Load Sharing Multicast, Load Sharing Unicast, and High Availability New Modes from scratch. Their configuration is identical, apart from the mode selection in SmartConsole Cluster object or Cluster creation wizard.

If you are still using the High Availability Legacy Mode, refer to the appendix (see "High Availability Legacy Mode" on page 134).

Creating Cluster Members

Important - The hardware for all cluster members must be exactly the same, including:

- CPU
- Motherboard
- Memory
- Number and type of interfaces

To create new cluster members for ClusterXL:

1. Install and configure Check Point Security Gateway for all cluster members. Each member must use the identical version and build. For installation and initial configuration procedures, refer to the *R80.10 Installation and Upgrade Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD.

   During the installation process, enable ClusterXL and State Synchronization:

   - For Gaia members, run `cpconfig` from the command line and select **Enable cluster membership for this gateway**.
   - For SecurePlatform and Solaris members, select **Enable cluster membership for this gateway**.
   - For Windows members, select **This Gateway is part of a cluster**.

     If you did not perform this action during installation, you can always do so by using the **cpconfig** utility at a later time. Run the **cpconfig** from the command line, and select the appropriate options to enable cluster capabilities for that member. You may be asked to reboot the member.
2. Define an IP address for each interface on all members. **Do not define IPv6 addresses for synchronization interfaces**.
3. For VPN cluster members, synchronize member clocks accurately to within one second of each other. If these members are constantly up and running it is usually enough to set the

time once. More reliable synchronization can be achieved using NTP or some other time synchronization services supplied by the operating system. Cluster member clock synchronization is not applicable for non VPN cluster functionality.

4. Connect the cluster members to each other and to the networks through switches. For the synchronization interfaces, you can use a cross cable or a dedicated switch. Make sure that each network (internal, external, Synchronization, DMZ, and so on) is configured on a separate VLAN, switch or hub.

Note - You can also perform synchronization over a WAN (see "Synchronizing Clusters on a Wide Area Network" on page 24)

Configuring Routing for Client Computers

To configure routing for client computers:

1. Configure routing so that communication with internal networks uses the external cluster virtual IP address. For example, configure a static route such that internal network 10.10.0.0 is accessible through 192.168.2.100.
2. Configure routing so that communication with external networks uses the internal cluster IP address. For example, define the internal network IP address 10.10.0.100 as the default Security Gateway for each computer on the internal side of the router.

Choosing the CCP Transport Mode on the Cluster Members

The ClusterXL Control Protocol (CCP) uses multicast by default, because it is more efficient than broadcast. If the connecting switch cannot forward multicast traffic, it is possible, though less efficient, for the switch to use broadcast to forward traffic.

To change the CCP mode between broadcast and multicast, run:

```
cphaconf set_ccp broadcast|multicast
```

Configuring the Cluster Object and Members

Overview

You can use one of these procedures to define a cluster object and its members:

- **Simple Mode (Wizard)** ("Using the Wizards" on page 47) - Lets you quickly create a new cluster and configure some basic Cluster properties:
  - Cluster properties and virtual addresses (VIP)
  - Member properties and topology
  - Synchronization interfaces and IP addresses
- **Classic Mode** ("Manual Configuration" on page 48) - Opens the **Cluster Gateway Properties** window, where you manually create a cluster and configure its properties.

The **Cluster Gateway Properties** window lets you:

- Manually create a new cluster

- Add and configure Software Blades for the cluster
- Configure other cluster properties that you cannot configure with the wizards
- Change the properties of an existing cluster to manually create new clusters and configure cluster features

Using the Wizards

This version includes two wizards:

- Check Point appliances and open servers
- Check Point small office appliances

Small Appliance Wizard

The **Small Appliance Wizard** is recommended for these Check Point appliances:

- 1100 appliances
- 600 appliances
- SG80 appliances with the R75.20 firmware upgrade

To create a new cluster with the Small Appliance Wizard:

1. In SmartConsole, right-click **Check Point** in the **Network Objects** tree.
2. Select **Security Cluster > Small Office Appliance**.
3. In the **Check Point Security Gateway Cluster Creation** window, click **Wizard Mode**.
4. In the **Cluster General Properties** window, enter a unique name for the cluster.
5. In the **Cluster Members** window:
   a) Enter the member name and IPv4 addresses for each member.
   b) Enter the one-time password for SIC trust.
6. In the **Configure WAN Interface** page, configure the cluster virtual interface IP address.
7. Define the virtual IP addresses for the other cluster interfaces.
8. Click **Next**, and then **Finish** to complete the wizard.

After you finish the wizard, we recommend that you open the cluster object and manually do these steps:

- Define Anti-Spoofing properties for each interface
- Change the topology type (Internal or External) if necessary
- Define the Network Objective
- Configure other Software Blades, features and properties as necessary.

Check Point Appliance or Open Server Wizard

The Check Point Appliance or Open Server Wizard is recommended for enterprise grade appliances and open server platforms.

To create a new cluster with the Appliance or Open Server Wizard:

1. In SmartConsole, right-click Check Point in the **Network Objects** tree.
2. Select **Security Cluster > Check Point Appliance/Open Server**.

3. In the **Check Point Security Gateway Cluster Creation** window, click Wizard Mode.
4. In the **Cluster General Properties** window, enter or select:
   - Cluster Name - Unique name for the cluster
   - Cluster IPv4 and IPv6 address - Virtual Management IP addresses for this cluster.

     Important: You must define a corresponding IPv4 address for every IPv6 address. This release does not support pure IPv6 addresses.
   - **Choose the Cluster Solution** - Select **Check Point ClusterXL** and then select **High Availability** or **Load Sharing**.
5. In the **Cluster Member Properties** window, click Add > New Cluster Member to configure each member.
   a) Enter the physical IPv4 and IPv6 addresses.

      Note: Make sure that you do not define IPV6 address for sync interfaces. The wizard does not let you define an interface with an IPv6 address as a sync interface.
   b) Enter and confirm the SIC trust activation key.
6. In the **Cluster Topology** window, define a network objective (Role) for each network interface and, if necessary, define the virtual cluster IP addresses.

   The wizard automatically calculates the subnet for each network and assigns it to the applicable interface on each member. The calculated subnet shows in the upper section of the window.

   The available network objectives are:
   - **Cluster Interface** - A cluster interface that connects to an internal or external network. Enter the cluster virtual IP addresses for each network (internal or external). These addresses must be located in the calculated subnet.
   - **Cluster Sync Interface** - A cluster synchronization interface. You must define one or more synchronization interfaces for redundancy. If you are using more than one synchronization interface, define which interface is the primary, secondary, or tertiary interface. Synchronization redundancy is not supported on Small Business appliances. On these appliances, you can only select **1st sync** and only for the LAN2/SYNC interface. You cannot configure VLANs on the synchronization interface.
   - **Monitored Private** - An interface that is not part of the cluster, but ClusterXL monitors the member state and failover occurs if a fault is detected.
   - **Non Monitored Private** - ClusterXL does not monitor the member state and there is no failover.

     This option is recommended for the management interface.
7. Click **Next** and then **Finish** to complete the wizard.

After you finish the wizard, we recommend that you open the cluster object and do these procedures:

- Define Anti-Spoofing properties for each interface
- Change the topology type (Internal or External) if necessary
- Configure other Software Blades, features and properties as necessary.

Manual Configuration

The **Cluster Gateway Properties** window contains many different ClusterXL properties as well as other properties related to Security Gateway and Software Blade functionality. This section

includes only the properties and procedures directly related to ClusterXL. See the applicable *Administration Guides* (http://supportcontent.checkpoint.com/solutions?id=sk92965) for these non-ClusterXL properties.

Configuration Steps

Configuring General Properties

To configure the general properties of a cluster:

1. Enter a unique name for this cluster object in the designated field.
2. Enter the virtual cluster IPv4 and IPv6 addresses.
3. Select the hardware platform, Check Point version and operating system.
4. Select **ClusterXL** and other Network Security Software Blades as necessary.

Defining Cluster Members

To configure a cluster member:

1. Go to the **Cluster Members** page.
2. Click **Add > New Cluster Member**.
3. In the **Cluster Members Properties** window **General** tab, enter a member **Name** and the physical, IPv4 and IPv6 addresses. These addresses must be routable from the Security Management Server and can be an internal, external or dedicated management interface.

   Important: You must define a corresponding IPv4 address for every IPv6 address. This release does not support pure IPv6 addresses.
4. Click **Communication**, and initialize Secure Internal Communication (SIC) trust.
5. Configure **NAT** and **VPN** settings on the appropriate tabs as required.

Removing a Member

To remove a member from the cluster:

1. Click **Remove** in the **Cluster Members** page.
2. Select **Detach Member from Cluster**.

   You can also right-click the cluster member and select **Detach from Cluster**.

Working with Cluster Topology

IPv6 Considerations

To activate IPv6 functionality for an interface, define an IPv6 address for the applicable interface on the cluster and on each member. All interfaces configured with an IPv6 address must also have a corresponding IPv4 address. If an interface does not require IPv6, only the IPv4 definition address is necessary.

Note - You must configure synchronization interfaces with an IPv4 address only. This is because the synchronization mechanism works using IPv4 only. All IPv6 information and states are synchronized using this interface.

To open the In the **Topology** page, click **Edit Topology**. The **Edit Topology** window opens.

	Network Objective	MyCluster	GW82	GW83	Topology
			Get Topology	Get Topology	
Name	Cluster	interface	eth2	eth2	
IPv4 Address		10.20.20.89	10.20.20.82	10.20.20.83	Internal
Net Mask		255.255.255.0	255.255.255.0	255.255.255.0	
IPv6 Address		2001:db8::20:20:89	2001:db8::20:20:82	2001:db8::20:20:83	
Prefix length		96	96	96	
Name	Cluster	interface-0	eth1	eth1	
IPv4 Address		10.10.10.89	10.10.10.82	10.10.10.83	Internal
Net Mask		255.255.255.0	255.255.255.0	255.255.255.0	
IPv6 Address		2001:db8::10:10:89	2001:db8::10:10:82	2001:db8::10:10:83	
Prefix length		96	96	96	
Name	Monitored Private		eth0	eth0	
IPv4 Address			192.168.3.82	192.168.3.83	External
Net Mask			255.255.255.0	255.255.255.0	
IPv6 Address			2001:db8::192:1...	2001:db8::192:1...	
Prefix length			96	96	
Name	1st Sync		eth3	eth3	
IPv4 Address			10.30.30.82	10.30.30.83	Internal
Net Mask			255.255.255.0	255.255.255.0	
IPv6 Address					
Prefix length					

This window is a table that shows topology information for all detected interfaces in the cluster object and its related members. The rows show the interfaces. The columns contain these information categories for each interface:

- **Cluster object** - Virtual IPv4 and IPv6 addresses for each interface connected to an internal or external network. Synchronization and private interfaces do not have virtual interfaces.
- **Members** - Physical IP addresses for each member interface.
- **Network Objective** - The purpose of the network connected to an interface.
- **Topology** - Type of network (Internal or External) for Anti-Spoofing protection.

If you used the Cluster Properties window to manually create a cluster, an empty table shows. You manually add and configure the interfaces in the table.

If you created the cluster using one of the wizards, the topology is calculated automatically and shows in the **Edit Topology** window. You can change the IP addresses and other properties directly in this window.

To configure cluster virtual interface Properties:

1. Right-click an interface in the Cluster column and select **Edit Interface**.
2. In the I**nterface Properties** window **General** tab, configure the name and IP addresses.

   Note: Make sure that you do not define IPV6 address for sync interfaces. The wizard does not let you define an interface with an IPv6 address as a sync interface.

   Important: You must define a corresponding IPv4 address for every IPv6 address. This release does not support pure IPv6 addresses.
3. In the **Topology** tab:

   a) Select **External** or **Internal**.

b) For internal interfaces configure the IP address located behind this interface and elect I**nterface leads to DMZ** if necessary.

4. In the **Member Network** tab, enter a member interface IP address in the same subnet as the cluster virtual IP address.

   Cluster members can be located on a different subnet than the cluster virtual IP address. When this occurs, use this tab to map the member IP address to the cluster virtual IP address. This advanced option is explained in Configuring Cluster Addresses on Different Subnets.

5. In the **Network Objective** column, select an objective for each network from the list. The options are explained in the online help.

   To define a new network, click **Add Network.**

   The available network objectives are:

   - **Cluster** - An interface that connects to an internal or external network.

     Enter the cluster virtual IPv4 and IPv6 addresses for each network (internal or external). These addresses must be located in the calculated subnet.

   - **Cluster + Sync Address** - A cluster interface that also works as a Synchronization Interface. These interfaces must be located in the calculated subnet.
   - **Sync Address** - An interface used exclusively for member synchronization.
   - **Monitored Private** - An interface that is not part of the cluster, but ClusterXL monitors the member state and failover occurs if a fault is detected.
   - **Non-Monitored Private** - An interface that is not part of the cluster. ClusterXL does not monitor the member state and there is no failover.

     This option is recommended for the management interface.

Completing the Definition

1. Configure other blades and options for the cluster object as required (**NAT, VPN**, **Remote Access**, and other advanced options).
2. Install a Security Policy on the cluster.

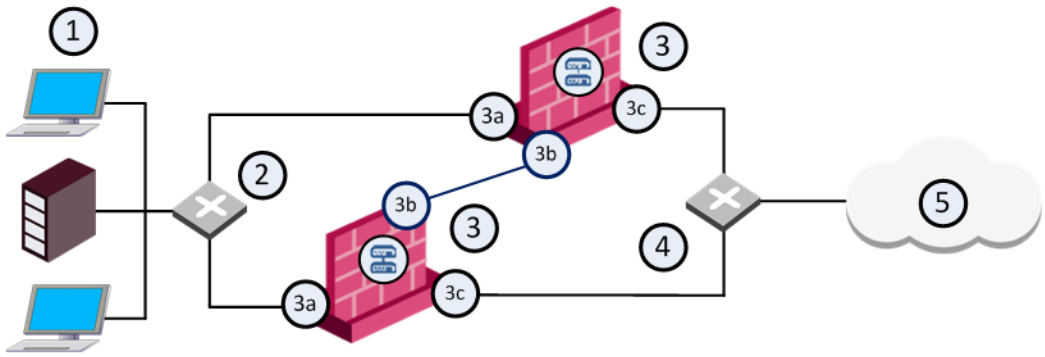
Changing the Synchronization Interface

To change the synchronization interface on your cluster members:

1. In the operating system WebUI or command line, add a new interface on each member.
2. In SmartConsole, open the cluster object.
3. In the **Gateway Cluster Properties** window, go to the **Topology** page and click **Edit**.
4. Click **Get > All Member's Interfaces with Topology**.
5. Select the old interfaces and click **Remove**.
6. Configure the new interfaces as Synchronization interfaces.
7. Install policy.
8. In the operating system WebUI or command line, delete the old interface on each member.

Configuring Gateway Cluster in Bridge Mode

You can configure cluster gateways for bridge mode in different deployments Active/Standby mode or Active/Active mode.



Item	Description
1	Internal network
2	Switch for internal network
3	Security Gateway bridging Layer-2 traffic
3a	eth1 - connects to the internal network
3b	eth3 - ClusterXL Sync interface
3c	eth2 - connects to the external network (192.168.10.1)
4	Switch for external network
5	Internet

Configuring Active/Standby Mode

This is the preferred mode in topologies that support it.

In Active-Standby mode, ClusterXL decides the cluster state. The standby member drops all packets. It does not pass any traffic, including STP/RSTP/MSTP. If there is a failover, the switches are updated by the Security Gateway to forward traffic to the new active member.

If you use this mode, it is best to disable STP/RSTP/MSTP on the adjacent switches.

To configure Active/Standby mode:

1. Configure the cluster ("Configuring Active/Active Mode " on page 53).
2. Run: `cpconfig`
3. Enter `8`, to select **Enable Check Point ClusterXL for Bridge Active/Standby**.
4. Confirm: `y`
5. Reboot the cluster member.
6. Install Policy.
7. Test the cluster state: `cphaprob stat`

The output should be similar to:

```
Cluster Mode: High Availability (Active Up, Bridge Mode) with IGMP Membership
Number     Unique Address     Firewall State (*)
1 (local>  2.2.2.3            Active
2          2.2.2.2            Standby
```

Configuring Active/Active Mode

When you define a bridge interface on a Security Gateway cluster, Active/Active mode is activated by default.

Before you begin, install ClusterXL High Availability on a Gaia appliance or open server.

To configure Active/Active mode, do these steps on each member of the cluster:

1. Configure dedicated management and Sync interfaces.
2. Add a bridge interface, as in a one-gateway deployment.

   Do not configure an IP address on the newly created bridge interface.
3. In SmartConsole, add the cluster object:

   a) Open the **Network Management** page of the of the cluster object.

   b) Get the cluster Interfaces with Topology.

   c) Make sure the dedicated management and Sync interfaces are configured.

   d) Make sure the bridge interface and bridge ports are not in the topology.

   Bridge port topology cannot be defined. It is **external** by default.
4. Install Policy.
5. See the cluster state: `cphaprob stat`

   Example of expected output:

   ```
   Cluster Mode: High Availability (Active Up, Bridge Mode) with IGMP
   Membership
   Number     Unique Address     Firewall State (*)
   1 (local>  192.0.2.3            Active
   2          192.0.2.2            Active
   ```
6. Make sure that cluster is configured for High Availability ("Confirming the High Availability Configuration" on page 53).

Confirming the High Availability Configuration

After you configure Active/Active mode, the output for `chpaprob stat` shows that the `Firewall State` is `Active/Active`. Make sure that the cluster is configured for High Availability.

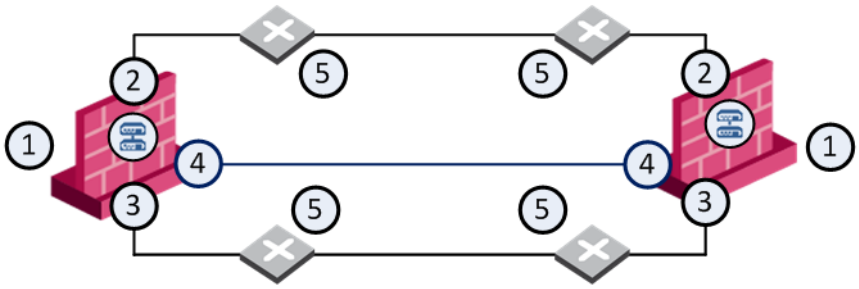
To confirm the High Availability configuration:

1. Open the cluster object.
2. In the cluster **Properties** window, click **ClusterXL**.
3. In the **Cluster Mode** section, make sure that **High Availability** is selected.
4. Click **OK**.

Cluster Between Four Switches

You can configure a bridged cluster between four switches, in Active/Active mode.

Active/Standby mode is not supported.



Item	Description
1	Security Gateway bridging Layer-2 traffic
2	eth1
3	eth2
4	eth3 - ClusterXL Sync interface
5	Switch

See also: Link Aggregation with ClusterXL in Layer-2
http://supportcontent.checkpoint.com/documentation_download?ID=23341

Configuring Link State Propagation

You can bind two ports together, so that when the link state for one port goes down, the other port also goes down. This lets a switch detect and react to a link failure on the other side of a bridge or another part of the network.

This feature is available in one of these modes:

- **Automatic port detection and port pair creation** - All bridge ports are assigned to a port pair (the pair in the bridge).
- **Manual port pair creation** - Up to four port pairs are supported.

Link state propagation is supported on these Check Point appliance line cards:

- CPAC-4-1C/CPAC-8-1C – Copper line cards with IGB driver
- CPAC-4-1F – 1Gbe fiber line card with an IGB driver
- CPAC-4-10F – 10Gbe fiber line card with an IXGBE driver

For example:
`fw_lsp_pair1=”eth1,eth2"`

Note - You can add up to four lines to this file, one for each pair.

Note: The below procedures are applicable to R77.20 and higher.

To configure Link State Propagation for automatic port detection:

1. Open `FWDIR/modules/fwkern.conf` in a text editor.

If there is no `fwkern.conf` file, create a new one.

2. Add this line:
   `fw_link_state_propagation_enabled=1`
3. Reboot the computer.

To create port pairs automatically:

1. Open `FWDIR/modules/fwkern.conf` in a text editor.
   If there is no `fwkern.conf` file, create a new one.
2. Add these lines:
   ```
   fw_link_state_propagation_enabled=1
   fw_manual_link_state_propagation_enabled=1
   fw_lsp_pair<1-4>=”<interface_name1,interface_name2>”
   ```
3. Reboot the computer.

Note - Link State Propagation is a Firewall Software Blade feature. It is supported for Security Gateways and clusters. You must configure Link State Propagation for each cluster member.

Working with OPSEC Certified Clustering Products

In This Section:

Introduction to OPSEC Certified Clustering Products

There are a number of OPSEC certified High Availability and Load Sharing products. These products are used to build highly available Security Gateway clusters and to distribute traffic evenly among the clustered Security Gateways.

Each OPSEC certified clustering application has its particular strengths and capabilities, whether it is monitoring, management, or performance. The role of these clustering applications is to:

1. Decide which cluster member will deal with each connection.
2. Perform health checks. This involves checking the status of a cluster member (for example, Active, Standby, or Down), and checking the status of the member interfaces.
3. Perform failover.

OPSEC certified clustering products use Check Point state synchronization mechanism (see "Synchronizing Connections in the Cluster" on page 16) to exchange and update connection data and other states between cluster members.

This section is general guidelines for working with OPSEC certified clustering products. Configuration details vary for each clustering product. Follow the instructions supplied with the OPSEC product.

Configuring OPSEC Certified Clustering Products

This procedure describes how to configure an OPSEC certified Security Gateway clustering solution.

Preparing the Switches and Configuring Routing

Follow the instructions in your clustering product documentation for:

- Preparing the switches and routers
- Configuring routing

Preparing the Cluster Members

To prepare the cluster members:

Note - You can run synchronization across a WAN (see "Synchronizing Clusters on a Wide Area Network" on page 24).

1. Define IP addresses for all interfaces on all the cluster members.
2. Connect the cluster network members, via the switches. For the Synchronization interfaces, a cross-over cable or a dedicated switch is recommended.
3. For IPSO clusters, configure VRRP or IP Clustering before installing Check Point Security Gateway.
4. For OPSEC certified clusters, follow the vendor recommendations.

   After the installation completes, make sure that **Enable VPN-1/FW-1 monitoring** is set to *Enable* in the IPSO configuration manager. This assures that IPSO will monitor changes in the status of the firewall. For VRRP and IP Clustering in IPSO 3.8.2 and above, the state of the firewall is reported to the IPSO cluster for failover purposes.
5. Install a Check Point Security Gateway on all cluster members. During the configuration phase (or later, using the `cpconfig` utility), enable State Synchronization by selecting **Enable cluster membership for this gateway**.

SmartConsole Configuration for OPSEC Clusters

Using SmartConsole, create the Cluster object. To define a new Cluster object, right click the **Network Objects** tree, and choose **New Check Point > Cluster** Configuration of the Cluster Object can be performed using:

- **Simple Mode (Wizard)** which guides you step by step through the configuration process. See the online help for further assistance.
- **Classic Mode**, described below.

Classic Mode Configuration

To use Classic Mode configuration with OPSEC:

1. In the **General Properties** page of the Cluster object, give the cluster a general IP address. In general, make it the external virtual IP address of the cluster.

   In the list of **Check Point Products**, ensure ClusterXL is *not* selected.
2. In the **Cluster Members** page, click **Add > New Cluster Member** to add cluster members to the cluster. Cluster members exist solely inside the Cluster object. For each cluster member:
   - In the **Cluster Members Properties > General** tab, define a name a **Name** and **IP Address**. Choose an IP address that is routable from the Security Management Server, so that the Security Policy installation will be successful. This can be an internal or an external address, or a dedicated management interface.
   - Click **Communication**, and Initialize Secure Internal Communication (SIC).
   - Define the **NAT** and **VPN** tabs, as required.

   You can also add an existing Security Gateway as a cluster member by selecting **Add > Add Gateway to Cluster** in the **Cluster Members** page and selecting the Security Gateway from the list in the **Add Gateway to Cluster** window.

If you want to remove a Security Gateway from the cluster, click **Remove** in the **Cluster Members** page and select **Detach Member from Cluster** or right-click on the cluster member in the **Network Objects** tree and select **Detach from Cluster**.

3. In the **3rd Party Configuration** page, specify the cluster operating mode, and for the **3rd Party Solution**, select *OPSEC*, and check **Use State Synchronization**.
4. The **Topology** page is used to define the virtual cluster IP addresses and cluster member addresses.

   For each cluster member, define the interfaces for the individual members.

   For OPSEC certified products, the configuration of virtual cluster IPs is mandatory in several products, while in others it is forbidden. Refer to your cluster product documentation for details.

   Define the synchronization networks. Depending on the OPSEC implementation, it might be possible to "get" the synchronization network from the OPSEC configuration if it is already defined. Refer to the OPSEC documentation to find out if this feature is implemented for a specific OPSEC product.
5. Now go back to the **3rd Party Configuration** page.

   A non-sticky connection is one in which packets from client to server and from server to client pass through different cluster members. Non-sticky connections are a problem because they can lead to out-of-state packets being received by the cluster member. The Security Gateway will reject out-of-state packets, even if they belong to a valid connection.

   The synchronization mechanism, or the OPSEC certified clustering product, must identify valid non-sticky connections. This is necessary to allow these connections to go through the cluster.

   Find out whether or not the OPSEC certified clustering product can identify valid non-sticky connections.

   - If your clustering product cannot identify valid non-sticky connections, the synchronization mechanism can do this instead. In this case, select **Support non-sticky connections**.
   - If your clustering product can identify valid non-sticky connections, disable **Support non-sticky connections**. It is safe to disable this option in High Availability environments, but not for Load Sharing. It is safe to disable this option for High Availability environments, but for Load Sharing. This can cause a slight improvement in the connection rate.

     If the Hide Cluster Members' outgoing traffic behind the Clusters IP Address option is enabled, enable Support non-sticky connections to support outgoing connections from a standby member (unless specifically directed by OPSEC certified clustering product guide).
6. Many Security Gateway clusters have a virtual cluster IP address that is defined in **Topology** page of the cluster object, in addition to physical cluster member interface addresses. The use of virtual cluster IP addresses affects the settings in the **3rd Party Configuration** page.

   When a client behind the cluster establishes an *outgoing* connection towards the Internet, the source address in the outgoing packets, is usually the physical IP address of the cluster member interface. If virtual cluster IP addresses are used, the clustering product usually changes the source IP address (using NAT) to that of the external virtual IP address of the cluster.

   This corresponds to the default setting of **Hide Cluster Members' outgoing traffic behind the Cluster IP address** being checked.

   When a client establishes an *incoming* connection to the external virtual address of the cluster, the clustering product changes the destination IP address (using NAT) to that of the physical external address of one of the cluster members.

   This corresponds to the default setting of **Forward Cluster incoming traffic to Cluster Members' IP addresses** being checked. In the **Topology** page, define the interfaces for the

individual members. In most OPSEC solutions, cluster IPs should not be added to the individual member **Topology** tab. Refer to your clustering product documentation for additional information.

7. Define the other pages in the cluster object as required (**NAT, VPN**, **Remote Access**, and so on).
8. Install the Security Policy on the cluster.

Note - When you create an IPSO VRRP or IPSO IP cluster with version 3.9 and later, You must deactivate **monitor fw state** before the first Policy installation. If you do not do this, the **Get Interfaces** operation in the **Topology** section of the **Gateway Cluster Properties** window fails. After Policy installation, the **monitor fw state** feature can be re-enabled.

CPHA Command Line Behavior in OPSEC Clusters

This section describes the behavior of specific command lines in OPSEC clusters.

Note - For details of the `cpha` commands see Monitoring and Troubleshooting Clusters (on page 61).

The cphastart and cphastop Commands in OPSEC Clusters

The behavior of the **cphastart** and **cphastop** commands on ClusterXL clusters are described in The cphastart and cphastop Commands (on page 72).

On OPSEC clusters, the **cphastart** command may not cause the cluster member to start working. On IPSO clusters the behavior is the same as with ClusterXL clusters.

The **cphastop** command may not cause failover on OPSEC clusters. On IPSO IP Clustering clusters (but *not* on VRRP clusters), the behavior is the same as with ClusterXL clusters.

As with ClusterXL clusters, these commands should only be run by the Security Gateway, and not directly by the user.

The cphaprob Command in OPSEC Clusters

Use the **cphaprob** command to verify that the cluster and the cluster members are working properly. This command is relevant only for IPSO IP clustering and IPSO VRRP.

In non-IPSO OPSEC clusters the command output is either empty or the command does not have any effect.

To produce a usage printout for **cphaprob** that shows all the available commands, type **cphaprob** at the command line and press **Enter**. The meaning of each of these commands is explained in the following sections.

```
cphaprob -d <device> -t <timeout(sec)> -s <ok|init|problem> [-p] register
cphaprob -f <file> register
cphaprob -d <device> [-p] unregister
cphaprob -d <device> -s <ok|init|problem> report
cphaprob [-i[a]] [-e] list
cphaprob state
cphaprob [-a] if
```

`cphaprob state`: When running this command, the member state is only Check Point status and is not really a member status. The command only monitors full sync success, and if a policy was successfully installed. For IP clustering, the state is accurate and also includes the status of the IPSO Cluster. For VRRP, the status is accurate for a firewall, but it does not correctly reflect the status of the IPSO member (for example, it does not detect interface failure).

`cphaprob [-a] if`: Shows only the relevant information - interface name, if it is a sync interface or not. "Multicast"/"Broadcast" refers to the cluster control protocol and is relevant only for the sync interface. Note that the status of the interface is not printed since it is not monitored. (This also applies in the IPSO member.)

Monitoring and Troubleshooting Clusters

In This Section:

Making Sure that a Cluster is Working

The cphaprob Command

Use the `cphaprob` command to verify that the cluster and the cluster members are working properly, and to define critical devices. A critical device is a process running on a cluster member that enables the member to notify other cluster members that it can no longer function as a member. The device reports to the ClusterXL mechanism regarding its current state or it may fail to report, in which case ClusterXL decides that a failover has occurred and another cluster member takes over. When a critical device (also known as a Problem Notification, or pnote) fails, the cluster member is considered to have failed.

There are a number of built-in critical devices, and the Administrator can define additional critical devices. The default critical devices are:

- The cluster interfaces on the cluster members.
- Synchronization — full synchronization completed successfully.
- Filter — the Security Policy, and whether it is loaded.
- cphad — which follows the ClusterXL process called cphamcset.
- fwd — the Security Gateway daemon.

These commands can be run automatically by including them in scripts.

To produce a usage printout for cphaprob that shows all the available commands, type cphaprob at the command line and press Enter. The meaning of each of these commands is explained in the following sections.

```
chaprob -d <device> -t <timeout(sec)> -s <ok|init|problem> [-p] register
cphaprob -f <file> register
cphaprob -d <device> [-p] unregister
cphaprob -d <device> -s <ok|init|problem> report
cphaprob [-i[a]] [-e] list
cphaprob statecphaprob [-a] if
```

Monitoring Cluster Status

To see the status of a single or multiple cluster members:

- Run `cphaprob state`

  Do this command after setting up the cluster, and whenever you want to monitor the cluster status.

  The following is an example of the output of **cphaprob state**:

```
cphaprob state

Cluster mode:   Load Sharing (Multicast)

Number     Unique Address  State

1 (local)  30.0.0.1        active
2          30.0.0.2        active
```

- **Cluster mode** can be
  - Load Sharing (Multicast).
  - Load Sharing (Unicast).
  - High Availability New Mode (Primary Up or Active Up).
  - High Availability Legacy Mode (Primary Up or Active Up).
  - For third-party clustering products: "Service", refer to Clustering Definitions and Terms, for further information.
  - The number of the member indicates the member ID for Load Sharing, and the Priority for High Availability.
  - In Load Sharing configuration, all members in a fully functioning cluster should be *Active*. In High Availability configurations, only one member in a properly functioning cluster must be *Active*, and the others must be in the *Standby* state.

  Third-party clustering products show *Active/Active* even if one of the members is in standby state. This is because this command only reports the status of the full synchronization process. For IPSO VRRP, this command shows the exact state of the firewall, but not the cluster member (for example, the member may not be working properly but the state of the firewall is active).

When examining the state of the cluster member, you need to consider whether it is forwarding packets, and whether it has a problem that is preventing it from forwarding packets. Each state reflects the result of a test on critical devices. This is a list that explains the possible cluster states, and whether or not they represent a problem.

State	Meaning	Forwarding packets?	Is this state a Problem ?
Active	Everything is OK.	Yes	No
Active attention	A problem has been detected, but the cluster member is still forwarding packets because it is the only member in the cluster or there are no other active members in the cluster. In any other situation the state of the member would be *down*.	Yes	Yes
Down	One of the critical devices is down.	No	Yes
Ready	• State *Ready* means that the member recognizes itself as a part of the cluster and is literally ready to go into action, but, by design, something prevents the member from taking action. Possible reasons that the member is not yet Active include: • Not all required software components were loaded and initialized yet and/or not all configuration steps finished successfully yet. Before a cluster member becomes *Active*, it sends a message to the rest of the cluster members, checking whether it can become *Active*. In High Availability mode it will check if there is already an Active member and in Load Sharing Unicast mode it will check if there is a Pivot member already. The member remains in the *Ready* state until it receives the response from the rest of the cluster members and decides which state to choose next (A*ctive, Standby, Pivot, or non-Pivot*). • Software installed on this member has a higher version than the rest of the members in this cluster. For example, when a cluster is upgraded from one version of Check Point Security Gateway to another, and the cluster members have different versions of Check Point Security Gateway, the members with a new version have the *Ready* state and the members with the previous version have the *Active / Active Attention* state. • If the software installed on all cluster members includes CoreXL, which is installed by default in versions R70 and higher, a member in *Ready* state may have a higher number of CoreXL instances than other members. See sk42096 for a solution	No	No
Standby	Applies only to a High Availability configuration, and means the member is waiting for an active member to fail in order to start packet forwarding.	No	No
Initializing	An initial and transient state of the cluster member. The cluster member is booting up, and ClusterXL product is already running, but the Security Gateway is not yet ready.	No	No

State	Meaning	Forwarding packets?	Is this state a Problem ?
ClusterXL inactive or member is down	Local member cannot hear anything coming from this cluster member.	Unknown	Yes

Monitoring Cluster Interfaces

To see the state of the cluster member interfaces and the virtual cluster interfaces:

Run this command on the cluster members:

```
cphaprob [-a] if
```

The output of this command must be identical to the configuration in the cluster object **Topology** page.

For example:

```
cphaprob -a if

Required interfaces: 4
Required secured interfaces: 1

qfe4      UP                        (secured, unique, multicast)
qfe5      UP                        (non secured, unique, multicast)
qfe6      DOWN (4810.2 secs)        (non secured, unique, multicast)
qfe7      UP                        (non secured, unique, multicast)

Virtual cluster interfaces: 2
qfe5           30.0.1.130
qfe6           30.0.2.130
```

Interfaces are ClusterXL critical devices. ClusterXL makes sure that interfaces can send and receive CCP packets. It also sets the required minimum number of functional interfaces to the largest number of functional interfaces seen since the last reboot. If the number of functional interfaces is less than the required number, ClusterXL starts a failover. The same applies for secured interfaces, where only good synchronization interfaces are counted.

An interface can be:

- *Non-secured* or *Secured*. A secured interface is a synchronization interface.
- *Shared* or *unique*. A shared interface applies only to High Availability Legacy mode.
- *Multicast or broadcast*. The Cluster Control Protocol (CCP) mode used in the cluster. CCP can be changed to use broadcast instead. To toggle between these two modes use the command **cphaconf set_ccp <broadcast|multicast>**

For third-party clustering products, except in the case of IPSO IP Clustering, **cphaprob -a if** should always show virtual cluster IP addresses.

When an interface is DOWN, it means that the interface cannot receive or transmit CCP packets, or both. This may happen when an interface is malfunctioning, is connected to an incorrect subnet, is unable to pick up Multicast Ethernet packets and so on. The interface may also be able

to receive but not transmit CCP packets, in which case the status field is read. The displayed time is the number of seconds that have elapsed since the interface was last able to receive/transmit a CCP packet.

See Defining Disconnected Interfaces (on page 113) for additional information.

Monitoring Critical Devices

When a critical device fails, the cluster member is considered to have failed. To see the list of critical devices on a cluster member, and of all the other members in the cluster, run the following command on the cluster member:

```
cphaprob [-i[a]] [-e] list
```

There are a number of built-in critical devices, and the Administrator can define additional critical devices. The default critical devices are:

- The cluster interfaces on the cluster members.
- **Synchronization** — full synchronization completed successfully.
- **Filter** — the Security Policy, and whether it is loaded.
- **cphad** — which follows the ClusterXL process called **cphamcset**.
- **fwd** — the Security Gateway daemon.

For IPSO Clustering, the output is the same as for ClusterXL Load Sharing. For other third-party products, this command produces no output. The following example output shows that the **fwd** process is down:

```
cphaprob list

Built-in Devices:

Device Name: Interface Active Check
Current state: OK

Registered Devices:

Device Name: Synchronization
Registration number: 0
Timeout: none
Current state: OK
Time since last report: 15998.4 sec

Device Name: Filter
Registration number: 1
Timeout: none
Current state: OK
Time since last report: 15644.4 sec

Device Name: fwd
Registration number: 3
Timeout: 2 sec
Current state: problem
Time since last report: 4.5 sec
```

Registering a Critical Device

```
cphaprob -d <device> -t <timeout(sec)> -s <ok|init|problem> [-p] register
```

It is possible to add a user defined critical device to the default list of critical devices. Use this command to register <device> as a critical process, and add it to the list of devices that must be running for the cluster member to be considered active. If <device> fails, then the cluster member is considered to have failed.

If <device> fails to contact the cluster member in <timeout> seconds, <device> will be considered to have failed. For no timeout, use the value **0**.

Define the status of the <device> that will be reported to ClusterXL upon registration. This initial status can be one of:

- **ok** — <device> is alive.
- **init** — <device> is initializing. The member is down. This state prevents the member from becoming active.
- **problem** — <device> has failed.

[-p] makes these changes permanent. After performing a reboot or after removing the Security Gateway (on Linux or IPSO for example) and re-attaching it, the status of critical devices that were registered with this flag will be saved.

Registering Critical Devices Listed in a File

```
cphaprob -f <file> register
```

Register all the user defined critical devices listed in <file>. <file> must be an ASCII file, with each device on a separate line. Each line must list three parameters, which must be separated by at least a space or a tab, as follows:

```
<device> <timeout> <status>
```

- <device> — The name of the critical device. It must have no more than 15 characters, and must not include white spaces.
- <timeout> — If <device> fails to contact the cluster member in <timeout> seconds, <device> will be considered to have failed. For no timeout, use the value **0**.
- <status> — can be one of
- **ok** — <device> is alive.
- **init** — <device> is initializing. The member is down. This state prevents the member from becoming active.
- **problem** — <device> has failed.

Unregistering a Critical Device

```
cphaprob -d <device> [-p] unregister
```

Unregistering a user defined <device> as a critical process. This means that this device is no longer considered critical. If a critical device (and hence a cluster member) was registered as "problem" before running this command, then after running this command the status of the cluster will depend only on the remaining critical devices.

[-p] makes these changes permanent. This means that after performing a reboot or after removing the kernel (on Linux or IPSO for example) and re-attaching it, these critical devices remain unregistered.

Reporting Critical Device Status to ClusterXL

```
cphaprob -d <device> -s <ok|init|problem> report
```

Use this command to report the status of a user defined critical device to ClusterXL.

<device> is the device that must be running for the cluster member to be considered active. If <device> fails, then the cluster member is considered to have failed.

The status to be reported. The status can be one of:

ok — <device> is alive

init — <device> is initializing. The member is down. This state prevents the member from becoming active.

problem — <device> has failed. If this status is reported to ClusterXL, the cluster member will immediately failover to another cluster member.

If <device> fails to contact the cluster member within the timeout that was defined when the it was registered, <device> and hence the cluster member, will be considered to have failed. This is true only for critical devices with timeouts. If a critical device is registered with the **-t 0** parameter, there will be no timeout, and until the device reports otherwise, the status is considered to be the last reported status.

Example cphaprob Script

Predefined **cphaprob** scripts are located on the location **$FWDIR/bin**. Two scripts are available

- **ClusterXL_monitor_ips**
- **ClusterXL_monitor_process**

The **ClusterXL_monitor_ips** script in the Appendix chapter Example cphaprob Script (on page 143) has been designed to provide a way to check end-to-end connectivity to routers or other network devices and cause failover if the ping fails. The **ClusterXL_monitor_process** script monitors the existence of given processes and causes failover if the processes die. This script uses the normal `pnote` mechanism.

Monitoring Cluster Status Using SmartConsole Clients

SmartView Monitor

SmartView Monitor displays a snapshot of all ClusterXL cluster members in the enterprise, enabling real-time monitoring and alerting. For each cluster member, state change and critical device problem notifications are displayed. SmartView Monitor allows you to specify the action to be taken if the status of a cluster member changes. For example, the Security Gateway can issue an alert notifying you of suspicious activity.

Starting and Stopping ClusterXL Using SmartView Monitor

To stop ClusterXL on the member and cause failover to another member, open SmartView Monitor, click the cluster object, select one of the member branches, right click a cluster member, and select **Down**.

To initiate a restart of ClusterXL, open SmartView Monitor, click the cluster object, select one of the member branches, right click a cluster member, and select **Up**.

Note - SmartView Monitor does not initiate full synchronization, so that some connections may be lost. To initiate full synchronization, run **cpstart**.

SmartView Tracker

Every change in status of a cluster member is recorded in SmartView Tracker according to the choice in the **Fail-Over Tracking** option of the cluster object **ClusterXL** page.

ClusterXL Log Messages

The following conventions are used in this section:

1. Square brackets are used to indicate place holders, which are substituted by relevant data when an actual log message is issued (for example, [NUMBER] will be replaced by a numeric value).
2. Angle brackets are used to indicate alternatives, one of which will be used in actual log messages. The different alternatives are separated with a vertical line (for example, <up|down> indicates that either "up" or "down" will be used).
3. These place holders are frequently used:
   - ID: A unique cluster member identifier, starting from "1". This corresponds to the order in which members are sorted in the cluster object GUI.
   - IP: Any unique IP address that belongs to the member.
   - MODE: The cluster mode (for example, New HA, LS Multicast, and so on).
   - STATE: The state of the member (for example, active, down, standby).
   - DEVICE: The name of a pnote device (for example, fwd, Interface Active Check).

General logs

```
Starting <ClusterXL|State Synchronization>.
```

Indicates that ClusterXL (or State Synchronization, for 3rd party clusters) was successfully started on the reporting member. This message is usually issued after a member boots, or after an explicit call to `cphastart`.

```
Stopping <ClusterXL|State Synchronization>.
```

Informs that ClusterXL (or State Synchronization) was deactivated on this member. The member will no longer be a part of the cluster (even if configured to be so), until ClusterXL is restarted.

```
Unconfigured cluster Computers changed their MAC Addresses. Please reboot
the cluster so that the changes take affect.
```

This message is usually issued when a member is shut down, or after an explicit call to `cphastop`.

State logs

```
Mode inconsistency detected: member [ID] ([IP]) will change its mode to
[MODE]. Please re-install the security policy on the cluster.
```

This message should rarely happen. It indicates that another cluster member has reported a different cluster mode than is known to the local member. This is usually the result of a failure to

install the security policy on all cluster members. To correct this problem, install the Security Policy again.

Note - The cluster will continue to operate after a mode inconsistency has been detected, by altering the mode of the reporting member to match the other cluster members. However, it is highly recommended that the policy will be re-installed as soon as possible.

```
State change of member [ID] ([IP]) from [STATE] to [STATE] was cancelled,
since all other members are down. Member remains [STATE].
```

When a member needs to change its state (for example, when an active member encounters a problem and needs to bring itself down), it first queries the other members for their state. If all other members are down, this member cannot change its state to a non-active one (or else all members will be down, and the cluster will not function). Thus, the reporting member continues to function, despite its problem (and will usually report its state as "active attention").

```
member [ID] ([IP]) <is active|is down|is stand-by|is initializing>
([REASON]).
```

This message is issued whenever a cluster member changes its state. The log text specifies the new state of the member.

Pnote logs

Pnote log messages are issued when a pnote device changes its state.

- **[DEVICE] on member [ID] ([IP]) status OK ([REASON])**.
  The pnote device is working normally.
- **[DEVICE] on member [ID] ([IP]) detected a problem ([REASON]).**
  Either an error was detected by the pnote device, or the device has not reported its state for a number of seconds (as set by the "timeout" option of the pnote)
- **[DEVICE] on member [ID] ([IP]) is initializing ([REASON]).**
  Indicates that the device has registered itself with the pnote mechanism, but has not yet determined its state.
- **[DEVICE]** on member **[ID]** (**[IP]**) is in an unknown state (**[STATE ID]**) (**[REASON]**).
  This message should not normally appear. Contact Check Point Support.

Interface logs

- **interface [INTERFACE NAME] of member [ID] ([IP]) is up.**
  Indicates that this interface is working normally, meaning that it is able to receive and transmit packets on the expected subnet.
- **interface [INTERFACE NAME] of member [ID] ([IP]) is down (receive <up|down>, transmit <up|down>).**
  This message is issued whenever an interface encounters a problem, either in receiving or transmitting packets. Note that in this case the interface may still be working properly, as far as the OS is concerned, but is unable to communicate with other cluster members due to a faulty cluster configuration.
- **interface [INTERFACE NAME] of member [ID] ([IP]) was added.**
  Notifies users that a new interface was registered with the Security Gateway (meaning that packets arriving on this interface are filtered by the firewall). Usually this message is the result of activating an interface (such as issuing an `ifconfig up` command on Unix systems).

The interface will now be included in the ClusterXL reports (such as in SmartView Monitor, or in the output of `cphaprob -a if`). Note that the interface may still be reported as "Disconnected", in case it was configured as such for ClusterXL.

- **interface [INTERFACE NAME] of member [ID] ([IP}) was removed.**

  Indicates that an interface was detached from the Security Gateway, and is therefore no longer monitored by ClusterXL.

SecureXL logs

- **SecureXL device was deactivated since it does not support CPLS.**

  This message is the result of an attempt to configure a ClusterXL in Load Sharing Multicast mode over Security Gateways using an acceleration device that does not support Load Sharing. As a result, acceleration will be turned off, but the cluster will work in Check Point Load Sharing mode (CPLS).

Reason Strings

- **member [ID] ([IP]) reports more interfaces up.**

  This text can be included in a pnote log message describing the reasons for a problem report: Another member has more interfaces reported to be working, than the local member does. This means that the local member has a faulty interface, and that its counterpart can do a better job as a cluster member. The local member will therefore go down, leaving the member specified in the message to handle traffic.

- **member [ID] ([IP]) has more interfaces - check your disconnected interfaces configuration in the <discntd.if file|registry>**.

  This message is issued when members in the same cluster have a different number of interfaces. A member having less interfaces than the maximal number in the cluster (the reporting member) may not be working properly, as it is missing an interface required to operate against a cluster IP address, or a synchronization network. If some of the interfaces on the other cluster member are redundant, and should not be monitored by ClusterXL, they should be explicitly designated as "Disconnected". This is done using the file **$FWDIR/conf/discntd.if** (under Unix systems), or the Windows Registry.

- **[NUMBER] interfaces required, only [NUMBER] up.**

  ClusterXL has detected a problem with one or more of the monitored interfaces. This does not necessarily mean that the member will go down, as the other members may have less operational interfaces. In such a condition, the member with the highest number of operational interfaces will remain up, while the others will go down.

Working with SNMP Traps

You can configure and see SNMP traps for ClusterXL High Availability.

To configure an SNMP trap:

1. Connect to the Security Management Server with an SSH connection or a serial console.
2. Go to the **Expert** mode.
3. Run `threshold_config`.
4. Select **(9) Configure Thresholds** from the **Threshold Engine Configuration Options** menu.
5. Select **(2) High Availability** from the **threshold Engine Configuration Options** menu.
6. Select a trap from the **High Availability Thresholds** menu.

7. Select and configure these actions for the specified trap:
   - **Enable/Disable Threshold**
   - **Set Severity**
   - **Set Repetitions**
   - **Configure Alert Destinations**
8. From the **Threshold Engine Configuration Options** menu, select **(7) Configure alert destinations**.
9. Configure your alert destinations.
10. From the **Threshold Engine Configuration Options** menu, select **(3) Save policy**.

    You can optionally save the policy to a file.
11. Install policy.

To see all defined SNMP traps, run `threshold_config` and select **(8) View thresholds overview** from the **Threshold Engine Configuration Options** menu.

You can download the most recent Check Point MIBs file https://supportcenter.checkpoint.com/supportcenter/portal?eventSubmit_doGoviewsolutiondetails=&solutionid=sk90470&js_peid=P-114a7ba5fd7-10001&partition=General&product=All%22 from the Support Center.

ClusterXL Configuration Commands

The cphaconf command

Description The cphaconf command configures ClusterXL.

Important - Running this command is not recommended. It should be run automatically, only by the Security Gateway or by Check Point support. The only exception to this rule is running this command with `set_cpp` option, as described below.

Usage

```
cphaconf [-i <computer id>] [-p <policy id>] [-b <db_id>] [-n <ClusterXL num>]
[-c <ClusterXL size>] [-m <service >] [-t <secured IF 1>...] start

cphaconf [-t <secured IF 1>...] [-d <disconnected IF 1>...] add
cphaconf clear-secured
cphaconf clear-disconnected
cphaconf stop
cphaconf init
cphaconf forward <on/off>
cphaconf debug <on/off>
cphaconf set_ccp <broadcast/multicast>
cphaconf mc_reload
cphaconf debug_data
cphaconf stop_all_vs
```

Syntax

Parameter	Description
`set_ccp <broadcast/multicast>`	Sets whether ClusterXL Control Protocol (CCP) packets should be sent with a broadcast or multicast destination MAC address. The default behavior is multicast. The setting created using this command will survive reboot. **Note:** The same value (either broadcast or multicast) should be set on all ClusterXL members.
`stop_all_vs`	Stops the ClusterXL product on all Virtual Systems on a VSX Gateway.

The cphastart and cphastop Commands

Running **cphastart** on a cluster member activates ClusterXL on the member. It does not initiate full synchronization. **cpstart** is the recommended way to start a cluster member.

Running **cphastop** on a cluster member stops the cluster member from passing traffic. State synchronization also stops. It is still possible to open connections directly to the cluster member. In the High Availability Legacy mode, running **cphastop** may cause the entire cluster to stop functioning.

These commands should only be run by the Security Gateway, and not directly by the user.

How to Initiate Failover

Method	To Stop ClusterXL	To Start ClusterXL

Run:
`cphaprob -d faildevice -t 0 -s ok register`
`cphaprob -d faildevice -s problem report`
and:
`cphaprob -d faildevice -s ok report`
`cphaprob -d faildevice unregister`
	Effect:
	• Disables ClusterXL
	• Does not disable synchronization
		Effect:
		• Enables ClusterXL
		• Does not initiate full synchronization

Recommended method: Run: • `clusterXL_admin down` • `clusterXL_admin up`	• Disables ClusterXL • Does not disable synchronization	• Enables ClusterXL • Does not initiate full synchronization
In SmartView Monitor: 1. Click the Cluster object. 2. Select one of the member Security Gateway branches. 3. Right click the cluster member. 4. Select **Down**.	• Disables ClusterXL • Disables synchronization	• Enables ClusterXL • Does not initiate full synchronization

In Load Sharing mode, the cluster distributes the load between the remaining active members.

In HA mode, the cluster fails over to next *active* member with the highest priority.

For more on initiating manual failovers, see: sk55081
http://supportcontent.checkpoint.com/solutions?id=sk55081

Monitoring Synchronization (fw ctl pstat)

To monitor the synchronization mechanism on ClusterXL or third-party OPSEC certified clustering products:

- Run this command on a cluster member: `fw ctl pstat`

The output of this command is a long list of statistics for the Security Gateway. At the end of the list there is a section called "Synchronization" which applies per Cluster member. Many of the statistics are counters that can only increase. A typical output is as follows:

```
Version: new
Status: Able to Send/Receive sync packets
Sync packets sent:
 total : 3976,  retransmitted : 0, retrans reqs : 58,  acks : 97
Sync packets received:
 total : 4290,  were queued : 58, dropped by net : 47
 retrans reqs : 0, received 0 acks
 retrans reqs for illegal seq : 0
 Callback statistics: handled 3 cb, average delay : 1,  max delay : 2
```

```
Delta Sync memory usage: currently using XX KB mem
Callback statistics: handled 322 cb, average delay : 2,  max delay : 8
Number of Pending packets currently held: 1
Packets released due to timeout: 18
```

The meaning of each line in this printout is explained below.

```
 Version: new
```

This line must appear if synchronization is configured. It indicates that new sync is working (as opposed to old sync from version 4.1).

```
Status: Able to Send/Receive sync packets
```

If sync is unable to either send or receive packets, there is a problem. Sync may be temporarily unable to send or receive packets during boot, but this should not happen during normal operation. When performing full sync, sync packet reception may be interrupted.

```
Sync packets sent:
 total : 3976,  retransmitted : 0, retrans reqs : 58,  acks : 97
```

The total number of sync packets sent is shown. Note that the total number of sync packets is non-zero and increasing.

The cluster member *sends* a retransmission request when a sync packet is received out of order. This number may increase when under load.

Acks are the acknowledgments *sent* for received sync packets, when an acknowledgment was requested by another cluster member.

```
Sync packets received:
  total : 4290,  were queued : 58, dropped by net : 47
```

The total number of sync packets received is shown. The queued packets figure increases when a sync packet is received that complies with one of the following conditions:

1. The sync packet is received with a sequence number that does not follow the previously processed sync packet.
2. The sync packet is fragmented. This is done to solve MTU restrictions.

This figure never decreases. A non-zero value does not indicate a problem.

The **dropped by net** number may indicate network congestion. This number may increase slowly under load. If this number increases too fast, a networking error may be interfering with the sync protocol. In that case, check the network.

```
retrans reqs : 0, received 0 acks
 retrans reqs for illegal seq : 0
 Callback statistics: handled 3 cb, average delay : 1,  max delay : 2
```

This message refers to the number of *received* retransmission requests, in contrast to the transmitted retransmission requests in the section above. When this number grows very fast, it may indicate that the load on the member is becoming too high for sync to handle.

Acks refer to the number of acknowledgments *received* for the "cb request" sync packets, which are sync packets with requests for acknowledgments.

Retrans reqs for illegal seq displays the number of retransmission requests for packets which are no longer in this member possession. This may indicate a sync problem.

Callback statistics relate to received packets that involve Flush and Ack. This statistic only appears for a non-zero value.

The callback **average delay** is how much the packet was delayed in this member until it was released when the member received an ACK from all the other members. The delay happens

because packets are held until all other cluster members have acknowledged reception of that sync packet.

This figure is measured in terms of numbers of packets. Normally this number should be small (~1-5). Larger numbers may indicate an overload of sync traffic, which causes connections that require sync acknowledgments to suffer slight latency.

```
dropped updates as a result of sync overload: 0
```

In a heavily loaded system, the cluster member may drop synchronization updates sent from another cluster member.

```
Delta Sync memory usage: currently using XX KB mem
```

`Delta Sync memory usage` only appears for a non-zero value. Delta sync requires memory only while full sync is occurring. Full sync happens when the system goes up- after reboot for example. At other times, Delta sync requires no memory because Delta sync updates are applied immediately. For information about Delta sync see How State Synchronization Works (on page 17).

```
 Number of Pending packets currently held: 1
   Packets released due to timeout: 18
```

Number of Pending packets currently held only appears for a non-zero value. ClusterXL prevents out-of-state packets in non-sticky connections. It does this by holding packets until a SYN-ACK is received from all other active cluster members. If for some reason a SYN-ACK is not received, the Security Gateway on the cluster member will not release the packet, and the connection will not be established.

Packets released due to timeout only appears for a non-zero value. If the Number of Pending Packets is large (more than 100 pending packets), and the number of `Packets released due to timeout` is small, you should take action to reduce the number of pending packets. To solve this problem, see Reducing the Number of Pending Packets (on page 112).

Troubleshooting Synchronization

Introduction to cphaprob [-reset] syncstat

Heavily loaded clusters and clusters with geographically separated members pose special challenges. High connection rates, and large distances between the members can lead to delays that affect the operation of the cluster.

The **cphaprob [-reset] syncstat** command is a tool for monitoring the operation of the State Synchronization mechanism in highly loaded and distributed clusters. It can be used for both ClusterXL and third-party OPSEC certified clustering products.

The troubleshooting process is as follows:

1. Run the **cphaprob syncstat** command.
2. Examine and understand the output statistics.
3. Tune the relevant synchronization global configuration parameters.
4. Rerun the command, resetting the statistics counters using the -reset option:

   cphaprob -reset syncstat
5. Examine the output statistics to see if the problem is solved.

The section Output of cphaprob [-reset] syncstat (on page 76) explains each of the output parameters, and also explains when the output represents a problem.

Any identified problem can be solved by performing one or more of the tips described in **Synchronization Troubleshooting Options** (on page 82).

Output of cphaprob [-reset] syncstat

The output parameters of the **cphaprob syncstat** command are shown below. The values (not shown) give an insight into the state and characteristics of the synchronization network. Each parameter and the meaning of its possible values is explained in the following sections.

Parameters:

Sync Statistics (IDs of F&A Peers - 1)

These statistics relate to the state synchronization mechanism. The F&A (Flush and Ack) peers are the cluster members that this member recognizes as being part of the cluster. The IDs correspond to IDs and IP addresses generated by the **cphaprob state** command.

Other Member Updates

The statistics in this section relate to *updates* generated by other cluster members, or to updates that were not received from the other members. Updates inform about changes in the connections handled by the cluster member, and are sent from and to members. Updates are identified by sequence numbers.

Sent Retransmission Requests

The number of retransmission requests, which were sent by this member. Retransmission requests are sent when certain packets (with a specified sequence number) are missing, while the sending member already received updates with advanced sequences.

A high value can imply connectivity problems.

Note - Compare the number of retransmission requests to the Total Regenerated Updates of the other members (see Total Generated Updates (on page 78)).

If its value is unreasonably high (more than 30% of the Total Generated Updates of other members), contact Technical Support equipped with the entire output and a detailed description of the network topology and configuration.

Avg |Missing Updates per Request

Each retransmission request can contain up to 32 missing consecutive sequences. The value of this field is the average number of requested sequences per retransmission request.

More than 20 missing consecutive sequences per retransmission request can imply connectivity problems.

Note - If this value is unreasonably high, contact Technical Support, equipped with the entire output and a detailed description of the network topology and configuration.

Old or too-new Arriving Updates

The number of arriving sync updates where the sequence number is too low, which implies it belongs to an old transmission, or too high, to the extent that it cannot belong to a new transmission.

Large values imply connectivity problems.

Note - See Enlarging the Receiving Queue (on page 83) If this value is unreasonably high (more than 10% of the total updates sent), contact Technical Support, equipped with the entire output and a detailed description of the network topology and configuration.

Unsynchronized Missing Updates

The number of missing sync updates for which the receiving member stopped waiting. It stops waiting when the difference in sequence numbers between the newly arriving updates and the missing updates is larger than the length of the receiving queue.

This value should be zero. However, the loss of some updates is acceptable as long as the number of lost updates is less than 1% of the total generated updates.

Note - To decrease the number of lost updates, expand the capacity of the Receiving Queue. See Enlarging the Receiving Queue (on page 83).

Lost Sync Connection (num of events)

The number of events in which synchronization with another member was lost and regained due to either Security Policy installation on the other member, or a large difference between the expected and received sequence number.

The value should be zero. A positive value indicates connectivity problems.

Note - Allow the sync mechanism to handle large differences in sequence numbers by expanding the Receiving Queue capacity. See Enlarging the Receiving Queue (on page 83).

Timed out Sync Connection

The number of events in which the member declares another member as not connected. The member is considered as disconnected because no ACK packets were received from that member for a period of time (one second), even though there are Flush and Ack packets being held for that member.

The value should be zero. Even with a round trip time on the sync network as high as 100ms, one second should be enough time to receive an ACK. A positive value indicates connectivity problems.

Note - Try enlarging the Sync Timer (see Enlarging the Sync Timer (on page 83)). However, you may well have to contact Technical Support equipped with the entire output and a detailed description of the network topology and configuration.

Local Updates

The statistics in this section relate to *updates* generated by the local cluster member. Updates inform about changes in the connections handled by the cluster member, and are sent from and to members. Updates are identified by sequence numbers.

Total Generated Updates

- The number of sync update packets generated by the sync mechanism since the statistics were last reset. Its value is the same as the difference between the sequence number when applying the **-reset** option, and the current sequence number.
- Can have any value.

Recv Retransmission requests

- The number of received retransmission requests. A member requests retransmissions when it is missing specified packets with lower sequence numbers than the ones already received.
- A large value can imply connectivity problems.

Note - If this value is unreasonably high (more than 30% of the Total Generated Updates) contact Technical Support, equipped with the entire output and a detailed description of the network topology and configuration.

Recv Duplicate Retrans request

- The number of duplicated retransmission requests received by the member. Duplicate requests were already handled, and so are dropped.
- A large value may indicate network problem or storms on the sync network.

Note - If this value is unreasonably high (more than 30% of the Total Generated Updates) contact Technical Support, equipped with the entire output and a detailed description of the network topology and configuration.

Blocking Scenarios

Under extremely heavy load conditions, the cluster may block new connections. This parameter shows the number of times that the cluster member started blocking new connections due to sync overload.

The member starts to block connections when its Sending Queue has reached its capacity threshold. The capacity threshold is calculated as 80% of the difference between the current sequence number and the sequence number for which the member received an ACK from all the other operating members.

A positive value indicates heavy load. In this case, observe the *Blocked Packets* to see how many packets we blocked. Each dropped packet means one blocked connection.

This parameter is only measured if the *Block New Connections* mechanism (described in Blocking New Connections Under Load (on page 110)) is active.

To activate the Block New Connections mechanism:

Apply the **fw ctl set int fw_sync_block_new_conns 0** command to all the cluster members.

Note - The best way to handle a severe blocking connections problem is to enlarge the sending queue. See Enlarging the Sending Queue (on page 82).

Another possibility is to decrease the timeout after which a member initiates an ACK. See Reconfiguring the Acknowledgment Timeout (on page 84). This updates the sending queue capacity more accurately, thus making the blocking process more precise.

Blocked Packets

The number of packets that were blocked because the cluster member was blocking all new connections (see *Blocking Scenarios*). The number of blocked packets is usually one packet per new connection attempt.

A value higher than 5% of the Sending Queue (see Avg Length of Sending Queue (on page 80)) can imply a connectivity problem, or that ACKs are not being sent frequently enough.

This parameter is only measured if the *Block New Connections* mechanism (described in Blocking New Connections Under Load (on page 110)) is active.

To activate the Block New Connections mechanism:

Apply the **fw ctl set int fw_sync_block_new_conns 0** command on all the cluster members.

Note - The best way to handle a severe blocking connections problem is to enlarge the sending queue. See Enlarging the Sending Queue (see "Enlarging the Sync Timer" on page 83).

Another possibility is to decrease the timeout after which a member initiates an ACK. See Reconfiguring the Acknowledgment Timeout (on page 84). This updates the sending queue capacity more accurately, thus making the blocking process more precise.

Max Length of Sending Queue

The size of the Sending Queue is fixed. By default it is 512 sync updates. As newer updates with higher sequence numbers enter the queue, older updates with lower sequence numbers drop off the end of the queue. An older update could be dropped from the queue before the member receives an ACK about that update from all the other members.

This parameter is the difference between the current sync sequence number and the last sequence number for which the member received an ACK from all the other members. The value of this parameter can therefore be greater than 512.

The value of this parameter should be less than 512. If larger than 512, there is not necessarily a sync problem. However, the member will be unable to answer retransmission request for updates which are no longer in its queue.

This parameter is only measured if the *Block New Connections* mechanism (described in Blocking New Connections Under Load (on page 110)) is active.

To activate the Block New Connections mechanism:

Apply the **fw ctl set int fw_sync_block_new_conns 0** command on all the cluster members.

Note - Enlarge the Sending Queue to value larger than this value. See Enlarging the Sending Queue (on page 82).

Avg Length of Sending Queue

The average value of the **Max Length of Sending Queue** parameter, since reboot or since the Sync statistics were reset.

The value should be up to 80% of the size of the Sending Queue.

This parameters is only measured if the *Block New Connections* mechanism (described in Blocking New Connections Under Load (on page 110)) is active.

To activate the Block New Connections mechanism:

Apply the **fw ctl set int fw_sync_block_new_conns 0** command on all the cluster members.

Note - Enlarge the Sending Queue so that this value is not larger than 80% of the new queue size. See Enlarging the Sending Queue (on page 82).

Hold Pkts Events

The number of occasions where the sync update required Flush and Ack, and so was kept within the system until an ACK arrived from all the other functioning members.

Should be the same as the number of **Unhold Pkt Events**.

Note - Contact Technical Support equipped with the entire output and a detailed description of the network topology and configuration.

Unhold Pkt Events

The number of occasions when the member received all the required ACKS from the other functioning members.

Should be the same as the number of **Hold Pkts Events**.

Note - Contact Technical Support equipped with the entire output and a detailed description of the network topology and configuration.

Not Held Due to no Members

The number of packets which should have been held within the system, but were released because there were no other operating members.

When the cluster has at least two live members, the value should be 0.

Note - The cluster has a connectivity problem. Examine the values of the parameters: Lost Sync Connection (num of events) (on page 78) and Timed out Sync Connection (on page 78) to find out why the member thinks that it is the only cluster member.

You may also need to contact Technical Support equipped with the entire output and a detailed description of the network topology and configuration.

Max Held Duration (ticks)

The maximum time in ticks (one tick equals 100ms) for which a held packet was delayed in the system for Flush and Ack purposes.

It should not be higher than 50 (5 seconds), because of the pending timeout mechanism which releases held packets after a certain timeout. By default, the release timeout is 50 ticks. A high value indicates connectivity problem between the members.

Note - Optionally change the default timeout by changing the value of the fwldbcast_pending_timeout global variable. See Advanced Cluster Configuration (on page 109) and Reducing the Number of Pending Packets (on page 112).

Also, examine the parameter Timed out Sync Connection (on page 78) to understand why packets were held for a long time.

You may also need to contact Technical Support equipped with the entire output and a detailed description of the network topology and configuration.

Avg Held Duration (ticks)

The average duration in ticks (tick equals 100ms) that held packets were delayed within the system for Flush and Ack purposes.

The average duration should be about the round-trip time of the sync network. A larger value indicates connectivity problem.

Note - If the value is high, contact Technical Support equipped with the entire output and a detailed description of the network topology and configuration in order to examine the cause to the problem.

Timers

The Sync and CPHA timers perform sync and cluster related actions every fixed interval.

Sync tick (ms)

The *Sync timer* performs cluster related actions every fixed interval. By default, the Sync timer interval is 100ms. The base time unit is 100ms (or 1 *tick*), which is also the minimum value.

CPHA tick (ms)

The *CPHA timer* performs cluster related actions every fixed interval. By default, the CPHA timer interval is 100ms. The base time unit is 100ms (or 1 *tick*), which is also the minimum value.

Queues

Each cluster member has two queues. The Sending Queue and the Receiving Queue.

Sending Queue Size

The Sending Queue on the cluster member stores locally generated sync updates. Updates in the Sending Queue are replaced by more recent updates. In a highly loaded cluster, updates are therefore kept for less time. If a member is asked to retransmit an update, it can only do so if the update is still in its Sending Queue. The default (and minimum) size of this queue is 512. Each member has one sending queue.

Receiving Queue Size

The Receiving Queue on the cluster member keeps the updates from each cluster member until it has received a complete sequence of updates. The default (and minimum) size of this queue is 256. Each member keeps a Receiving Queue for each of the peer members.

Synchronization Troubleshooting Options

The following options specify the available troubleshooting options. Each option involves editing a global system configurable parameter to reconfigure the system with different value than the default.

Enlarging the Sending Queue

The Sending Queue on the cluster member stores locally generated sync updates. Updates in the Sending Queue are replaced by more recent updates. In a highly loaded cluster, updates are therefore kept for less time. If a member is asked to retransmit an update, it can only do so if the update is still in its Sending Queue. The default (and minimum) size of this queue is 512. Each member has one sending queue.

To enlarge the sending queue size:

1. Change the value of the global parameter **fw_sync_sending_queue_size**. See Advanced Cluster Configuration (on page 109).
2. You must also make sure that the required queue size survives boot. See How to Configure a Security Gateway to Survive a Boot (see "How to Configure Reboot Survival" on page 109).

Enlarging this queue allows the member to save more updates from other members. However, be aware that each saved update consumes memory. When changing this variable you should consider carefully the memory implications. Changes will only take effect after reboot.

Enlarging the Receiving Queue

The Receiving Queue on the cluster member keeps the updates from each cluster member until it has received a complete sequence of updates. The default (and minimum) size of this queue is 256. Each member keeps a Receiving Queue for each of the peer members.

To enlarge the receiving queue size:

1. Change the value of the global parameter **fw_sync_recv_queue_size**. See Advanced Cluster Configuration (on page 109).
2. You must also make sure that the required queue size survives boot. See How to Configure Security Gateway to Survive a Boot (see "How to Configure Reboot Survival" on page 109).

Enlarging this queue means that the member can save more updates from other members. However, be aware that each saved update consumes memory. When changing this variable you should carefully consider the memory implications. Changes will only take effect after reboot.

Enlarging the Sync Timer

The *sync timer* performs sync related actions every fixed interval. By default, the sync timer interval is 100ms. The base time unit is 100ms (or 1 *tick*), which is therefore the minimum value.

To enlarge the sync timer:

Change the value of the global parameter **fwha_timer_sync_res**. See Advanced Cluster Configuration (on page 109). The value of this variable can be changed while the system is working. A reboot is not needed.

By default, **fwha_timer_sync_res** has a value of **1**, meaning that the sync timer operates every base time unit (every 100ms). If you configure this variable to n, the timer will be operated every n*100ms.

Enlarging the CPHA Timer

The *CPHA timer* performs cluster related actions every fixed interval. By default, the CPHA timer interval is 100ms. The base time unit is 100ms (or 1 *tick*), which is also the minimum value.

If the cluster members are geographically separated from each other, set the CPHA timer to be around 10 times the round-trip delay of the sync network.

Enlarging this value increases the time it takes to detect a failover. For example, if detecting interface failure takes 0.3 seconds, and the timer is doubled to 200ms, the time needed to detect an interface failure is doubled to 0.6 seconds.

To enlarge the CPHA timer:

Change the value of the global parameter **fwha_timer_cpha_res**. See Advanced Cluster Configuration (on page 109). The value of this variable can be changed while the system is working. A reboot is not needed.

By default, **fwha_timer_cpha_res** has a value of **1**, meaning that the CPHA timer operates every base time unit (every 100ms). If you configure this variable to n, the timer will be operated every n*100ms.

Reconfiguring the Acknowledgment Timeout

A cluster member deletes updates from its Sending Queue (described in Sending Queue Size (on page 82)) on a regular basis. This frees up space in the queue for more recent updates.

The cluster member deletes updates from this queue if it receives an ACK about the update from the peer member.

The peer member sends an ACK in one of two circumstances — on condition that the Block New Connections mechanism (described in Blocking New Connections Under Load (on page 110)) is active:

- After receiving a certain number of updates.
- If it didn't send an ACK for a certain time. This is important if the sync network has a considerable line delay, which can occur if the cluster members are geographically separated from each other.

To reconfigure the timeout after which the member sends an ACK:

Change the value of the global parameter **fw_sync_ack_time_gap**. See Advanced Cluster Configuration (on page 109). The value of this variable can be changed while the system is working. A reboot is not needed.

The default value for this variable is 10 ticks (10 * 100ms). Thus, if a member didn't send an ACK for a whole second, it will send an ACK for the updates it received.

Contact Technical Support

If the other recommendations do not help solve the problem, contact Technical Support for further assistance.

Troubleshooting Dynamic Routing (routeD) Pnotes

In R76, Check Point added a new ClusterXL Pnote called **routeD** that works with Dynamic Routing for Gaia clusters. This Pnote makes sure that traffic is not assigned to a cluster member before it is ready to handle the traffic. The Gaia RouteD daemon handles all routing (static and dynamic) operations.

There can be an issue with Dynamic Routing that shows one or more of these symptoms:

- Cluster IP address connectivity problems
- Unexpected failovers
- SmartView Tracker logs show that a member is down because a routeD Pnote is set to **problem**.

- The `cphaprob list` command shows:

```
Device Name: routed
Registration number: 4
Timeout: none
Current state: problem
```

These are some of the common causes of this issue:

- Cluster misconfiguration
- Port 2010 is blocked by the Firewall
- The routeD daemon did not get all of its routes
- The routeD daemon did not start correctly

Standard RouteD Pnote Behavior

Typically, the routeD Pnote is set to **Problem** when:

- A member fails over
- A cluster member reboots
- There is an inconsistency in the Dynamic Routing configuration on cluster members

The routeD pnote is set to **ok** when:

- A ClusterXL member tells the routeD daemon that it is a Master
- The routeD daemon gets the entire routing state from the Master

Basic Troubleshooting Steps

1. Run `cphaprob -a if` to make sure that your cluster and member interfaces are configured correctly.
2. Run `dbset routed:instance:default:traceoptions:traceoptions:Cluster` to generate routeD cluster messages. The messages are located at `/var/log/routed/log`.
3. Make sure that Firewall rules do not block port 2010.
4. Make sure that the routeD daemon is running on the Active member.
5. Look for a `router-id` mismatch in the OSPF configuration.
6. Make sure that the OSPF interface is up on the Standby member.

For advanced troubleshooting procedures and more information, see sk92787 http://supportcontent.checkpoint.com/solutions?id=sk92787.

For troubleshooting OSPF and the routeD daemon, see sk84520 http://supportcontent.checkpoint.com/solutions?id=sk84520.

ClusterXL Error Messages

This section lists the ClusterXL error messages.

General ClusterXL Error Messages

- **FW-1: changing local mode from <mode1> to <mode2> because of ID <member_id>**

This log message can happen if the working mode of the cluster members is not the same, for example, if one member is running High Availability, and another Load Sharing Multicast or Unicast mode. In this case, the internal ClusterXL mechanism tries to synchronize the configuration of the cluster members, by changing the working mode to the lowest common mode. The order of priority of the working modes (highest to lowest) is: 1. Synchronization only 2. Load Sharing 3. High Availability (Active Up) 4. High Availability (Primary Up).

- **CPHA: Received confirmations from more members than the cluster size**

  This log message can occur during policy installation on the cluster. It means that a serious configuration problem exists in that cluster. Probably some other cluster has been configured with identical parameters and both of them have common networks.

- **fwldbcast_timer: peer X probably stopped...**

  This is caused when the member that printed this message stops hearing certain types of messages from member X. Verify that **cphaprob state** shows all members as active and that **fw ctl pstat** shows that sync is configured correctly and working properly on all members. In such a case it is fair to assume that there was a temporary connectivity problem that was fixed in the meantime. There may be several connections that may suffer from connectivity problems due to that temporary synchronization problem between the two members. On the other hand, this can indicate that the other member is really down.

- **FW-1: fwha_notify_interface: there are more than 4 IPs on interface <interface name> notifying only the first ones**

  A member of the same cluster as the reporting member has more than three virtual IP addresses defined on the same interface. This is not a supported configuration and will harm ClusterXL functionality.

- **FW-1: h_slink: an attempt to link to a link**
  kbuf id not found
  fw_conn_post_inspect: fwconn_init_links failed

  Several problems of this sort can happen during a full sync session when there are connections that are opened and closed during the full sync process. Full sync is automatic as far as possible, but it is not fully automatic for reasons of performance, A Security Gateway continues to process traffic even when it is serving as a full sync server. This can cause some insignificant problems, such as a connection that is being deleted twice, a link to an existing link, and so forth. It should not affect connectivity or cause security issues.

- **Error SEP_IKE_owner_outbound: other cluster member packet in outbound**

  Cluster in not synchronized. Usually happens in OPSEC certified third-party Load Sharing products for which **Support non-sticky connections** is unchecked in the cluster object **3rd Party Configuration** page.

- **FW-1: fwha_pnote_register: too many registering members, cannot register**

  The critical device (also known as Problem Notification, or pnote) mechanism can only store up to 16 different devices. An attempt to configure the 17th device (either by editing the **cphaprob.conf** file or by using the **cphaprob -d ... register** command) will result in this message.

- **FW-1: fwha_pnote_register: <NAME> already registered (# <NUMBER>)**

  Each device registered with the pnote mechanism must have a unique name. This message may happen when registering new pnote device, and means that the device **<NAME>** is already registered as with pnote number **<NUMBER>**.

- **FW-1: fwha_pnote_unregister: attempting to unregister an unregistered device <DEVICE NAME>**

Indicates an attempt to unregister a device which is not currently registered.

- **FW-1: alert_policy_id_mismatch: failed to send a log**

  A log indicating that there is a different policy id between the two or more members was not sent. Verify all cluster members have the same policy (using **fw stat**). It is recommended to re-install the policy.

- **FW-1: fwha_receive_fwhap_msg: received incomplete HAP packet (read <number> bytes)**

  This message can be received when ClusterXL hears CCP packets of clusters of version 4.1. In that case it can be safely ignored.

SmartView Tracker Active Mode Messages

The following error messages can appear in SmartView Tracker Active mode. These errors indicate that some entries may not have been successfully processed, which may lead to missing synchronization information on a cluster member and inaccurate reports in SmartView Tracker.

- **FW-1: fwlddist_adjust_buf: record too big for sync. update Y for table <id> failed. fwlddist_state=<val>**

  Indicates a configuration problem on a clustered member. Either synchronization is misconfigured, or there is a problem with transmitting packets on the sync interface. To get more information on the source of the problem

- Run **fw ctl pstat** (described in Monitoring Synchronization (fw ctl pstat) (on page 73)).
- In ClusterXL clusters, run **cphaprob -a if** to get the statuses of the interfaces (see Monitoring Cluster Interfaces (on page 64)).
- **FW-1: fwldbcast_flush: active connections is currently enabled and due to high load it is making sync too slow to function properly. X active updates were dropped**

  Indicates that a clustered member has dropped SmartView Tracker Active mode updates in order to maintain sync functionality.

Sync Related Error Messages

- **FW-1: fwldbcast_retreq: machine <MACHINE_ID> sent a retrans request for seq <SEQ_NUM> which is no longer in my possession (current seq <SEQ_NUM>)**

  This message appears when the local member receives a retransmission request for a sequence number which in no longer in its sending window. This message can indicate a sync problem if the sending member didn't receive the requested sequence.

- **FW-1: fwlddist_save: WARNING: this member will not be fully synchronized !**
  FW-1: fwlddist_save: current delta sync memory during full sync has reached the maximum of <MEM_SIZE> MB
  FW-1: fwlddist_save: it is possible to set a different limit by changing fw_sync_max_saved_buf_mem value

  These messages may appear only during full sync. While performing full sync the delta sync updates are being saved and are applied only after the full sync process has finished. It is possible to limit the memory used for saving delta sync updates by setting the **fw_sync_max_saved_buf_mem** variable to this limit.

- **FW-1: fwldbcast_flush: fwlddist_buf_ldbcast_unread is not being reset fast enough (ur=<UNREAD_LOC>,fwlddist_buflen=<BUFFER_LEN>)**

  This message may appear due to high load resulting in the sync buffer being filled faster than it is being read.

- **FW-1: fwlddist_mode_change: Failed to send trap requesting full sync**

  This message may appear due to a problem starting the full sync process, and indicates a severe problem. Contact Technical Support.

- **FW-1: State synchronization is in risk. Please examine your synchronization network to avoid further problems!**

  This message could appear under extremely high load, when a synchronization update was permanently lost. A synchronization update is considered to be permanently lost when it cannot be retransmitted because it is no longer in the transmit queue of the update originator. This scenario does not mean that the Security Gateway will malfunction, but rather that there is a potential problem. The potential problem is harmless if the lost sync update was to a connection that runs only on a single member as in the case of unencrypted (clear) connections (except in the case of a failover when the other member needs this update).

  The potential problem can be harmful when the lost sync update refers to a connection that is non-sticky (see Non-Sticky Connections (on page 23)), as is the case with encrypted connections. In this case the other cluster member(s) may start dropping packets relating to this connection, usually with a **TCP out of state** error message (see TCP Out-of-State Error Messages (on page 88)). In this case it is important to block new connections under high load, as explained in Blocking New Connections Under Load (on page 110).

  The following error message is related to this one.

- **FW-1: fwldbcast_recv: delta sync connection with member <MACHINE_ID> was lost and regained. <UPDATES_NUM> updates were lost.**
  FW-1: fwldbcast_recv: received sequence <SEQ_NUM> (fragm <FRAG_NUM>, index <INDEX_NUM>), last processed seq <SEQ_NUM>

  These messages appear when there was a temporary sync problem and some of the sync updates were not synchronized between the members. As a result some of the connections might not survive a failover.

  The previous error message is related to this one.

- **FW-1: The use of the non_sync_ports table is not recommended anymore. Refer to the user guide for configuring selective sync instead**

  Previous versions used a kernel table called **non_sync_ports** to implement selective sync, which is a method of choosing services that don't need to be synchronized. Selective sync can now be configured from SmartConsole. See Choosing Services That Do Not Require Synchronization (see "Configuring Services not to Synchronize" on page 17).

TCP Out-of-State Error Messages

When the synchronization mechanism is under load, TCP packet out-of-state error messages may appear in the Information column of SmartView Tracker. This section explains how to resolve each error.

- **TCP packet out of state - first packet isn't SYN tcp_flags: FIN-ACK**
  TCP packet out of state - first packet isn't SYN tcp_flags: FIN-PUSH-ACK

  These messages occur when a FIN packet is retransmitted after deleting the connection from the connection table. To solve the problem, in SmartConsole **Global properties** for **Stateful Inspection**, enlarge the TCP end timeout from 20 seconds to 60 seconds. If necessary, also enlarge the connection table so it won't fill completely.

- **SYN packet for established connection**

  This message occurs when a SYN is received on an established connection, and the sequence verifier is turned off. The sequence verifier is turned off for a non-sticky connection in a cluster

(or in SecureXL). Some applications close connections with a RST packet (in order to reuse ports). To solve the problem, enable this behavior to specific ports or to all ports. For example, run the command:
fw ctl set int fw_trust_rst_on_port <port>
Which means that the Security Gateway should trust a RST coming from every port, in case a single port is not enough.

Platform Specific Error Messages

IPSO Specific Error Messages

- **FW-1: fwha_nok_get_mc_mac_by_ip: received a NULL query**
  FW-1: fwha_nok_get_mc_mac_by_ip: nokcl_get_clustermac returned unknown type <TYPE>

  These messages mean that automatic proxy ARP entries for static NAT configuration might not be properly installed.

- **FW-1: fwha_nokcl_sync_rx_f: received NULL mbuf from ipso. Packet dropped.**
  FW-1: fwha_nokcl_sync_rx_f: received packet with illegal flag=<FLAG>. drop packet.

  These messages mean that an illegal CPHA packet was received and will be dropped. If this happens more than few times during boot, the cluster malfunctions.

- **FW-1: fwha_nokcl_reregister_rx: unregister old magic mac values with IPSO.**
  FW-1: fwha_nokcl_reregister_rx: new magic mac values <MAC,FORWARD MAC> registered successfully with IPSO.

  A notification that the operation **fw ctl set int fwha_magic_mac** succeeded.

- **FW-1: fwha_nokcl_reregister_rx: error in de-registration to the sync_rx (<ERR NUM>) new magic macs values will not be applied**

  A notification that the operation **fw ctl set int fwha_magic_mac** failed. Previous MAC values will be retained.

- **FW-1: fwha_nokcl_creation_f: error in registration ...**
  FW-1: fwha_nok_init: NOT calling nokcl_register_creation since did not de-register yet.
  FW-1: fwha_nok_fini: failed nokcl_deregister_creation with rc=<ERROR NUM>

  These messages mean that an internal error in registration to the IPSO clustering mechanism has occurred. Verify that the IPSO version is supported by this the Security Gateway version and that the IPSO IP Clustering or VRRP cluster is configured properly.

- **FW-1: successfully (dis)connected to IPSO Clustering**

  A notification that should be normally received during Security Gateway initialization and removal.

- **FW-1: fwha_pnote_register: noksr_register_with_status failed**
  FW-1: fwha_IPSO_pnote_expiration: mismatch between IPSO device to ckp device <DEVICE NAME>
  FW-1: fwha_nokia_pnote_expiration: cannot find the expired device
  FW-1: fwha_noksr_report_wrapper: attempting to report an unregistered device <DEVICE NAME>

  These messages may appear as a result of a problem in the interaction between the IPSO and ClusterXL device monitoring mechanisms. A reboot should solve this problem. Should this problem repeat itself contact Check Point Technical support.

Member Fails to Start After Reboot

If a reboot (or **cpstop** followed by **cpstart**) is performed on a cluster member while the cluster is under severe load, the member may fail to start correctly. The starting member will attempt to perform a full sync with the existing active member(s) and may in the process use up all its resources and available memory. This can lead to unexpected behavior.

To overcome this problem, define the maximum amount of memory that the member may use when starting up for synchronizing its connections with the active member. By default this amount is not limited. Estimate the amount of memory required as follows:

Number of open Connections	**New connections/second**			
	100	**1000**	**5000**	**10,000**
1000	1.1	6.9		
10000	11	69	329	
20000	21	138	657	1305
50000	53	345	1642	3264

Note - These figures were derived for cluster members using the Windows platform, with Pentium 4 processors running at 2.4 GHz.

For example, if the cluster holds 10,000 connections, and the connection rate is 1000 connections/sec you will need 69 MB for full sync.

Define the maximum amount of memory using the Security Gateway global parameter: **fw_sync_max_saved_buf_mem**.

The units are in megabytes. For details, see Advanced Cluster Configuration (on page 109).

Advanced Features and Procedures

In This Section:

Working with VPNs and Clusters

Configuring VPN and Clusters

Configuring a Security Gateway cluster using SmartConsole is very similar to configuring a single Security Gateway. All attributes of the VPN are defined in the Cluster object, except for two attributes that are defined per cluster member.

1. Go to the **Cluster Properties** window, **Cluster Members** page. For each cluster member, in the **Cluster member Properties** window, configure the **VPN** tab:
   - **Office Mode for Remote access** — If you wish to use Office Mode for remote access, define the IP pool allocated to each cluster member.
   - **Hardware Certificate Storage List** — If your cluster member supports hardware storage for IKE certificates, define the certificate properties. In that case, Security Management Server directs the cluster member to create the keys and supply only the required material for creation of the certificate request. The certificate is downloaded to the cluster member during policy installation.
2. In a VPN cluster, IKE keys are synchronized. In the **Synchronization** page of the **Cluster Properties** window, make sure that **Use State Synchronization** is selected, even for High Availability configurations.
3. In the **Topology** page of the **Cluster Properties** window, define the encryption domain of the cluster. Under **VPN Domain**, choose one of the two possible settings:
   - **All IP addresses behind cluster members based on topology information**. This is the default option.
   - **Manually Defined**. Use this option if the cluster IP address is not on the member network, in other words, if the cluster virtual IP address is on a different subnet than the cluster

member interfaces. In that case, select a network or group of networks, which must include the virtual IP address of the cluster, and the network or group of networks behind the cluster.

Defining VPN Peer Clusters with Separate Security Management Servers

When working with a VPN peer that is a Check Point Cluster, and the VPN peer is managed by a different Security Management Server, do NOT define another cluster object. Instead, do the following:

1. In the objects tree, **Network Objects** branch, right click and select **New Check Point Externally Managed Security Gateway**.
2. In the **Topology** page, add the external and internal *cluster* interface addresses of the VPN peer. Do not use the cluster member interface addresses, except in the following cases:
   - If the external cluster is of version 4.1, add the IP addresses of the cluster member interfaces.
   - If the cluster is an OPSEC certified product (excluding IPSO), you may need to add the IP addresses of the cluster members.

   When adding cluster member interface IP addresses, in the interface **Topology** tab, define the interface as **Internal**, and the **IP Addresses behind this interface** as **Not defined**.
3. In the **VPN Domain** section of the page, define the encryption domain of the externally managed Security Gateway to be behind the internal virtual IP address of the Security Gateway. If the encryption domain is just one subnet, choose **All IP addresses behind cluster members based on topology information**. If the encryption domain includes more than one subnet, it must be **Manually Defined**.

Working with NAT and Clusters

Cluster Fold and Cluster Hide

Network Address Translation (NAT) is a fundamental aspect of the way ClusterXL works.

When a cluster member establishes an *outgoing* connection towards the Internet, the source address in the outgoing packets, is the physical IP address of the cluster member interface. The source IP address is changed using NAT to that of the external virtual IP address of the cluster. This address translation is called "Cluster Hide".

For OPSEC certified clustering products, this corresponds to the default setting in the **3rd Party Configuration** page of the cluster object, of **Hide Cluster Members' outgoing traffic behind the Cluster IP address** being checked.

When a client establishes an *incoming* connection to external (virtual) address of the cluster, ClusterXL changes the destination IP address using NAT to that of the physical external address of one of the cluster members. This address translation is called "Cluster Fold".

For OPSEC certified clustering products, this corresponds to the default setting in the **3rd Party Configuration** page of the cluster object, of **Forward Cluster incoming traffic to Cluster Members' IP addresses** being checked.

Configuring NAT on the Cluster

Network Address Translation (NAT) can be performed on a Cluster, in the same way as it is performed on a Security Gateway. This NAT is in addition to the automatic "Cluster Fold" and "Cluster Hide" address translations.

To configure NAT, edit the Cluster object, and in the **Cluster Properties** window, select the **NAT** page. Do NOT configure the **NAT** tab of the cluster member object.

Configuring NAT on a Cluster Member

It is possible to perform Network Address Translation (NAT) on a non-cluster interface of a cluster member.

A possible scenario for this is if the non-Cluster interface of the cluster member is connected to another (non-cluster) internal Security Gateway, and you wish to hide the address of the non-Cluster interface of the cluster member.

Performing this NAT means that when a packet originates behind or on the non-Cluster interface of the cluster member, and is sent to a host on the other side of the internal Security Gateway, the source address of the packet will be translated.

To configure NAT on a non-cluster interface of a cluster member Security Gateway:

1. Edit the Cluster object.
2. In the **Cluster Member** page of the **Cluster Properties** window, edit the Cluster Member object.
3. In the **Cluster Member Properties** window, click the **NAT** tab.
4. Configure Static or Hide NAT as desired.

Working with VLANS and Clusters

VLAN Support in ClusterXL

A VLAN switch tags packets that originate in a VLAN with a four-byte header that specifies which switch port it came from. No packet is allowed to go from a switch port in one VLAN to a switch port in another VLAN, apart from ports ("global" ports) that are defined so that they belong to all the VLANs.

The cluster member is connected to the global port of the VLAN switch, and this logically divides a single physical port into many VLAN ports each associated with a VLAN tagged interface (VLAN interface) on the cluster member.

When defining VLAN tags on an interface, cluster IP addresses can be defined only on the VLAN interfaces (the tagged interfaces). Defining a cluster IP address on a physical interface that has VLANs is not supported. This physical interface has to be defined with the Network Objective **Monitored Private**.

Note - ClusterXL does not support VLANS on Windows 2000 or Windows 2003 Server.

Connecting Several Clusters on the Same VLAN

It is not recommended to connect the non-secured interfaces (the internal or external cluster interfaces, for example) of multiple clusters to the same VLAN. A separate VLAN, and/or switch is needed for each cluster.

Connecting the secured interfaces (the synchronization interfaces) of multiple clusters is also not recommended for the same reason. Therefore, it is best to connect the secured interfaces of a given cluster via a crossover link when possible, or to an isolated VLAN.

If there is a need to connect the secured or the non-secured interfaces of multiple clusters to the same VLAN you need to make changes to:

- The destination MAC address, to enable communication between the cluster and members outside the cluster (for ClusterXL Load Sharing Multicast Mode clusters only).
- The source MAC address of the cluster, to enable Cluster Control Protocol communication between cluster members.

Changes to the Destination MAC Address

This section applies to ClusterXL Load Sharing Multicast Mode only.

How the Destination Cluster MAC Address is Assigned in Load Sharing Multicast Mode

When a member that is outside the cluster wishes to communicate with the cluster, it sends an ARP query with the cluster (virtual) IP address. The cluster replies to the ARP request with a multicast MAC address, even though the IP address is a unicast address.

This destination multicast MAC address of the cluster is based on the unicast IP address of the cluster. The upper three bytes are **01.00.5E**, and they identify a Multicast MAC in the standard way. The lower three bytes are the same as the lower three bytes of the IP address. An example MAC address based on the IP address **10.0.10.11** is shown below.

10. 0. 10. 11 Destination unicast IP address for the cluster

01	00	5E	00	0A	0B	Destination multicast MAC address for the cluster

Upper 3 bytes Identify a Multicast MAC	Lower 3 bytes From IP address

Duplicate Multicast MAC Addresses: The Problem

When more than one cluster is connected to the same VLAN, the last three bytes of the IP addresses of the cluster interfaces connected to the VLAN must be different. If they are the same, then communication from outside the cluster that is intended for one of the clusters will reach both clusters, which will cause communication problems.

For example, it is OK for the cluster interface of one of the clusters connected to the VLAN to have the address **10.0.10.11**, and the cluster interface of a second cluster to have the address **10.0.10.12**. However, the following addresses for the interfaces of the first and second clusters will cause complications: **10.0.10.11** and **20.0.10.11**.

Duplicate Multicast MAC Addresses: The Solution

The best solution is to change to the last three bytes of the IP address of all but one of the cluster interfaces that share the same last three bytes of their IP address.

If the IP address of the cluster interface cannot be changed, you must change the automatically assigned multicast MAC address of all but one of the clusters and replace it with a user-defined multicast MAC address. Proceed as follows:

1. In the **ClusterXL** page of the cluster object, select **Load Sharing >Multicast Mode**. In the **Topology** tab, edit the cluster interface that is connected to same VLAN as the other cluster.
2. In the **Interface Properties** window, **General** tab, click **Advanced**.
3. Change the default MAC address, and carefully type the new user defined MAC address. It must be of the form **01:00:5e:xy:yy:yy** where **x** is between 0 and 7 and **y** is between 0 and f(hex).

Changes to the Source MAC Address

This section applies to all ClusterXL modes, both High Availability and Load Sharing, and to OPSEC certified clustering products.

How the Source Cluster MAC Address is Assigned

Cluster members communicate with each other using the Cluster Control Protocol (CCP). CCP packets are distinguished from ordinary network traffic by giving CCP packets a unique source MAC address.

- The first four bytes of the source MAC address are all zero: **00.00.00.00**
- The fifth byte of the source MAC address is a magic number. Its value indicates its purpose

Default value of fifth byte	Purpose
0xfe	CCP traffic
0xfd	Forwarding layer traffic

- The sixth byte is the ID of the sending cluster member

Duplicate Source Cluster MAC Addresses: The Problem

When more than one cluster is connected to the same VLAN, if CCP and Forwarding Layer traffic uses Multicast MAC address for the destination, this traffic reaches only the intended cluster.

If the Broadcast MAC address is used for Destination for CCP and for Forwarding Layer traffic (and in certain other cases), cluster traffic intended for one cluster is seen by all connected clusters. If this traffic is processed by the wrong cluster, it will cause communication problems.

Duplicate Source Cluster MAC Addresses: the Solution

To resolve the issue, change the source MAC address (MAC magic id) of the cluster interfaces connected to the broadcast domain in all but one of the clusters.

MAC magic has two modes, manual and automatic. Automatic is the default mode and the recommended mode.

To change the MAC magic value:

1. Open the Check Point DataBase Tool **GuiDBedit** and connect to the Security Management Server.

2. Under the `Network_objects` table, select a cluster object and search for its `cluster_magic` field.
3. Set a MAC magic value to determine the mode:
   a) To work in *manual mode*, enter a value between 1 and 253.

      Enter a unique value for each cluster in the domain.

   b) To work in *automatic mode*, enter 254.

      254 is the default value and should be already set. If duplicate MAC addresses are occurring even though automatic mode is set, enter unique values for each cluster (manual mode).
4. Save and exit **GuiDBedit**.
5. Repeat steps 2-4 for each cluster.
6. Install policy on all clusters.
7. To verify the new values:
   a) Connect to each cluster member.
   b) Run `cphaprob mmagic`

   All members of the same cluster should have the MAC magic value.

To change the MAC magic id during a connectivity upgrade (R80.10 and higher):

Before the upgrade, find out the configuration mode.

1. Connect to one of the cluster members.
2. Run `cphaprob mmagic`

   The Configuration Mode field will show `manual` or `automatic`.

   If the configuration field is `automatic`, upgrade the cluster. The upgraded cluster member will learn the MAC magic value from a member that has not yet been upgraded (if a value exists). Select a value if no previous value exists.

   Note - If the configuration field is `manual`, and you want to continue to use manual configuration, the same MAC magic value must be reused.
3. Find the previous MAC magic value:
   - For **R80.10** and above, run: `cphaprob mmagic`
   - For **R77.30**, run `cphaconf cluster_id get`
   - For **R77.20** and below, run: `fwha_mac_magic`
4. If you are working in manual mode, connect to the management server using GuiDBedit.
5. Locate the cluster object under the `Network_objects` table.
   a) Search for the objects `cluster_magic` field.
   b) Enter the previous value.
6. Upgrade the cluster members.

   Note - When working in manual mode, the MAC magic value must be configured using **GuiDBedit** edit *before* first policy installation.
7. Install policy.
8. Run `cphaprob mmagic` to verify the MAC magic value.

To solve in R77.30:

1. In the First Time Configuration Wizard, set a value for **Cluster Global ID**.
2. See the cluster ID of each member: `cphaconf cluster_id get`
3. Change the ID in all members of each cluster (but leave one unchanged):
   `cphaconf cluster_id set` *<value>*
   For more details, see sk36055 http://supportcontent.checkpoint.com/solutions?id=sk36055.

To solve in R77.20 and below:

Use these Security Gateway configuration parameters to set more than one cluster on the same VLAN. These parameters apply to ClusterXL and OPSEC certified clustering products.

Parameter	Default value
fwha_mac_magic	**0xfe**
fwha_mac_forward_magic	**0xfd**

When you change the values of these parameters, you change the fifth part of the source MAC address of Cluster Control Protocol and forwarded packets. The two values must be different. To avoid confusion, do not use the value *0x00*.

The MAC magic parameters are explained in more detail later.

Monitoring the Interface Link State

Enabling Interface Link State Monitoring shortens the time it takes for ClusterXL to detect an interface failure. By monitoring the link state (i.e. the electrical state) of an interface, ClusterXL is immediately alerted to connectivity issues concerning a certain network interface, such as a disconnected cable, or an electrical failure (real or simulated) on a switch.

Interface Link State Monitoring requires an interface device driver that supports link state detection. The device driver reports the link state as either connected or disconnected.

Monitoring the interface link state is particularly useful in scenarios where a monitored interface (either a cluster interface or a monitored private interface) sends ICMP ECHO probe requests which are not answered by hosts or routers on the connected subnet.

When enabled, ClusterXL immediately detects when an interface goes down. When disabled, ClusterXL determines whether an interface is malfunctioning by watching subsecond timeout expiration.

Monitoring Interface Link State is enabled by default for R76 and higher.

Note - Interface Link State Monitoring requires an interface device driver that supports link state detection, and is supported for Gaia, SecurePlatform, and Linux only. This feature was tested on two cluster interfaces connected directly with a cross-cable. But it also works for interfaces connected to a switch. See sk31336 http://supportcontent.checkpoint.com/solutions?id=sk31336.

Enabling Interface Link State Monitoring

The global parameter `fwha_monitor_if_link_state` accepts these values:

- 0 – Disable Interface Link State Monitoring. This is the default setting for version R75.40VS and earlier.
- 1 – Enable Interface Link State Monitoring. This is the default setting for version R76 and higher.

To enable or disable Interface Link State Monitoring, run this command:

```
fw ctl set int fwha_monitor_if_link_state <0|1>
```

When you run this command, the change is not permanent. The Interface Link State Monitoring setting goes back to its default state when you reboot the member.

To enable or disable Interface Link State Monitoring permanently, see SecureKnowledge sk26202 http://supportcontent.checkpoint.com/solutions?id=sk26202.

Link Aggregation and Clusters

In This Section

Overview

Link Aggregation is a technique that bonds two or more network interfaces together on a Security Gateway. The interface bond gives High Availability redundancy in the event of interface failure and, in Load Sharing mode, can significantly increase throughput.

Note - Link Aggregation is supported on SecurePlatform, Gaia, and IPSO.

In an interface bond, between two and eight interfaces are set to act as a single interface, using the same IP address.

The bond is a virtual interface, defined on the OS, similar to a physical interface. Each physical interface in a bond is called a slave of that bond. Enslaved interfaces do not function independently of the bond.

Link Aggregation can be configured to one of two modes:

- **High Availability (Active/Backup) mode** - only one interface at a time is active. Upon interface failure, the bond fails over to another interface. Different interfaces of the bond can be connected to different switches, to add switch High Availability to interface High Availability.

  Note - Link-state initiated internal bond failover requires a network interface that supports the Media-Independent Interface (MII) standard.

- **Load Sharing (Active/Active) mode** - all interfaces are active, for different connections. Connections are balanced between interfaces according to network layers three and four, and follow either the IEEE 802.3ad standard or XOR. Load Sharing mode has the advantage of increasing throughput, but requires connecting all the interfaces of the bond to one switch.

For Link Aggregation High Availability mode and for Link Aggregation Load Sharing mode:

- The number of bond interfaces that can be defined is limited by the maximum number of interfaces supported by each platform. See the Release Notes for the appropriate Check Point release.
- Up to 8 NICs can be configured in a single High Availability or Load Sharing bond.

Link Aggregation - High Availability Mode

In This Section

When dealing with mission-critical applications, an enterprise requires its network to be highly available.

Clustering provides redundancy, and thus, High Availability, at the Security Gateway level. Without Link Aggregation, redundancy of Network Interface Cards (NICs) or of the switches on either side of the Security Gateway are only possible in a cluster, and only by failover of the Security Gateway to another cluster member.

Simple Redundant Topology

You can have redundancy of clustering without Link Aggregation. If a switch or member fails, a High Availability cluster solution provides system redundancy. For example, you can have a redundant system with two synchronized Security Gateway cluster members deployed in a redundant topology.

Item	Description
1	Security Gateway cluster member GW1 with interfaces connected to the external switches (items 5 and 6)
2	Security Gateway cluster member GW2 with interfaces connected to the external switches (items 5 and 6)

Item	Description
3	Interconnecting network C1
4	Interconnecting network C2
5	Switch S1
6	Switch S2

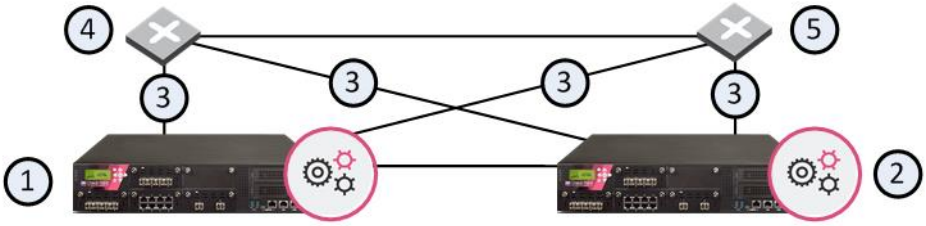
If GW1, its NIC, or S1 fails, GW2 becomes the only active member, connecting to switch S2 over network C2. If any component fails (Security Gateway, NIC, or switch), the result of the failover is that no further redundancy exists. A further failure of any active component completely stops network traffic.

Link Aggregation provides High Availability of NICs. If one fails, the other can function in its place. This functionality is in High Availability mode and in Load Sharing mode.

Fully Meshed Redundancy

The Link Aggregation High Availability mode, when deployed with ClusterXL, enables a higher level of reliability by providing granular redundancy in the network. This granular redundancy is achieved by using a fully meshed topology, which provides for independent backups for both NICs and switches.

A fully meshed topology further enhances the redundancy in the system by providing a backup to both the interface and the switch, essentially backing up the cable. Each cluster member has two external interfaces, one connected to each switch.



Item	Description
1	Security Gateway cluster member GW1 with interfaces connected to the external switches (items 4 and 5)
2	Security Gateway cluster member GW2 with interfaces connected to the external switches (items 4 and 5)
3	Interconnecting network
4	Switch S1
5	Switch S2

In this scenario:

- GW1 and GW2are cluster members in the High Availability mode, each connected to the two external switches
- S1 and S2 are switches

- Item 3 are the network connections

Bond Failover

In Link Aggregation High Availability mode, when the Security Gateway is part of a cluster, bond internal failover can occur in one of these cases:

- An active interface detects a failure in the link state, in a monitored interface.
- ClusterXL detects a failure in sending or receiving Cluster Control Protocol (CCP) keep-alive packets.

Either of these failures will induce a failover within the interface bond, or between cluster members, depending on the circumstances. The section below describes the two types of failover processes.

When a failure is detected, a log is recorded. You can see it in SmartView Tracker.

Creating an Interface Bond in High Availability Mode

Do these procedures to create an interface bond:

Removing IP Addresses from Slave Interfaces - SecurePlatform

Before you define an interface bond, make sure the slave (physical) interfaces do not have IP addresses:

1. Start the SecurePlatform configuration utility:
   `sysconfig`
2. Select `Network Connections`.
3. For each slave interface:
   a) Select `Configure connection`.
   b) Select the physical interface.
   c) Select `Remove IP from interface`.
   d) Return to Network Connections.
4. Exit the SecurePlatform configuration utility.

Setting Slave Interfaces as Disconnected

Disconnected interfaces are cluster member interfaces that are not monitored by the ClusterXL mechanism. If a disconnected interface fails, failover does not occur.

To define a slave interface as disconnected in SecurePlatform:

1. In $**FWDIR/conf/** create a file with this name: `discntd.if`
2. On separate lines in the file, enter the name of each physical interface that will function as a slave/bond pair.

Defining the Interface Bond

When the slave interfaces are without IP addresses, define the bond:

1. Start the SecurePlatform configuration utility:
   `sysconfig`
2. Select `Network Connections`.
3. Select `Add new connection`.
4. Select `Bond`.
5. For each interface to be enslaved under the bond, type its number in the list, and press Enter.
6. Enter `n` to go to the next step.
7. Select `High Availability`.
8. Choose whether to use default parameters (recommended) or to customize them.
9. Choose whether to set a primary slave interface, or not (recommended).
   A primary slave interface, after failing and coming back up, automatically returns to Active status, even if failover to the other interface occurred. If there is no primary interface, failover causes the other interface to become active and remain so until it fails.
10. Define the IP address and network mask of the new interface bond.
11. Exit the SecurePlatform configuration utility.

Verifying that the Bond is Functioning Properly

After installation or failover, it is recommended to verify that the bond is up, by displaying bond information.

1. Run:
   `cphaprob -a if`
   Make sure that the bond status is reported as `UP`.
2. Run:
   `cphaconf show_bond <bond name>`
   Check that the bond is correctly configured.

Failover Support for VLANs

In Link Aggregation High Availability mode, ClusterXL monitors VLAN IDs for connectivity failure or miscommunication, and initiate a failover when a failure is detected.

In a VLAN-enabled switched environment, ClusterXL monitors the VLAN with the lowest ID number. The monitoring is conducted by sending ClusterXL Control Protocol (CCP) packets on round-trip paths at a set interval. The lowest VLAN ID indicates the status of the physical connection. This VLAN ID is always monitored, and a connectivity failure causes ClusterXL to initiate a failover. ClusterXL will not detect a VLAN configuration problem on the switch.

Sync Redundancy

You use bond interfaces for synchronization interface redundancy on Gaia and SecurePlatform platforms. The use of more than one physical synchronization interface (**1st sync, 2nd sync, 3rd sync**) is not supported.

Requirements and Limitations:

- The bond interface for each member must connect to the same switch or VLAN.

- We recommend that interfaces and other network hardware support the IEEE 802.3 bond mode.
- If you use an HA bond, you must add slave interfaces in the same order for all members.

To configure bond interfaces for sync High Availability:

1. Define a bond interface ("Creating an Interface Bond in High Availability Mode" on page 101) on each member with unused slave interfaces.

   See the R80.10 Gaia Administration Guide http://supportcontent.checkpoint.com/documentation_download?ID=TBD for the procedures for defining bond interfaces on Gaia platforms.

   Make sure that the slave interfaces do not have IP addresses assigned to them.
2. In SmartConsole, use the **Get topology** feature to get the member IP addresses.
3. In the **GatewayCluster Properties** window, change the **Network Type** to **Sync**.
4. Install policy.
5. Run `cphaprob -a if` on all members to make sure that the sync interfaces are in the bond mode.

Link Aggregation - Load Sharing Mode

In This Section

In Load Sharing mode, Link Aggregation supplies Load Sharing, in addition to High Availability. All slave interfaces are active, and connections are balanced between the bond slave interfaces, similar to the way ClusterXL balances connections between cluster members.

In Load Sharing mode, each connection is assigned to a specific slave interface. For the individual connection, only one slave interface is active. On failure of that interface, the bond does failover of the connection to one of the other interfaces, which adds the failed interface connection to the connections it is already handling.

Connections are balanced between slave interfaces according to network layers three and four, and follow one of these standards:

- 802.3ad - includes LACP and is the recommended mode, but some switches may not support this mode.
- XOR.

In Load Sharing mode, all the interfaces of a bond must be connected to the same switch. The switch itself must support and be configured for Link Aggregation, by the same standard (802.3ad or XOR) as the Security Gateway bond.

Load Sharing needs Performance Pack to be running.

Workflow of Interface Bond in Load Sharing Mode

Creating a Load Sharing bond is similar to creating a High Availability bond. The procedures for removing IP addresses from slaves, disconnecting slave interfaces, and verifying the bond are the same.

To create a Load Sharing bond:

1. Make sure the switches are configured for the standard you are using (802.3ad or XOR).
2. Removing IP Addresses from Slave Interfaces ("Removing IP Addresses from Slave Interfaces - SecurePlatform" on page 101)
3. Setting Slave Interfaces as Disconnected (on page 101)
4. Defining Interface Bond in Load Sharing Mode
5. Setting Critical Required Interfaces
6. Verifying that the Bond is Functioning Properly (on page 102)

Defining Interface Bond in Load Sharing Mode

To define the interface bond:

1. Start the SecurePlatform configuration utility:
   `sysconfig`
2. Select `Network Connections`.
3. Select `Add new connection`.
4. Select `Bond`.
5. For each interface to be enslaved under the bond, type its number in the list, and press Enter.
6. Enter `n` to go to the next step.
7. Select `Load Sharing`.
8. Choose the Load Sharing standard: 802.3ad or XOR.
9. Choose whether to use default parameters (recommended) or to customize them.
10. Define the IP address and network mask of the new interface bond.
11. Exit the SecurePlatform configuration utility.

Setting Critical Required Interfaces

Note - The Critical Required Interfaces feature is supported for ClusterXL only.

A bond in Load Sharing mode is considered to be down when fewer than a critical minimum number of slave interfaces remain up. When not explicitly defined, the critical minimum number of interfaces in a bond of n interfaces is n-1. Failure of a second interface will cause the entire bond to be considered down, even if the bond contains more than two interfaces.

If a smaller number of interfaces will be able to handle the expected traffic, you can increase redundancy by explicitly defining the number of critical interfaces. Divide your maximal expected traffic speed by the speed of your interfaces and round up to a whole number to determine an appropriate number of critical interfaces.

To explicitly define the number of critical interfaces, create and edit the following file:

$FWDIR/conf/cpha_bond_ls_config.conf

Each line of the file should be of the following syntax:

<bondname> <critical#>

For example, if bond0 has seven interfaces and bond1 has six interfaces, file contents could be:

bond0 5

bond1 3

In this case bond0 would be considered down when three of its interfaces have failed. bond1 would be considered down when four of its interfaces have failed.

Configuring Cisco Switches for Load Sharing

These are sample configuration commands for Cisco switches.

For 802.3ad:

```
Switch#conf t
Switch(config)#port-channel load-balance src-dst-ip
Switch(config)#interface FastEthernet <all the participating interfaces>
Switch(config-if)#channel-group 1 mode active
Switch(config-if)#channel-protocol lacp
Switch(config-if)#exit
Switch(config)#interface port-channel 1
Switch(config-if)#switchport access vlan <the wanted vlan number>
Switch(config-if)#end
Switch#write
```

For XOR:

```
Switch#conf t
Switch(config)#port-channel load-balance src-dst-ip
Switch(config)#interface FastEthernet <all the participating interfaces>
Switch(config-if)#channel-group 1 mode on
Switch(config-if)#exit
Switch (config)#interface port-channel 1
Switch(config-if)#switchport access vlan <the wanted vlan number>
Switch(config-if)#end
Switch#write
```

Defining VLANs on an Interface Bond

VLANs can be defined on an interface bond in the same way as on a regular interface.

To define a VLAN on an interface bond:

1. Start the SecurePlatform configuration utility:
   `sysconfig`
2. Select `Network Connections`.
3. Select `Add new connection`.
4. Select `VLAN`.
5. Select the interface or interface bond on which to define the VLAN.
6. Enter a VLAN ID.
7. Define the IP addresses for the VLAN.
8. Exit the SecurePlatform configuration utility.

Performance Guidelines for Link Aggregation

To get the best performance, use static affinity for Link Aggregation.

Setting Affinities

If you are running Performance Pack in a multi-core system, after you define bonds, set affinities manually. Use the `-s` parameter of the `sim affinity` command.

Note - sim affinity commands take effect only if the Performance Pack is enabled and actually running. Performance Pack begins running when you install a Policy for the first time.

For optimal performance, set affinities according to the following guidelines:

1. Run `sim affinity` using the `-s` option.
2. Whenever possible, dedicate one processing core to each interface. See sk33520 http://supportcontent.checkpoint.com/solutions?id=sk33250.
3. If there are more interfaces than cores, one or more cores handle two interfaces. Use interface pairs of the same position with internal and external bonds.

   a) To view interface positions in a bond, run:

   `cat /proc/net/bonding/<bond name>.`

   b) Note the sequence of the interfaces in the output, and compare this for the two bonds (external bond and its respective internal bond). Interfaces that appear in the same position in the two bonds are interface pairs and set to be handled by one processing core.

   For example, you might have four processing cores (0-3) and six interfaces (0-5), distributed among two bonds:

bond0	bond1
eth0	eth3
eth1	eth4
eth2	eth5

Two of the cores will need to handle two interfaces each. An optimal configuration can be:

bond0		bond1	
eth0	core 0	eth3	core 0
eth1	core 1	eth4	core 1
eth2	core 2		
		eth5	core 3

ClusterXL Commands for Interface Bonds

cphaconf show_bond	**See status of one interface bond or summary of all bonds**

Syntax

```
cphaconf show_bond [<bond-name>|-a]
```

Options

Parameter	Description
`bond-name`	name of target bond
`-a`	show summary of all bonds

cphaconf show_bond	See status of one interface bond or summary of all bonds

Example

```
[Expert@GW-1]# cphaconf show_bond bond0

Bond name: bond0
Bond mode: Load Sharing
Bond status: Up
Balancing mode: 802.3ad Layer3+4 Load Balancing
Configured slave interfaces: 4
In use slave interfaces: 4
Required slave interfaces: 2
Slave Name | Status | Link
-------------------------------------------
eth2 | Active | Yes
eth3 | Active | Yes
eth4 | Active | Yes
eth5 | Active | Yes
```

Comments

The report results show:

- Required slave interfaces ("Setting Critical Required Interfaces" on page 104)
- Status value:
  - **Down -** (Load Sharing only) the physical link is down.
  - **Active** - currently handling traffic.
  - **Standby** - (High Availability only) the interface is ready and can support internal bond failover.
  - **Not Available** - (High Availability only) the physical link is broken, or the Cluster member is in status *down*. The bond cannot failover in this state.
- **Link** - if the physical link exists.

cphaconf failover_bond	Starts interface bond internal failover (High Availability only)

Syntax

```
cphaconf failover_bond <bond-name>
```

Parameters

Parameter	Description
bond-name	name of target bond

chaprob -a if	Displays status of all interface bonds and VLANs

Syntax

```
cphaprob -a if
```

Example

```
[Expert@GW-1]# cphaprob -a if
Required interfaces: 5
Required secured interfaces: 1
bond0 UP non sync(non secured), broadcast, bond, can failover
bond2 UP sync(secured), multicast, bond Load Sharing
bond1 UP non sync(non secured), multicast, bond Load Sharing
Virtual cluster interfaces: 4
bond0 192.168.34.60
bond1.60 10.34.60.1
bond1.61 10.34.61.1
bond1.62 10.34.62.1
```

chaprob -a if	Displays status of all interface bonds and VLANs
Comments	Use this command to see if a High Availability bond can failover.

Troubleshooting Bonded Interfaces

In This Section

Troubleshooting Workflow

1. Check the status of the bond ("Verifying that the Bond is Functioning Properly" on page 102).
2. If there is a problem, see if the physical link is down:
   a) Run:
      `cphaconf show_bond <bond-name>`
   b) Look for a slave interface that reports the status of the link as no.
   c) Check the cable connections and other hardware.
   d) Check the port configuration on the switch.
3. See if a cluster member is down:
   `cphaprob state`
   If any of the cluster members have a `firewall State` other than `active`, continue with the `cphaprob state` troubleshooting ("The cphaprob Command" on page 61).
4. View the logs in SmartView Tracker.

Connectivity Delays on Switches

When using certain switches, connectivity delays may occur during some internal bond failovers. With the various features that are now included on some switches, it can take close to a minute for a switch to begin servicing a newly connected interface. These are suggestions for reducing the startup time after link failure.

1. Disable auto-negotiation on the relevant interface.
2. On some Cisco switches, enable the PortFast feature.
3. Disable STP on the ports.

Warnings about PortFast

The PortFast feature should never be used on ports that connect to switches or hubs. It is important that the Spanning Tree complete the initialization procedure in these situations. Otherwise, these connections may cause physical loops where packets are continuously forwarded (or even multiply) in such a way that can cause the network to fail.

Sample Configuration of PortFast Feature on a Cisco Switch

The following are the commands necessary to enable PortFast on a Gigabit Ethernet 1/0/15 interface of a Cisco 3750 switch running IOS.

1. Enter configuration mode:
   `cisco-3750A#conf t`
2. Specify the interface to configure:
   `cisco-3750A(config)#interface gigabitethernet1/0/15`
3. Set PortFast on this interface:
   `cisco-3750A(config-if)#spanning-tree portfast`

Advanced Cluster Configuration

A number of synchronization and ClusterXL capabilities are controlled by means of Security Gateway configuration parameters. Run these commands on the Security Gateway as follows:

```
fw ctl set int Parameter <value>
```

Parameter is any of the parameters described in the following sections.

Changes to their default values must be implemented on all cluster members. Setting different values on cluster members can cause configuration problems and possibly connection failures.

All these configuration parameters can be configured to survive a boot. The way to do this varies with the operating system.

How to Configure Reboot Survival

Security Gateway configuration parameters that are changed using the **fw ctl set int** command do not survive reboot. The way to do make them survive a reboot varies with the operating system. In the following instructions, *Parameter* is any of the parameters described in the following sections.

Gaia

To add or change a parameter in the fwkern.conf file:

1. Open `$FWDIR/boot/modules/fwkern.conf` in a text editor.
2. Add or change the line: *Parameter* `=` *<value_in_hex>*.
3. Reboot the member.

Do these steps for each cluster member.

Windows

1. Edit the registry.
2. Add a DWORD value named *Parameter* under the key **HKEY_LOCAL_COMPUTER\SYSTEM\CurrentControlSet\Services\FW1\Parameters\Globals**.
3. Reboot.

Setting Module Variables in IPSO 6.1 and Later

When you install IPSO or run Voyager for the first time on a new platform, the Firewall Kernel Tuning Configuration page does not appear. If a customer service representative instructs you to use this page, you must first display it by performing these steps:

1. Establish a command line connection to the platform (using a network connection or console connection).
2. At the IPSO shell prompt, enter
   `# dbset advanced:loader t`
3. Run Voyager (or exit Voyager and run it again if Voyager was open when you entered the previous command).
4. Click **Configuration** > **Tools** > **Firewall Kernel Tuning** in the navigation tree.
5. Configure the variables as instructed by support and click **Apply**. Clicking Apply applies the firewall kernel variables and also saves the Voyager configuration so that the Firewall Kernel Tuning Configuration page will appear again if you reboot the platform.

Controlling the Clustering and Synchronization Timers

The following Security Gateway configuration parameters are used to control the clustering and synchronization timers. Changing the default values is not recommended.

Clustering and Synchronization timers

Parameter	Meaning	Default Value
`fwha_timer_cpha_res`	The frequency of ClusterXL operations on the cluster. Operations occur every: **10** multiplied by **fwha_timer_cpha_res** multiplied by **fwha_timer_base_res** milliseconds	1
`fwha_timer_sync_res`	The frequency of sync flush operations on the cluster. Operations occur every: **10** multiplied by **fwha_timer_sync_res** multiplied by **fwha_timer_base_res** milliseconds	**1**
`fwha_timer_base_res`	Must be divisible by 10 with no remainders.	10

Blocking New Connections Under Load

The reason for blocking new connections is that new connections are the main source of new synchronization traffic, and synchronization may be put at risk if new traffic continues to be processed at this rate.

A related error message is: **"FW-1: State synchronization is in risk. Please examine your synchronization network to avoid further problems!"** ("Sync Related Error Messages" on page 87).

Reducing the amount of traffic passing through the Security Gateway protects the synchronization mechanism. See sk43896 https://supportcontent.checkpoint.com/solutions?id=sk43896.

- **fw_sync_block_new_conns** allows Security Gateway to detect heavy loads and start blocking new connections. Load is considered heavy when the synchronization transmit queue of the firewall starts to fill beyond the **fw_sync_buffer_threshold**.
  - To enable blocking new connections under load, set to **0**.
  - To disable blocking new connections under load, set to **-1 (0xFFFFFFFF hex)** (default).

  Note that blocking new connections when sync is busy is only recommended for Load Sharing ClusterXL deployments. While it is possible to block new connections in High Availability mode, doing so does not solve inconsistencies in sync, as High Availability mode precludes that from happening. This parameter can be set to survive boot using the mechanism described in How to Configure a Security Gateway to Survive a Boot (see "How to Configure Reboot Survival" on page 109).
- **fw_sync_buffer_threshold is** the maximum percentage of the buffer that may be filled before new connections are blocked. By default it is set to 80, with a buffer size of 512. By default, if more than 410 consecutive packets are sent without getting an ACK on any one of them, new connections are dropped. When new connection blocking starts, **fw_sync_block_new_conns** is set to **0**. When the situation is stable, it is set back to **-1**.
- **fw_sync_allowed_protocols** is used to determine the type of connections that can be opened while the system is in a blocking state. Thus, the user can have better control over the system behavior in cases of unusual load. The **fw_sync_allowed_protocols** variable is a combination of flags, each specifying a different type of connection. The required value of the variable is the result of adding the separate values of these flags. For example, the default value of this variable is 24, which is the sum of **TCP_DATA_CONN_ALLOWED** (**8**) and **UDP_DATA_CONN_ALLOWED** (**16**), meaning that the default allows only TCP and UDP data connections to be opened under load.

ICMP_CONN_ALLOWED	1
TCP_CONN_ALLOWED	2 (except for data connections)
UDP_CONN_ALLOWED	4 (except for data connections)
TCP_DATA_CONN_ALLOWED	8 (the control connection should be established or allowed)
UDP_DATA_CONN_ALLOWED	16 (the control connection should be established or allowed)

Working with SmartView Tracker Active Mode

The **Active** mode in SmartView Tracker shows open connections through Security Gateways that send logs to the active log file on the Security Management Server. The Active mode can slow down synchronization because the synchronization mechanism randomly drops Active connection

updates. This issue generates SmartView Tracker error messages. For this reason, Check Point does not recommend using the Active mode view for a heavily loaded cluster.

The **fwlddist_buf_size** parameter controls the size of the synchronization buffer, as expressed in words (one word equals four Bytes). Words are used for synchronization and the SmartView Tracker Active mode. The default buffer size is 16k words. The maximum value is 64k words and the minimum value is 2k words.

You can change the **fwlddist_buf_size** parameter as necessary and the change is applied only after you restart the member. Make sure that that changed parameter is correct after you restart the member. See How to Configure Security Gateway Configuration Parameters for the procedures.

Reducing the Number of Pending Packets

ClusterXL prevents out-of-state packets in non-sticky connections. It does this by holding packets until a Sync ACK is received from all other active cluster members. If for some reason a Sync ACK is not received, the Security Gateway on the cluster member will not release the packet, and the connection will not be established.

To find out if held packets are not being released, run the **fw ctl pstat** command. If the output of the command shows that the **Number of Pending Packets** is large under normal loads (more than 100 pending packets), and this value does not decrease over time, use the **fwldbcast_pending_timeout** parameter to reduce the number of pending packets.

Change the value of **fwldbcast_pending_timeout** from the default value of 50 to a value lower than 50.

The value is in ticks units, where each tick is equal to 0.1 sec, so that 50 ticks is 5 seconds.

The value represents the time after which packets are released even if Sync ACKs are not received.

Configuring Full Synchronization Advanced Options

When a cluster member comes up after being rebooted (or after **cpstart**), it has to perform Full Synchronization. As a first step in the Full Synchronization process, it performs a handshake with one of the other active cluster members. Only if this handshake succeeds does the cluster member continue with the Full Synchronization process.

The extended handshake that takes place (by default) exchanges information between cluster members. This information includes version information, information about the installed Check Point products, and can include information about which the VPN kernel tables are currently active. The extended handshake is unrelated to the exchange of kernel table information that happens later in the Full Synchronization.

All cluster members must have the same Check Point products and versions installed. The extended handshake identifies when different products are installed on the cluster members. When different products are installed, a console warning and a log message are issued.

In order to support backward compatibility, it is possible to change the behavior of the extended handshake by means of the following Gateway Configuration Parameters. How to edit these parameters is explained in Advanced Cluster Configuration (on page 109):

- **fw_sync_no_ld_trans** has the default the value of **1**. Set to **0** in order to exchange kernel table information between members in the first phase of the Full Synchronization process.

- **fw_sync_no_conn_trans** has the default value of **0**. Set to **1** in order not to exchange installed product information between members in the first phase of the Full Synchronization process.
- **fw_sync_fcu_ver_check** has the default value of **1**. set to **0** to allow Full Connectivity Upgrade for versions that do not comply with requirements specified in the *R80.10 Installation and Upgrade Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD.

Defining Disconnected Interfaces

Disconnected interfaces are cluster member interfaces that are not monitored by the ClusterXL mechanism.

You may wish to define an interface as disconnected if the interface is down for a long time, and you wish the cluster member to continue to be active.

The processes listed below are equivalent to defining a non-monitored interface from the **Topology** page, with the exception that the GUI method works only for interfaces that have a defined IP address.

Defining a Disconnected Interface on Unix

Create a file under `$FWDIR/conf/discntd.if` and write the name of each interface that you do not want monitored by ClusterXL on a separate line.

Defining a Disconnected Interface on Windows

1. Open the **regedt32** registry editor. Do not use **regedit**.
2. Under **HKEY_LOCAL_COMPUTERS\System\CurrentControlSet\Services\CPHA** create a new value with the following characteristics:
   Value Name : DisconnectedInterfaces
   Data Type : REG_MULTI_SZ
3. Add the interface name. To obtain the interface system name run the command:
   fw getifs
4. Add this name to the list of disconnected interfaces using the following format:
   \device\<System Interface Name>
5. Run **cphastop** and then **cphastart** to apply the change.

Configuring Policy Update Timeout

When policy is installed on a Cluster, the cluster members undertake a negotiation process to make sure all of them have received the same policy before they actually apply it. This negotiation process has a timeout mechanism which makes sure a cluster member does not wait indefinitely for responses from other cluster members, which is useful in cases when another cluster member goes down when policy is being installed (for example).

In configurations on which policy installation takes a long time (usually caused by a policy with a large number of rules), a cluster with more than two members, and slow members, this timeout mechanism may expire prematurely.

It is possible to tune the timeout by setting the following parameter:

fwha_policy_update_timeout_factor.

The default value is 1 which should be sufficient for most configurations. For configurations where the situation described above occurs, setting this parameter to 2 should be sufficient. Do NOT set this parameter to a value larger than 3.

Enhanced 3-Way TCP Handshake Enforcement

The standard enforcement for a 3-way handshake that initiates a TCP connection provides adequate security by guaranteeing one-directional stickiness. This means that it ensures that the SYN-ACK will always arrive after the SYN. However, it does not guarantee that the ACK will always arrive after the SYN-ACK, or that the first data packet will arrive after the ACK.

If you wish to have stricter policy that denies all out-of-state packets, you can configure the synchronization mechanism so that all the TCP connection initiation packets arrive in the right sequence (SYN, SYN-ACK, ACK, followed by the data). The price for this extra security is a considerable delay in connection establishment.

To enable enhanced enforcement, use the Database Tool to change the **sync_tcp_handshake_mode** property from **minimal_sync** (default value) to **complete_sync**.

Cluster IP Addresses on Different Subnets

Introduction

You can configure cluster virtual IP addresses in different subnets than the members. The cluster virtual interfaces must have routable IP addresses that connect to internal and external networks.

The network "sees" the cluster as one Security Gateway that operates as a network router. The network is not aware of the internal cluster structure and member addresses.

Advantages of using different subnets:

- Lets you create a cluster in an existing subnet that has a shortage of available IP addresses.
- You use only one routable, virtual IP address for the cluster. All other IP addresses can be on other subnets.
- Lets you 'hide' cluster physical cluster addresses behind the virtual cluster IP address. This security practice is almost the same as NAT.

  Note - This capability is available only for ClusterXL Clusters. For details about OPSEC certified clusters, see the vendor documentation.

Traffic sent from cluster members to internal or external networks is hidden behind the cluster virtual IP addresses and MAC addresses. The MAC address assigned to cluster interfaces is the:

- MAC address of the active member interface, in the new High Availability mode.
- Multicast MAC, in the Load Sharing Multicast mode.
- Pivot member MAC in the Load Sharing Unicast mode.

The use of different subnets with cluster objects has some limitations (see "Limitations of Cluster Addresses on Different Subnets" on page 117).

Configuring Cluster Addresses on Different Subnets

These are the steps necessary to configure a cluster with different subnets:

1. For each member, define a static route from the member interface to the cluster virtual interface.

   You can do this with operating system commands or with the Check Point `cpconfig` utility,
2. Configure the cluster topology ("Working with Cluster Topology" on page 49) so that each member has one interface that connects to each cluster virtual IP address.

   Usually, cluster virtual IP addresses are automatically related to an interface based on membership in the same subnet. When the subnets are different, you must explicitly define the relationship between a member interface and a cluster virtual IP address.

Defining the Member Network.

When using a cluster with the cluster virtual IP and members on different subnets, it is necessary to manually define the member.

To manually define the member network.

1. In SmartConsole, use the Classic Mode ("Manual Configuration" on page 48) to manually create a new cluster.
2. Define the cluster members and their physical interfaces.
3. Go to the **Topology** page.
4. Click **Edit**.
5. In the **Edit Topology** window, enter the IP address for each virtual cluster interface.
6. Save the database.
7. Install policy.

For more details, see the Configuring Cluster Objects ("Configuring the Cluster Object and Members" on page 46) chapter.

Configuring a Static Route - Gaia

Use this procedure to configure a static route on all Gaia members. If you do not define the static routes correctly, the member interface IP address is not routable.

Note - It is not necessary to configure on a Gaia Security Gateway in the VSX mode. This is done automatically when you configure routes in SmartConsole.

To configure a static route on a member - clish:

1. Run `set static-route <VIP-subnet/mask> nexthop gateway logical <interface> on`.

   <**VIP-subnet**> - Cluster Virtual IP address and subnet mask for the cluster interface.

   <i**nterface**> - Member interface name.
2. Run `set static-route <VIP-subnet> scopelocal on`.

   <**VIP-subnet**> - Subnet virtual IP address for the cluster interface.
3. To make sure that the `scopelocal` attribute is set correctly, run:

   `cat /etc/routed.conf`

Sample output:

```
static {
10.16.6.0 masklen 24 gateway eth1 scopelocal;
  default gateway 192.168.2.11;
};
```

Important - For R75.40, R75.40VS, and R75.45, you must download and install a Gaia Hotfix. This Hotfix includes a new configuration attribute '`scopelocal`' for static routes and a Gaia command to set this attribute. See sk92799 https://supportcenter.checkpoint.com/supportcenter/portal?eventSubmit_doGoviewsolutiondetails=&solutionid=sk92799.

To configure a static route on a member - WebUI:

1. In the WebUI navigation tree, select **IPv4 Static Routes**.
2. In the **IPv4 Static Routes** pane, click **Add**
   or
   Select a route and click **Edit** to change an existing route.
3. In the **Add** (or **Edit**) **Destination Route** window, enter the
   - **Destination -** Cluster Virtual IP address.
   - **Subnet mask** - for the cluster interface.
   - **Next Hop Type -** Select **Normal**. This accepts and sends packets to the specified destination.
4. Click **Add Gateway**.
5. Select **Network Interface.**
6. Select a **Logical Interface**. This identifies the next hop gateway by the cluster member interface that connects to it. Use this option only if the next hop gateway has an unnumbered interface.
7. Click **OK**.
8. Select **Local Scope**. This lets the cluster member accept static routes on the subnet of the Cluster Virtual address.
9. Click **Save**.

Configuring a Static Route - SecurePlatform

Use this procedure to configure a static route on all SecurePlatform members. If you do not define the static routes correctly, the member interface IP address is not routable.

Note - It is not necessary to configure static routes the Advanced Dynamic Routing Suite is installed on your members. Advanced Dynamic Routing Suite automatically adds the static routes to the cluster network.

To configure a static route on a member:

1. Go to the expert mode.
2. Run `sysconfig`.
3. Select **Routing** > **Add New Network Route**.
4. When prompted for the **network IP address**, enter the cluster virtual IP address and net mask.
5. When prompted for the **gateway IP address**, press **Enter** to accept the default local address.
6. When prompted for the **outgoing interface**, enter the member interface name.

7. Run this command to make sure that the `scopelocal` attribute is set correctly:

   `cat /etc/routed.conf`

   Sample output:

```
static {
10.16.6.0 masklen 24 gateway eth1 scopelocal;
  default gateway 192.168.2.11;
};
```

Configuring a Static Route - Other Operating Systems.

To configure a static route in other operating systems, refer to the documentation for those systems. For more assistance, contact Check Point Support (https://supportcenter.checkpoint.com).

Limitations of Cluster Addresses on Different Subnets

This new feature does not yet support all the capabilities of ClusterXL. Some features require additional configuration to work properly, while others are not supported.

Connectivity Between Cluster Members

Since ARP requests issued by cluster members are hidden behind the cluster IP and MAC, requests sent by one cluster member to the other may be ignored by the destination computer. To allow cluster members to communicate with each other, a static ARP should be configured for each cluster member, stating the MAC addresses of all other members in the cluster. IP packets sent between members are not altered, and therefore no changes should be made to the routing table.

Note - Static ARP is not required in order for the members to work properly as a cluster, since the cluster synchronization protocol does not rely on ARP.

Load Sharing Multicast Mode with "Semi-Supporting" Hardware

Although not all types of network hardware work with multicast MAC addresses, some routers can pass such packets, even though they are unable to handle ARP replies containing a multicast MAC address. Where a router *semi-supports* Load Sharing Multicast mode, it is possible to configure the cluster MAC as a static ARP entry in the router internal tables, and thus allow it to communicate with the cluster.

When different subnets are used for the cluster IPs, static ARP entries containing the router MAC need to be configured on each of the cluster members. This is done because this kind of router will not respond to ARP requests containing a multicast source MAC. These special procedures are not required when using routers that fully support multicast MAC addresses.

Manual Proxy ARP

When using static NAT, the cluster can be configured to automatically recognize the hosts hidden behind it, and issue ARP replies with the cluster MAC address, on their behalf. This process is known as *Automatic Proxy ARP*.

However, if you use the ClusterXL VMAC mode or different subnets for the cluster IP addresses, this mechanism will not work, and you must configure the proxy ARP manually. To do so, in SmartConsole, select **Policy menu > Global Properties > NAT Network Address Translation**, and

disable **Automatic ARP Configuration**. Then create a file called **local.arp** in the firewall configuration directory (**$FWDIR/conf**).

Each entry in this file is a triplet, containing the:

- host address to be published
- MAC address that needs to be associated with the IP address
- unique IP of the interface that responds to the ARP request.

The MAC address that should be used is the cluster multicast MAC defined on the responding interface, when using multicast LS, or this interface unique IP, for all other modes.

Connecting to the Cluster Members from the Cluster Network

Since the unique IPs may be chosen arbitrarily, there is no guarantee that these addresses are accessible from the subnet of the cluster IP. In order to access the members through their unique IPs, you must configure routes on the accessing member, such that the cluster IP is the Security Gateway for the subnet of the unique IPs.

Default Security Gateway on SecurePlatform

Run **sysconfig > routing > add network route > add the routable network with its subnet**, and choose the correct physical interface in this direction.

Now go to **routing > add default Security Gateway** and add the IP address of the default (routable) Security Gateway. This will usually be the IP address of the router in one of the cluster IP subnet.

If you have the *different subnets* feature configured on more than one interface, repeat the addition of the network address (as above) for all these interfaces. (It is NOT required to define a default Security Gateway for the other subnets as well.)

Anti-Spoofing

When the *different subnets* feature is defined on a non-external interface, the cluster IP in the **Cluster Topology** tab should not be defined with the **Network defined by interface IP and Net Mask** definition in the **Topology** tab of the **Interface Properties** window of the cluster interface. You must add a group of networks that contain both the routable network and the non-routable network, and define the Anti-Spoofing for this interface as **specific:** network with this new group.

Converting a Security Gateway to a ClusterXL Cluster

This section tells you how convert a Security Gateway to a ClusterXL cluster. The source Security Gateway becomes one of the members and you add one or more new members to the cluster. To help you identify the members of the new ClusterXL cluster, the procedures use these names:

- **Standalone Computer** - An existing computer that is configured as both a Security Gateway and a Security Management Server.
- **Source Security Gateway -** The physical Security Gateway that will be converted into a member of the new ClusterXL cluster.
- **Source Member** - The member created from the **Source Security Gateway**.
- **New Member** - A newly created cluster member. There can be more than one new member.

You must have sufficient available IP address for the source Security Gateway and new members. If not, see Configuring Cluster Addresses on Different Subnets.

Converting a Standalone Deployment to ClusterXL.

Before you can convert a Standalone Deployment to ClusterXL, you must first migrate the Security Gateway and the Security Management Server to two different computers. We recommend that you keep the existing Standalone Computer available until you complete and test the new ClusterXL environment.

Notes and Cautions:

- We recommend that you do the conversion during scheduled maintenance downtime.
- The new Security Management Server and Security Gateway must have the same version as the existing Standalone Computer. Do version upgrades and/or install hotfixes before you start the conversion process.
- We recommend that you keep the same interface IP addresses for your internal and external networks on the new Security Gateway. This can minimize the necessity to reconfigure gateway topologies.
- We recommend that you run `cpinfo -z -o Standalone.cpinfo` in the Expert Mode on the Standalone Computer and the new Security Management Server. This gives you "before" and "after" status and debugging information that can help resolve migration issues.

  Copy these files to another computer or external storage.
- For more detailed information about these procedures, see sk61681 http://supportcontent.checkpoint.com/solutions?id=sk61681.

To prepare the Standalone Computer for migration:

1. Backup the Standalone Computer. Use one of the procedures included in the *Backing Up* section of the *R80.10 Installation and Upgrade Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD. Copy the backup file to another computer or external storage.
2. Disconnect the Standalone Computer from the network.
3. Disable all Security Gateway functionality:
   a) Connect with SmartConsole and open the Standalone Computer object.
   b) On the **General Properties** > **Network Properties** tab, clear all Software Blades including Firewall. Click **OK** to continue.
   c) Save the changes (**Menu** > **File** > **Save**).
   d) Go to **Menu** > **Policy** > **Install Database**.
   e) In the **Install Database** window, select the Standalone Computer object and click **OK**.

      This operation must complete successfully.
   f) Close SmartConsole and all other SmartConsole clients.

To Export the Management Database:

1. Connect with the CLI to the Standalone Computer in the Expert mode.

2. Export the management databases:
   - On Gaia, SecurePlatform, Linux and IPSO, run:

     ```
     # cd $FWDIR/bin/upgrade_tools/
     # ./upgrade_export /var/<export_file_name>
     ```

   - On Windows, run:

     ```
     cd /d "%FWDIR%\bin\upgrade_tools\"
     upgrade_export  C:\<export_file_name>
     ```

To Create the new Security Management Server:

Important - The new Security Management Server must have the same host name as the existing Standalone Computer.

1. Do a clean Security Management Server installation based on the procedures in the *R80.10 Installation and Upgrade Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD. Make sure that you only select **Management Server** options.

   Make sure that you install all Hotfixes and plug-ins that were installed in the existing Standalone computer.
2. On Gaia, SecurePlatform, Linux and IPSO, close all of the Expert mode shells. Log into the regular shell.
3. Copy the exported database files to a temporary folder on the new Security Management Server.
4. Import the management databases:
   - From the Expert Mode on Gaia, SecurePlatform, Linux and IPSO, run:

     ```
     # cd $FWDIR/bin/upgrade_tools/
     # ./upgrade_import  /<path_to>/<export_file_name>
     ```

   - On Windows, run:

     ```
     cd /d "%FWDIR%\bin\upgrade_tools\"
     upgrade_import  C:\<path_to>\<export_file_name>
     ```

   Important - If the import fails with the **Database migration between standalone and management only machines is not supported** error, see sk61681 http://supportcontent.checkpoint.com/solutions?id=sk61681 for a workaround.
5. Connect with SmartConsole to the new Security Management Server and make sure that all settings are correct.
6. Close SmartConsole and reboot the computer.

To Create the New Security Gateway:

1. Do a clean Security Gateway installation based on the procedures in the *R80.10 Installation and Upgrade Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD. Make sure that you only select **Network Security** tab options.

   Make sure that you install all Hotfixes and plug-ins that were installed in the existing Standalone computer.
2. In SmartConsole, create and configure the Security Gateway object.

   Make sure that you establish SIC trust.
3. Open SmartConsole and install policy to this gateway.

4. Connect the systems to the network.
5. Thoroughly test and debug the deployment.
   Make sure that the rules for all Software Blades work correctly.

This Security Gateway will become the **Source Member** for the new ClusterXL cluster.

Creating the New Member

To create and configure a new cluster member:

1. Install a new Security Gateway.
2. Use the standard procedure to create a new cluster member ("Creating the ClusterXL Object" on page 121).
3. Make sure that the cluster object definition and all applicable settings are the same as for the **Source Security Gateway**. For example:
   - Interface, topology and Anti-Spoofing definitions
   - Authentication types
   - IPsec VPN settings, including Link Selection
   - Office mode settings
   - Firewall rules settings
   - Software Blade selections and configuration

Creating the ClusterXL Object

To create the ClusterXL object:

1. In SmartConsole, create a new cluster object ("Configuring the Cluster Object and Members" on page 46).
2. Make sure that the cluster object definition and all applicable settings are the same as for the **Source Security Gateway**. For example:
   - Interface, topology and Anti-Spoofing definitions
   - Authentication types
   - IPsec VPN settings, including Link Selection
   - Office mode settings
   - Firewall rules settings
   - Software Blade selections and configuration
3. If you assign Office Mode IP address from a pool, create a new pool

In SmartConsole, for Computer 'B'

1. Create a ClusterXL object.
2. In the **Cluster Members** page, click **Add**, and select **New Cluster Member**.
3. Connect to computer 'B', and define its topology.
4. Define the Synchronization networks for the cluster.
5. Define the cluster topology. To avoid reconfiguring network devices, the cluster IP addresses should be the same as the addresses of computer 'A', on its proposed cluster interfaces.
6. Install the policy on the cluster, currently including member 'B' only.

On Computer 'A'

1. Disconnect all proposed cluster and Synchronization interfaces. New connections now open through the cluster, instead of through computer 'A'.
2. Change the addresses of these interfaces to some other unique IP address which is on the same subnet as computer B.
3. Connect each pair of interfaces of the same subnet using a dedicated network. Any hosts or Security Gateways previously connected to the Security Gateway must now be connected to both members, using a hub/switch.

Note - It is possible to run synchronization across a WAN. For details, see Synchronizing Clusters over a Wide Area Network (see "Synchronizing Clusters on a Wide Area Network" on page 24).

In SmartConsole for Computer 'A'

1. Update the topology of Security Gateway A, either manually or by clicking **Get Topology**.

   If the IP address of the management interface was changed, the **Get Topology** action will fail. If this happens, manually change the main IP address in the Security Gateway object and save the policy prior to performing an automatic topology fetch.
2. In the **Cluster Members** page, click **Add**, and select **Add Security Gateway to Cluster**.
3. Select computer 'A' in the window.
4. In the **Edit Topology** page, determine which interface is a cluster interface, and which is an internal or an external interface.
5. Install the policy on the cluster.

Adding Another Member to an Existing Cluster

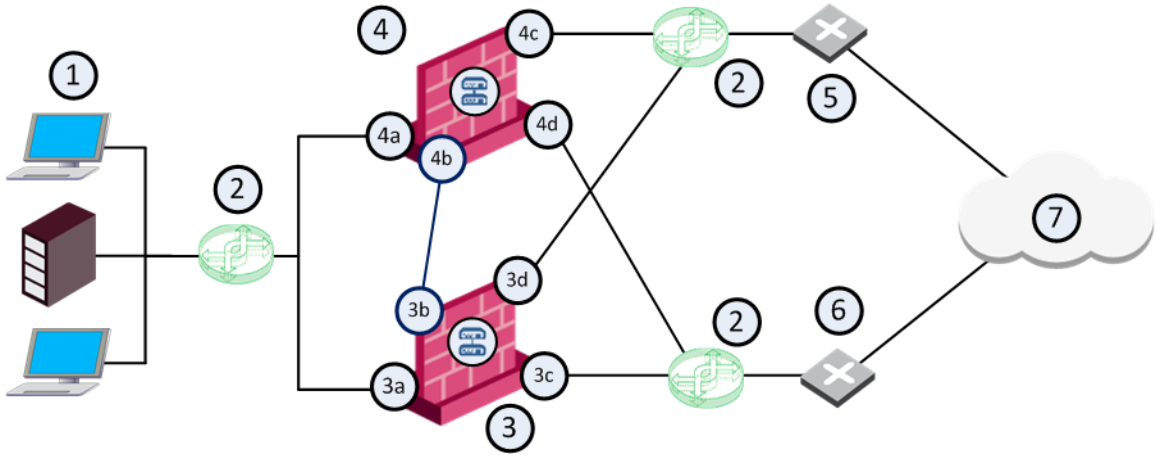
1. On the cluster member, run **cpconfig** to enable ClusterXL.
2. Change the IP addresses of the new cluster member to reflect the correct topology (either shared IP addresses or unique IP addresses, depending on the clustering solution).
3. Ensure that all required Check Point products are installed on the new cluster member.
4. In the **Cluster Members** page of the Cluster object, either create a new cluster member (if it is a new Security Gateway computer) with the appropriate properties, or convert an existing Security Gateway to a cluster member.
5. If this is a new Security Gateway computer, ensure that SIC is initialized. In the **Edit Topology** page, ensure that the topology is correctly defined.
6. If the **Cluster Mode** is **Load Sharing** or **New HA**, ensure that the proper interfaces on the new cluster member are configured as **Cluster Interfaces**.
7. Install the security policy on the cluster.
8. The new member is now part of the cluster.

Configuring ISP Redundancy on a Cluster

Make Internet connectivity more reliable with ISP Redundancy. This connects a Security Gateway or cluster member to the Internet through redundant Internet Service Provider (ISP) links. ISP Redundancy monitors the ISP links and chooses the best current link.

R80.10 supports two ISPs.

If you have a ClusterXL Cluster, connect each cluster member to the two ISPs through a LAN with two interfaces. The member interfaces must be on the same subnet as the cluster external interfaces. Configure ClusterXL in the usual way.



- ISP A link address - 192.168.1.100
- ISP B link address - 172.16.2.100

Item	Description
1	Internal network
2	VLAN switches
3	Security Gateway - Cluster member A
3a	Virtual interface to the internal network (10.10.0.1)
3b	Interface to the Cluster Sync network (10.0.10.1)
3c	Virtual interface to **ISP B** (172.16.2.2)
3d	Virtual interface to VLAN switch that connects to **ISP A** (192.168.1.3)
4	Security Gateway - Cluster member B
4a	Virtual interface to the internal network (10.10.0.2)
4b	Interface to the Cluster Sync network (10.0.10.2)
4c	Virtual interface to **ISP A** (192.168.1.2)
4d	Virtual interface to VLAN switch that connects to **ISP B** (172.16.2.3)
5	Router that connects to **ISP A** (192.168.1.1)
6	Router that connects to **ISP B** (172.16.2.1)
7	Internet

You can configure the ISP preference to be for Load Sharing or Primary/Backup.

Load Sharing - Uses the two links with a distributed load of connections going out from the Security Gateway. Connections coming in are alternated. You can configure best relative loads for the links (set a faster link to handle more load). New connections are randomly assigned to a link. If one link fails, the other takes the load.

Primary/Backup - Uses one link for connections going out from the Security Gateway and coming in. It switches to the backup if the primary link fails. When the primary link is restored, new connections are assigned to it. Existing connections continue on the backup link until they are complete.

Note: ISP Redundancy settings override VPN Link Selection settings.

To enable ISP Redundancy:

1. Open the network object properties of the Security Gateway or cluster.
2. Click **Other** > **ISP Redundancy**.
3. Select **Support ISP Redundancy**.
4. Select **Load Sharing** or **Primary/Backup**.
5. Configure the links ("Configuring the ISP Links" on page 124).
6. Configure the Security Gateway to be the DNS server ("Configuring Security Gateway as DNS" on page 125).
7. Configure the policy for ISP Redundancy ("Configuring the Firewall" on page 126).

Configuring the ISP Links

Before you begin, make sure you have the ISP data - the speed of the link and next hop IP address. If the Security Gateway has only one external interface, configure two subnets on this interface. You will need routers and a switch.

If the Security Gateway has two external interfaces in the **Network Management** page of the gateway object, you can configure the links automatically.

If the gateway is a ClusterXL cluster member, configure the two cluster members to the two ISP. Use a LAN with two interfaces. Make sure the member interfaces are on the same subnet as the cluster external interfaces.

To configure ISP links automatically:

1. In the Security Gateway object go to the **Other** > **ISP Redundancy** page.
2. Click **Set initial configuration**.
   The ISP Links are added automatically.
3. For **Primary/Backup**, make sure the primary interface is first in the list. Use the arrows to change the order.

To configure ISP links manually:

1. In the Security Gateway object go to the **Other** > **ISP Redundancy** page.
2. Click **Add**.
3. In the ISP Link window, give the link a **Name**.
   Note the names you give here. They are used in the ISP Redundancy script and commands.
4. Select the **Interface** of the Security Gateway for this ISP link.

- If the Security Gateway has two external interfaces, set each link to a different interface. If one of the ISP links is dialup connection to a backup ISP, configure the ISP Redundancy Script ("Editing the ISP Redundancy Script" on page 127).
- If the Security Gateway has only one external interface, set each ISP link to connect to this interface.

5. Configure the **Next hop IP Address**.
   - If the Security Gateway has two external interfaces, leave this field empty and click **Get from routing table**. The next hop is the default gateway.
   - If the Security Gateway has one external interface, set each ISP link to a different next hop router.
6. For Load Sharing, enter the **Weight**. For equal weight distribution, enter **50**. If one link is faster, raise this value and lower it for the other link, so that the two equal 100.
7. Define hosts to be monitored, to make sure the link is working. Open the **Advanced** tab of the **ISP Link** window, and add **Selected hosts**.

Configuring Security Gateway as DNS

The Security Gateway, or a DNS server behind it, must respond to DNS queries. It resolves IP addresses of servers in the DMZ (or another internal network).

Get a routable IP address from each ISP. If routable IP addresses are not available, register the domain to make the DNS server accessible from the Internet.

To enable DNS on the Security Gateway:

1. In the Security Gateway object **ISP Redundancy** page, select **Enable DNS Proxy**.

   The gateway intercepts Type A DNS queries for the web servers in its domain, that come from external hosts. If the Security Gateway recognizes the external host, it replies:
   - In Load Sharing mode, the Security Gateway replies with two addresses, alternating their order.
   - In Primary/Backup mode, the Security Gateway replies with the addresses of the active link.

   If the Security Gateway does not recognize the host, it passes the DNS query on to the original destination or to the domain DNS server.
2. Click **Configure**.
3. Add your DMZ or web servers. Give each two routable IP addresses, one for each ISP.
4. Enter a number of seconds in **DNS TTL**.

   This sets a Time To Live for each DNS reply. DNS servers in the Internet cannot cache your DNS data in the reply for longer than the TTL.
5. Configure Static NAT to translate the routable addresses to the real server address. External clients use one of the two addresses.

   Note - If the servers use different services (for example, HTTP and FTP), you can use NAT for only two routable IP addresses.
6. Define an Access Control Policy rule: **allow** DNS traffic through the Security Gateway using the **domain_udp** service.

To register the domain and get IP addresses:

1. Register your domain with the two ISP.
2. Tell the ISP the two addresses of the DNS server that respond to DNS queries for the domain.

3. For each server in the DMZ, get two routable IP addresses, one from each ISP.
4. In SmartConsole, click **Menu** > **Global Properties** > **NAT** and select **Manual NAT rules** - **Translate destination on client side**.

Configuring the Firewall

The Firewall must allow connections through the ISP links, with Automatic Hide NAT on network objects that start outgoing connections.

To configure the firewall for ISP Redundancy:

1. In the properties of the object for an internal network, select **NAT** > **Add Automatic Address Translation Rules**.
2. Select **Hide behind the gateway**.
3. Click **OK**.
4. Define rules for publicly reachable servers (web servers, DNS servers, DMZ servers).

   If you have one routable IP address from each ISP for the Security Gateway, define Static NAT. Allow specific services for specific servers. For example, make NAT rules so that incoming HTTP connections from the two ISP reach a Web server, and DNS traffic from the ISP reach the DNS server.

Example: Manual Static Rules for a Web Server and a DNS Server

Original Source	Original Destination	Original Service	Original Source	Translated Destination	Translated Services	
Any	IP of web server	http	=	10.0.0.2 (Static)	=	Incoming Web - ISP A
Any	IP of web server	http	=	10.0.0.2 (Static)	=	Incoming Web - ISP B
Any	IP of DNS server	domain_udp	=	10.0.0.3 (Static)	=	Incoming DNS - ISP A
Any	IP of DNS server	domain_udp	=	10.0.0.3 (Static)	=	Incoming DNS - ISP B

   If you have a routable address from each ISP for each publicly reachable server (in addition to the Security Gateway), define NAT rules:

   a) Give each server a non-routable address.

   b) Use the routable addresses in the **Original Destination**.

   c) Use the non-routable address in the **Translated Destination**.

   d) Select **Any** as the **Original Service**.

   Note - If using Manual NAT, automatic arp does not work for the NATed addresses. On Linux use local.arp.

When done, install the Access Control policy.

Configuring with VPN

When ISP Redundancy is enabled, VPN encrypted connections survive a failure of an ISP link. The settings in the ISP Redundancy page override settings in the Link Selection page.

To configure ISP Redundancy with VPN on one Security Gateway:

1. In **Topology** > **ISP Redundancy**, select **Apply settings to VPN traffic**.
2. In **IPsec VPN** > **Link Selection**, see that **Use ongoing probing** shows the mode of the ISP Redundancy: **Load Sharing** or **High Availability** (for Primary/Backup).

   Link Selection now only probes the ISP configured in ISP Redundancy.

To configure for VPN with a third-party peer:

If the Security Gateway peer is not a Check Point computer or appliance, the VPN may fail, or the third-party device may continue to encrypt traffic to a failed link.

- Make sure the device recognizes encrypted traffic from the secondary link as coming from the gateway.
- Change the configuration of ISP Redundancy to not use these Check Point technologies:
  - **Use Probing** - Make sure that **Link Selection** uses another option.
  - **Load Sharing, Service Based Link Selection, Route based probing** - Work only on Check Point Security Gateways. If used, the Security Gateway uses one link to connect to the third-party peer. The link with the highest prefix length and lowest metric is used.

Force ISP Link State

Use the **fw isp_link** command to force the ISP link state to Up or Down. Use this to test installation and deployment, or to force the Security Gateway to recognize the true link state if it cannot (the ISP link is down but the gateway sees it as up).

You can run this command on the Security Gateway or the Security Management Server: `fw isp_link [target-gw] <link_name> {up|down}`

`<link_name>` is the name in the ISP Link window.

Editing the ISP Redundancy Script

When the Security Gateway starts, or an ISP link state changes, the `$FWDIR/bin/cpisp_update` script runs. It changes the default route of the Security Gateway. For example, you can force the Security Gateway to change the state of a dialup interface to match that state of its ISP link.

Edit this script to enable a dialup connection for one of the ISP links.

To configure a dialup connection:

1. In the script on the Security Gateway, enter the command to change the dialup interface state:
   - If the link goes down: `fw isp_link <link_name> down`
   - If the link goes up: `fw isp_link <link_name> up`
2. If you use PPPoE or PPTP xDSL modems, in the PPPoE or PPTP configuration of SecurePlatform, the **Use Peer Gateway** option must not be selected.

Enabling Dynamic Routing Protocols in a Cluster Deployment

ClusterXL supports Dynamic Routing (Unicast and Multicast) protocols as an integral part of SecurePlatform. As the network infrastructure views the clustered Security Gateway as a single logical entity, failure of a cluster member will be transparent to the network infrastructure and will not result in a ripple effect.

Components of the System

Virtual IP Integration

All cluster members use the cluster IP address(es).

Routing Table Synchronization

Routing information is synchronized among the cluster members using the Forwarding Information Base (FIB) Manager process. This is done to prevent traffic interruption in case of failover, and used for Load Sharing and High Availability modes. The FIB Manager is the responsible for the routing information.

The FIB Manager is registered as a critical device (Pnote), and if the slave goes out of sync, a Pnote will be issued, and the slave member will go down until the FIB Manager is synchronized.

Failure Recovery

Dynamic Routing on ClusterXL avoids creating a ripple effect upon failover by informing the neighboring routers that the router has exited a maintenance mode. The neighboring routers then reestablish their relationships to the cluster, without informing the other routers in the network. These restart protocols are widely adopted by all major networking vendors. The following table lists the RFC and drafts compliant with Check Point Dynamic Routing:

Protocol	RFC or Draft
OSPF LLS	draft-ietf-ospf-lls-00
OSPF Graceful restart	RFC 3623
BGP Graceful restart	draft-ietf-idr-restart-08

Dynamic Routing in ClusterXL

The components listed above function "behind-the-scenes." When configuring Dynamic Routing on ClusterXL, the routing protocols automatically relate to the cluster as they would to a single device.

When configuring the routing protocols on each cluster member, each member is defined identically, and uses the cluster IP address(es) (not the member physical IP address). In the case of OSPF, the router ID must be defined and identical on each cluster member. When configuring OSPF restart, you must define the restart type as **signaled** or **graceful**. For Cisco devices, use type **signaled**.

Use the SecurePlatform command line to configure each cluster member.

```
--------- Launch the Dynamic Routing Module
[Expert@GWa]# router
localhost>enable
localhost#configure terminal
--------- Enable OSPF and provide an OSPF router ID
localhost(config)#router ospf 1
localhost(config-router-ospf)#router-id 192.168.116.10
localhost(config-router-ospf)#restart-type [graceful | signaled]
localhost(config-router-ospf)#redistribute kernel
--------- Define interfaces/IP addresses on which OSPF runs (Use the cluster IP
address as defined in topology) and the area ID for the interface/IP address
localhost(config-router-ospf)#network 1.1.10.10 0.0.0.0 area 0.0.0.0
localhost(config-router-ospf)#network 1.1.10.20 0.0.0.0 area 0.0.0.0
-------- Exit the Dynamic Routing Module
localhost(config-router-ospf)#exit
localhost(config)#exit
-------- Write configuration to disk
localhost#write memory
IU0 999 Configuration written to '/etc/gated.ami'
```

The same configuration needs to be applied to each cluster member.

As the FIB Manager uses TCP 2010 for routing information synchronization, the Security Policy must accept all traffic on port TCP 2010 between cluster members.

For detailed information regarding Dynamic Routing, see the *R80.10 Advanced Routing Suite CLI Reference guide* http://supportcontent.checkpoint.com/solutions?id=sk111841.

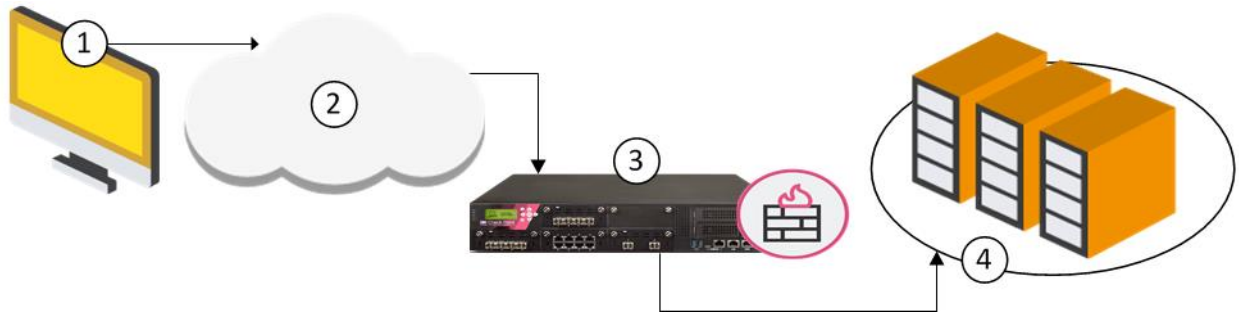
ConnectControl - Server Load Balancing

ConnectControl is a solution for server load balancing. ConnectControl distributes network traffic between a number of servers, to improve network response time and to provide High Availability.

ConnectControl runs on the gateway and does not require more memory or processing power.

ConnectControl Packet Flow

Load-balanced servers are represented by one virtual IP address. Define a *Logical server*, a network object that represents a group of physical servers. The Logical server takes service requests for the load-balanced application and directs the requests to the applicable physical server.



When a client requests access to an application that is load balanced by ConnectControl, the request goes through the Security Gateway.

Item	Description
1	**Client request** - A client starts a connection with the logical IP address of the application server (the address assigned to the Logical server).
2	**Internet** - The service request goes through the Internet.
3	**Security Gateway** - The service request arrives at the destination public IP address of the Logical Server, which is on the Security Gateway. The request is matched to the Logical Server rule in the Rule Base. The gateway directs the request to the internal IP address of the Logical Server group.
4	**Logical Server** - ConnectControl determines which server in the Logical Server group is best for the request, based on the selected load-balancing method.

Note - Make sure that rules that allow traffic for services to ConnectControl Logical Servers and that server groups are before Access Control Policy Firewall rules that allow traffic for those services.

To define a Logical Server:

1. In the Object Explorer, click **New** > **Network Object** > **More** > **Logical Server**.
2. In the **New Logical Server** window, enter a name for the ConnectControl Logical server.
3. Enter a virtual IP address.

4. Select the **Server type** ("Logical Server Types" on page 131).
5. Select the **persistent server mode**.
6. Select the Load **Balance method**.
7. Select a server group, or define a new group (**Servers group** > **New**).
   The members of the group must be hosts, gateways, or OSE devices.
8. Click **OK**.

Logical Server Types

When you create the Logical server object, configure the server type as **HTTP** or **Other**. This distinction is important. ConnectControl handles the connection to the client differently for each server type.

The **HTTP** server type uses HTTP redirection. This type supports offsite HTTP servers and form-based applications, but only works with the HTTP protocol. An HTTP Logical server makes sure that all HTTP-connection sessions are directed to one server, which is a requirement for many Web applications. ConnectControl finds the correct physical server, behind the firewall or offsite, based on the selected load-balancing method ("Load-Balancing Methods" on page 132). The session connections continue to go to that one server.

The **Other** server type uses NAT (address translation) to send traffic to the grouped servers. This Logical server supports all protocols (including HTTP) and gives the most effectively balanced load. It requires servers to be NATed by the gateway. ConnectControl mediates each service request and then selects the server to get that request. It uses NAT to change the destination IP address of the incoming packet. If a return connection is opened, the connection is automatically established between the server and the client. The server's source address in the packet is translated to the IP address of the Logical server. On the packet's return, the firewall translates the packet's original address to the IP address of the Logical server.

Persistent Server Mode

Persistent server mode maintains a client's connection to the server that ConnectControl first selected.

- **Persistency by server** is useful for HTTP applications, such as forms, in a load-balanced environment with multiple Web servers. ConnectControl directs an HTTP client to one server for all requests. This allows clients to fill forms without the data loss that occurs if different servers take the requests.
- **Persistency by service** is useful if you are load balancing multiple services in your server group. For example, in a redundant environment of two servers, each running HTTP and FTP, ConnectControl directs traffic from one client to the server of the correct service. This prevents heavy load on one server, which can happen with **Persistency by server**.

Item	Description
1	Multiple client requests for HTTP and FTP.
2	**Internet**
3	**Security Gateway** - The service requests arrive at the destination public IP address of the Logical Server, which is on the Security Gateway. The gateway directs the requests to the internal IP address of the Logical Server group.

4	**Logical Server** group with two servers, each with FTP and HTTP services. ConnectControl balances the load between the servers.

Persistent Server Timeout

If you enable Persistent server mode, you can set a timeout for a client to use one server. If a server becomes unavailable, ConnectControl directs new connections to a new, available server. This bypasses the persistency and optimizes load balancing.

To set persistent server mode timeout:

1. Open **Global Properties**.
2. Click **ConnectControl**.
3. In **Persistent server timeout**, enter the timeout in seconds.

Load-Balancing Methods

ConnectControl distributes network traffic to load-balanced servers according to predefined balancing methods:

- **Round Trip**: Directs incoming requests to the server with the fastest response time. ConnectControl calculates the fastest server from average round-trip time to respond to ICMP echo requests. The round trip method is a good choice if there are large variations in the traffic load on your network or when load balancing over WAN connections.
- **Round Robin**: Directs service requests to the next server in the sequence. The round robin method is a good choice when all the load balanced servers have similar RAM and CPU and are on the same segment.
- **Random**: Directs service requests to servers at random. The random method is a good choice when all the load-balanced servers have similar RAM and CPU and are located on the same segment.
- **Domain**: Directs service requests based on domain name.

Server Availability

You can configure how ConnectControl finds available servers.

To set server availability configurations:

1. Open **Global Properties**.
2. Click **ConnectControl**.
3. In **Server availability check interval**, enter the number of seconds between pings from the gateway to the servers.
4. In **Server check retries**, enter the number of attempts to contact a nonresponsive server after ConnectControl stops directing connections to it.

End to End ConnectControl

This procedure explains the steps to set up ConnectControl in your environment.

To configure ConnectControl:

1. In the SmartConsole open the **Object Explorer** (Ctrl+E)
2. Click **New** > **Host**.
3. Define the objects for the servers that will be load-balanced.
4. In the **Object Explorer**, click **New** > **Network Group**.
5. Name the group (for example, HTTP_Server_Group).
6. Add the server objects to the group.

   We recommend to add no more than 29 Logical servers to a group.
7. In the **Object Explorer**, click **New** > **Network Object** > **More** > **Logical Server**.
8. Define the Logical server (on page 130).

   Make sure the IP address you assign is a routable IP address. All traffic to be load-balanced is directed through the gateway.
9. Select the **Server type** ("Logical Server Types" on page 131).
10. Select the server group that you defined in these steps.
11. Select the **Persistent Server Mode** (on page 131) that fits your environment.
12. Select a **Balance Method** ("Load-Balancing Methods" on page 132).
13. Add the Load Balancing rule to the Access Control Policy Rule Base:

    Source = `Any`

    Destination = *<Logical Server>*

    Services & Applications = *<load-balanced services>*

    Action = `Accept`, `User Auth`, or `Client Auth`
14. For applications using HTTP redirection, add a rule to allow the server group to communicate directly with clients:

    Source = `Any`

    Destination = `HTTP_Server_Group`

    Services & Applications = `http`

    Action = `Accept`
15. Click **Menu** > **Global Properties** > **ConnectControl**.
16. Set the Persistent Server Timeout (on page 132) and Server Availability (on page 132) controls for your environment.

High Availability Legacy Mode

In This Appendix

Introduction to High Availability Legacy Mode

When configured to work in the High Availability Legacy Mode, all cluster members are assigned the same *shared* IP and MAC addresses. A *shared interface* is an interface whose MAC and IP addresses are identical to those of another interface.

The principal advantage of using this mode is that moving from one Security Gateway deployment to a High Availability cluster requires no changes to IP addresses or routing. Any switch or hub can connect to cluster interfaces. The disadvantage is that configuring this mode is complicated, and must be performed in a precise sequence. You must connect the Security Management Server either to the cluster synchronization network or to a dedicated management network.

Example Legacy Mode Deployment

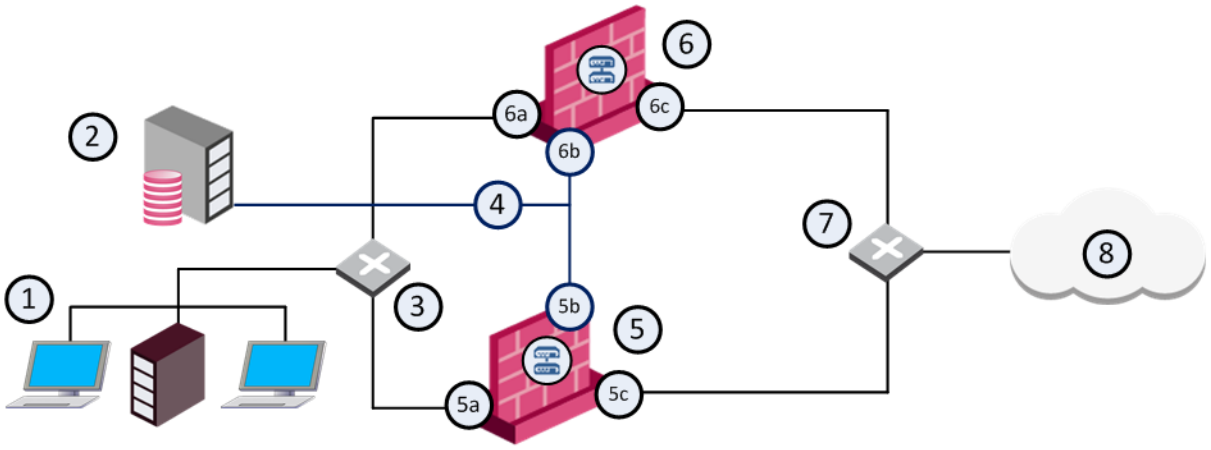
This example shows a typical High Availability Legacy mode deployment. The diagram shows the physical cluster topology for the SmartConsole configuration. There are two cluster members: Member_A (the primary) and Member_B (the secondary), and each gateway has with interfaces.

You must connect the Security Management Server to the Sync Network or to a dedicated management network.

These are the details for the interfaces in the diagram:

- Cluster IP address: 192.168.10.100
- External IP address: 192.168.10.100/24
- Internal IP address: 172.20.0.1/24

- Sync IP address: 10.10.10.0/24



Item	Description
1	Internal network
2	Security Management Server (10.0.10.3)
3	Router or switch for the internal network
4	Synchronization network
5	Security Gateway - Cluster member A
5a	Virtual shared interface to the internal network (172.20.0.1)
5b	Interface to the Cluster Sync network (10.0.10.1)
5c	Virtual shared interface to the external network (192.168.10.1)
6	Security Gateway - Cluster member B
6a	Virtual shared interface to the internal network (172.20.0.1)
6b	Interface to the Cluster Sync network (10.0.10.2)
6c	Virtual shared interface to the external network (192.168.10.1)
7	Router or switch for the external network
8	Internet

Shared Interfaces IP and MAC Address Configuration

With the High Availability Legacy mode, member interfaces connected to the same network (internal, external, DMZ, etc.) share common IP and MAC addresses. Each shared interface on cluster members connects to the appropriate network via a hub or switch.

Only one cluster member is active at any given time. Therefore, the outside world can see only the shared interfaces on one member at any given time.

In the above example deployment, the external interface (facing the Internet) uses the shared IP address **192.168.0.1** for both Member A and Member B. The internal interface (facing the local network) uses IP address **172.20.10.1** for both Member A and Member B.

The Synchronization Interface

State Synchronization between cluster members ensures that existing connections handled by the failed member are maintained during and after failover. The synchronization network passes connection synchronization and other state information between cluster members. Since this network carries sensitive security policy information, it is essential that this connection is secure. You can define more than one synchronization network for backup purposes.

To secure synchronization interfaces, they should be directly connected by a cross-cable, or in the case of three or more cluster members, by means of a dedicated hub, switch, or VLAN.

High Availability cluster members do not have to be synchronized. However, if they are not synchronized, active connections may be lost during failover.

Planning Considerations

IP Address Migration

Legacy mode deployments can be easily migrated from an existing stand-alone Security Gateway configuration to a High Availability cluster. In such cases, we recommend that you assign the existing interfaces IP addresses on your stand-alone Security Gateway as the cluster addresses when feasible. Doing so will avoid altering current IPsec endpoint identities, and in many cases, will make it unnecessary to change Hide NAT configurations.

Security Management Server Location

The Security Management Server downloads Security Policies to all cluster members. The Security Management Server cannot be connected to any network that connects to member *shared interfaces*, because these interfaces are configured to use the same IP and MAC addresses.

The Security Management Server must connect to the cluster synchronization network or a to dedicated management network, because the members will have unique IP addresses.

Routing Configuration

Configure routing so that external network and internal networks are routable to each other.

For example, the sample deployment ("Example Legacy Mode Deployment" on page 134) shows routing configuration as follows:

- Internal networks are defined using **172.20.0.1** as the default gateway.
- The external router is configured with a static route such that network **172.20.0.1** is reached via **192.168.10.1**.

Switch (Layer 2 Forwarding) Considerations

The Cluster Control Protocol (CCP) makes use of layer two multicast. In keeping with multicast standards, this multicast address is used only as the destination, and is used in all CCP packets sent on "non-secured" interfaces.

A Layer 2 switch connected to non-secured interfaces, must be capable of forwarding multicast packets to switch ports, or within a VLAN, if it is a VLAN switch. It is acceptable that the switch forward such traffic to all ports, or to ports within the given VLAN. However, it is considered more efficient to forward to only those ports connecting cluster members.

Most switches support multicast by default. Please check your switch documentation for details.

If the connecting switch is incapable of forwarding multicast, CCP can be changed to use broadcast instead.

To toggle between these two modes:

Use the command: **'cphaconf set_ccp broadcast/multicast'**

Configuring High Availability Legacy Mode

1. Disconnect the Security Gateways that are to become cluster members from switches and/or hubs.
2. Assign the same IP addresses to the shared interfaces on each member. To avoid network conflicts due to sharing MAC addresses, define the IP addresses before physically connecting members to the cluster topology.
3. Install and configure Check Point Security Gateway on all cluster members. Each member must use the identical version and build. See the *R80.10 Installation and Upgrade Guide* http://downloads.checkpoint.com/dc/download.htm?ID=TBD.

   Do not reboot the members at this time.
4. Connect (or reconnect) the members to their switches and hubs. Make sure that you connect each interface to the appropriate physical network port. Connect each network (internal, external, Synchronization, DMZ, etc.) to a separate VLAN, switch or hub. No special switch configuration is required.

Routing Configuration

1. Configure routing so that communication with the networks on the internal side of the cluster is via the cluster IP address on the external side of the cluster. For example, in the sample deployment ("Example Legacy Mode Deployment" on page 134), the external router is configured as a static route such that network **10.255.255.100** is reached via **192.168.10.100**.
2. Configure routing so that communication with the networks on the external side of the cluster is via the cluster IP address on the internal side of the cluster. For example, in the sample deployment ("Example Legacy Mode Deployment" on page 134), the internal router ports are configured with **10.255.255.100** as the default gateway.
3. Reboot the members. MAC address configuration will take place automatically.

SmartConsole Configuration

1. In the **Network Objects** tree, right-click **Check Point** and then select **Security Cluster**.
2. In the **Security Gateway Cluster Creation** window, select **Classic Mode**.

3. In the **Cluster Members** page, click **Add > New Cluster Member** to add cluster members to the cluster. Cluster members exist solely inside the Cluster object. For each cluster member:
   a) In the **Cluster Members Properties** window **General** tab, define a **Name** and **IP Address**. Choose an IP address that is routable from the Security Management Server so that the Security Policy installation will be successful. This can be an internal or an external address, or a dedicated management interface.
   b) Click **Communication**, and Initialize Secure Internal Communication (SIC).
   c) Define the **NAT** and **VPN** tabs, as required.
   d) You can also add an existing Security Gateway as a cluster member by selecting **Add > Add Gateway to Cluster** in the **Cluster Members** page and selecting the Security Gateway from the list in the **Add Gateway to Cluster** window.
   e) If you want to remove a member from the cluster, click **Remove** in the **Cluster Members** page and select **Detach Member from Cluster** or right-click on the cluster member in the **Network Objects** tree and select **Detach from Cluster**.
4. In the **ClusterXL** page,
   - Check **High Availability Legacy Mode**,
   - Choose whether to **Use State Synchronization**. This option is checked by default. If you clear this option, the cluster members will not be synchronized, and existing connections on the failed member will be closed when failover occurs.
   - Specify the action **Upon Gateway Recovery** (see What Happens When a member Recovers? (see "What Happens When a Security Gateway Recovers?" on page 36) for additional information).
   - Define the **Fail-over Tracking** method.
5. In the **Topology** page, define the cluster member addresses. Do not define any virtual cluster interfaces. If converting from another cluster mode, the virtual cluster interface definitions are deleted. In the **Edit Topology** window:
   - Define the topology for each cluster member interface. To automatically read all the predefined settings on the member interfaces, click **Get all members' topology**.
   - In the **Cluster** column, define the purpose of the network by choosing one of the options from the drop-down list. Define the interfaces with shared IP addresses as belonging to a **Monitored Private** network, and define one (or more) interfaces of each cluster member as synchronization interface in a synchronization network (**1st Sync/2nd Sync/3rd Sync**). The options are explained in the Online Help. To define a new network, click **Add Network**.
6. Define the other pages in the Cluster object as required (**NAT, VPN**, **Remote Access**, etc.).
7. Install the Security Policy on the cluster.
8. Reboot all the cluster members in order to activate the MAC address configuration on the cluster members.

Configuring General Properties

To configure the general properties of a cluster:

1. Enter a unique name for this cluster object in the designated field.
2. Enter the virtual cluster IPv4 and IPv6 addresses.
3. Select the hardware platform, Check Point version and operating system.
4. Select **ClusterXL** and other Network Security Software Blades as necessary.

Defining Cluster Members

To configure a cluster member:

1. Go to the **Cluster Members** page.
2. Click **Add > New Cluster Member**.
3. In the **Cluster Members Properties** window **General** tab, enter a member **Name** and the physical, IPv4 and IPv6 addresses. These addresses must be routable from the Security Management Server and can be an internal, external or dedicated management interface.

   Important: You must define a corresponding IPv4 address for every IPv6 address. This release does not support pure IPv6 addresses.
4. Click **Communication**, and initialize Secure Internal Communication (SIC) trust.
5. Configure **NAT** and **VPN** settings on the appropriate tabs as required.

Removing a Member

To remove a member from the cluster:

1. Click **Remove** in the **Cluster Members** page.
2. Select **Detach Member from Cluster**.

   You can also right-click the cluster member and select **Detach from Cluster**.

Configuring ClusterXL Properties

To configure ClusterXL properties for the legacy mode:

1. Enable the **High Availability** option and select the **Legacy** mode.
2. Select the action to perform upon primary member recovery:
   - Maintain the current active cluster member
   - Switch back to a higher priority cluster member
3. For High Availability New deployments, State Synchronization is optional, but enabled by default. If you choose to disable State Synchronization, cluster members do not synchronize, and existing connections on the failed Security Gateway will be terminated once failover occurs.
4. Select tracking options from the list.

Completing the Definition

1. Configure other blades and options for the cluster object as required (**NAT, VPN**, **Remote Access**, and other advanced options).
2. Install a Security Policy on the cluster.

Moving from High Availability Legacy with Minimal Effort

This procedure describes how to move from High Availability Legacy mode to Load Sharing Multicast mode or to High Availability New mode, when the consideration is simplicity of configuration, rather than the minimal downtime.

The shared internal and external interfaces become cluster interfaces. The general IP address of the cluster therefore stays as an external cluster IP address.

On the Security Gateways

1. Run **cpstop** on all members (all network connectivity will be lost).
2. Reconfigure the IP addresses on all the cluster members, so that unique IP addresses are used instead of shared (duplicate) IP addresses.

   Note - SecurePlatform only: These address changes delete any existing static routes. Copy them down for restoration in step 4
3. Remove the shared MAC addresses by executing the command:
   cphaconf uninstall_macs
4. SecurePlatform cluster members only: Redefine the static routes deleted in **step 2.**
5. Reboot the members.

From SmartConsole

In SmartConsole, open the cluster object, select the **ClusterXL** tab, change the cluster mode from **Legacy mode** to **new mode** or to **Load Sharing mode**. Then follow the Check Point Cluster Wizard. For *manual* configuration, proceed as follows:

1. In the **Topology** tab of the cluster object,
   - For each cluster member, get the interfaces which have changed since the IP addresses were changed. The interfaces which were previously shared interfaces should now be defined as Cluster interfaces.
   - Define the cluster IP addresses of the cluster. The cluster interfaces' names may be defined as you wish as they will be bound to physical interfaces according to the IP addresses.

     If the new IP addresses of the cluster members on a specific interface reside on a different subnet than the cluster IP address in this direction, the cluster members' network should be defined in the **Members Network** fields of the cluster interface (Configuring Cluster Addresses on Different Subnets).
2. Install the policy on the new cluster object (Security policy, QOS policy and so on).

Moving from High Availability Legacy with Minimal Downtime

This procedure describes how to move from Legacy Check Point High Availability to New Check Point High Availability or to Load Sharing while minimizing the downtime of the cluster.

The shared internal and external interfaces become the cluster interfaces. As the cluster members will need additional IP addresses these must be prepared in advance.

If downtime of the cluster during the change is not a major issue, it is recommended to use the easier process described in Moving from High Availability Legacy with Minimal Effort (on page 139).

Note - 1. Make sure that you have all the IP addresses needed before you start implementing the changes described here.
2. Backup your configuration before starting this procedure, because this procedure deletes and recreates the objects in SmartConsole.

In this procedure we use the example of members 'A' and 'B', with the starting point being that computer 'A' is active, and computer 'B' is on standby.

1. Disconnect computer 'B' from all interfaces except the interface connecting it to the Security Management Server (the management interface).
2. Run **cphastop** on computer 'B'.
3. Change the IP addresses of computer 'B' (as required by the new configuration).

   Note - SecurePlatform only: These address changes delete any existing static routes. Copy them down for restoration in **step 5**.
4. Reset the MAC addresses on computer 'B' by executing **cphaconf uninstall_macs.** The Windows computer must be rebooted for the MAC address change to take effect.
5. SecurePlatform cluster members only: Redefine the static routes deleted in **step 3**.
6. In SmartConsole, right-click member 'A' and select **Detach from cluster**.
7. In the **Topology** tab of the **Cluster Member Properties** window, define the topology of cluster member 'B' by clicking **Get**. Make sure to mark the appropriate interfaces as **Cluster Interfaces**.
8. In the **Cluster Object**, define the new topology of the cluster (define the cluster interfaces in the cluster **Topology** tab).
9. In the **ClusterXL** page, change the cluster **High Availability** mode from **Legacy Mode** to **New Mode** or select **Load Sharing** mode.
10. Verify that the other pages in the **Cluster Object** (NAT, VPN, Remote Access and so on) are correct. In Legacy Check Point High Availability, the definitions were per cluster member, while now they are on the cluster itself.
11. Install the policy on the cluster, which now only comprises cluster member 'B'.
12. Reconnect computer 'B' (which you disconnected in step 1) to the networks.
13. In this example the cluster comprises only two members, but if the cluster comprises more than two members, repeat steps 1-9 for each cluster member.
14. For **Load Sharing Multicast** mode, configure the routers ("Load Sharing Multicast Mode" on page 40).
15. Disconnect computer 'A' from the all networks accept the management network. The cluster stops processing traffic.
16. Run **cphastop** on computer 'A'.
17. Run **cpstop** and then **cpstart** on computer 'B' (if there are more than two members, run these commands on all members except 'A').
18. Computer 'B' now becomes active and starts processing traffic.
19. Change the IP addresses of computer 'A' (as required by the new configuration).
20. Reset the MAC addresses of computer 'A' by executing **cphaconf uninstall_macs**. The Windows computer must be rebooted for the MAC address change to take effect.
21. In SmartConsole, open the **Cluster Object** and select the **Cluster Members** page. Click **Add > Add Security Gateway to Cluster** and select member 'A' to re-attach it to the cluster.
22. Reconnect computer 'A' to the networks from which it was disconnected in step 15.
23. Install the security policy on the cluster.

24. Run **cpstop** and then **cpstart** on computer 'A'.
25. Redefine static routes

The cluster now operates in the new mode.

ClusterXL Sync Network Configuration

When connecting the members of a cluster to the sync network:

- If the cluster has only two members, connect the sync ports with a cross cable or straight cable
- For more than two members, connect the sync ports through a switch.

Greater redundancy

For greater redundancy, Check Point recommends using:

- *Sync over Bond HA*. Connect the sync bond interface of each member to the same switch or VLAN.
- An isolated secured network segment for the Sync network.

Example cphaprob Script

The **ClusterXL_monitor_process** script shown below has been designed to monitor the existence of given processes and cause failover if the processes die. It uses the normal `pnote` mechanism.

The **ClusterXL_monitor_process** script is located in **$FWDIR/bin.**

In This Appendix

More Information

- The **cphaprob** command is described in Verifying that a Cluster is Working Properly (see "Making Sure that a Cluster is Working" on page 61).

The clusterXL_monitor_process script

```
#!/bin/sh
# This script monitors the existence of processes in the system. The process names
should be written
# in the $FWDIR/conf/cpha_proc_list file one every line.
# USAGE :
# cpha_monitor_process X silent
# where X is the number of seconds between process probings.
# if silent is set to 1, no messages will appear on the console.
# We initially register a pnote for each of the monitored processes
# (process name must be up to 15 characters) in the problem notification mechanism.
# when we detect that a process is missing we report the pnote to be in "problem"
state.
# when the process is up again - we report the pnote is OK.
if [ "$2" -le 1 ]
then
      silent=$2
else
      silent=0
fi
if [ -f $FWDIR/conf/cpha_proc_list ]
then
      procfile=$FWDIR/conf/cpha_proc_list
else
      echo "No process file in $FWDIR/conf/cpha_proc_list "
      exit 0
fi
arch=`uname -s`
for process in `cat $procfile`
do
      $FWDIR/bin/cphaprob -d $process -t 0 -s ok -p register > /dev/null 2>&1
done
while [ 1 ]
do
        result=1
        for process in `cat $procfile`
        do
               ps -ef  | grep $process | grep -v grep > /dev/null 2>&1
```

```
                status=$?
                if [ $status = 0 ]
                then
             if [ $silent = 0 ]
                then
                          echo " $process is alive"
                       fi
#                       echo "3, $FWDIR/bin/cphaprob -d $process -s ok report"
            $FWDIR/bin/cphaprob -d $process -s ok report
                else
            if [ $silent = 0 ]
               then
                         echo " $process is down"
                          fi
                      $FWDIR/bin/cphaprob -d $process -s problem report
                       result=0
                fi
        done
        if [ $result = 0 ]
        then
            if [ $silent = 0 ]
          then
                echo " One of the monitored processes is down!"
      fi
   else
   if [ $silent = 0 ]
          then
                echo " All monitored processes are up "
              fi
        fi
      if [ "$silent" = 0 ]
      then
      echo "sleeping"
      fi
       sleep $1
done
```