Hewlett Packard
Enterprise

HPE Security ArcSight Connectors

SmartConnector for Check Point OPSEC NG

Configuration Guide

June 15, 2017

Configuration Guide

SmartConnector for Check Point OPSEC NG

June 15, 2017

Warranty

Restricted Rights Legend

Revision History

Date	Description
06/15/2017	Added important information about LEA server and sha-256 certificate requirement.
05/15/2017	Added note regarding upgrade of connector to configuration section.
02/15/2017	End of support for Check Point Security Gateway versions R71, R75 and R76 as these versions are no longer supported by Check Point.
11/30/2016	Updated troubleshooting information regarding missing DLL file. Updated installation procedure for setting preferred IP address mode. Note that IPv6 is not supported for this connector.
08/30/2016	Added clarification to opsec_sscla_file parameter description.
06/30/2016	Updated all steps in the "Installing PAM Package for CentOS and RHEL OS" section.
05/16/2016	Added link to Microsoft information in Troubleshooting item regarding missing DLL files.
03/31/2016	Added troubleshooting information for required Microsoft Visual Studio redistributable.
02/15/2016	Added troubleshooting information for IP address incorrect error when pulling certificate. Changed incorrect specification of /export/home/$ARCSIGHT_HOME variable to /opt/arcsight/chkpoint in shared library examples.

Contents

SmartConnector for Check Point OPSEC NG

This guide provides information for installing the SmartConnector for Check Point OPSEC NG and configuring the device for event collection. This connector supports Check Point Security Gateway version R77. Solaris 11 x86 is not a supported platform for this connector.

Check Point has updated their servers to be able to use SHA-256 certificates. A newer LEA client is needed to support these SHA-256 certificates. The SmartConnector for Check Point OPSEC NG does not use this new LEA client; therefore, if you upgrade the certificate to SHA-256, the connector will no longer be able to connect to collect events. The recommendation is to use the SmartConnector for Check Point Syslog to collect Check Point events. Note that the R77.30 Add-On on the Security Management Server or Multi-Domain Server is required for syslog event collection (see sk105412 http://supportcontent.checkpoint.com/solutions?id=sk105412) .

The following Check Point Security Gateway event types are supported:

Anti-bot
Anti-spam and Email Security
Anti-virus
Application Control
Data Loss Prevention
Firewall
Identity Awareness
IPS
URL Filtering

Product Overview

Check Point's Open Platform for Security (OPSEC) integrates and manages all aspects of network security through an open, extensible management framework. The Check Point OPSEC Software Development Kit (SDK) provides Application Programming Interfaces (APIs) for open protocols. It includes the Log Export API (LEA), which lets ArcSight securely receive both realtime and historical log data generated by Check Point VPN-1/FW-1.

The ArcSight SmartConnector for Check Point OPSEC NG uses LEA exclusively. The LEA lets Check Point log data be exported to the ArcSight SmartConnector.

Configuration

The following sections apply to Check Point firewall devices; for configuration information when using Provider-1/SiteManager-1, see "Configuring Provider-1/SiteManager-1 to Accept OPSEC Connections" later in this document. (Provider-1 is also referred to as "Security Management and Multi-Domain Security Management.") For complete configuration information, see your Check Point product documentation.

When upgrading this connector, to avoid performing all configuration steps again, you can copy your previous checkpoint folder to the upgraded connector to maintain your initial configuration.

For example, replace the `SmartConnectorNewVersion\current\currentSupport\user\agent\checkpoint` folder with the `SmartConnectorOldVersion\current\currentSupport\user\agent\checkpoint` folder.

Overview

ArcSight recommends you install the ArcSight SmartConnector (the LEA Client) on a different machine than the Check Point Management Server. Be sure to grant the SmartConnector access rights to this Management Server.

Information you need to determine the values for the following SmartConnector parameters required during connector installation and configuration is provided in this section. Note that installation parameters required differ based upon the connection type. The required parameters for the **sslca**, **clear**, and **ssl_opsec** connection types are specified at the end of each subsection for the connection type. The default and recommended connection method is **sslca**.

Some configuration is required before SmartConnector installation and, for **sslca** and **ssl_opsec** connection methods, additional configuration steps are required during the SmartConnector installation process.

The configuration steps to be performed *prior* to SmartConnector installation are as follows:

- For **clear** connections, configuration steps to be performed before SmartConnector installation include:
  - Edit the `fwopsec.conf` file to add lines specifying port information.
  - Restart the Management Server.
- For **sslca** and **ssl_opsec** connections:
  - Edit the `fwopsec.conf` file to add lines specifying authentication port information. (Usually not required for **sslca** when it is the default connection type.)
  - Restart the Management Server.
  - Create an OPSEC Application Object and get the Client SIC Name
  - Obtain the Server OPSEC SIC Name.

The following configuration steps are to be followed after the SmartConnector core software has been installed, but before individual connector configuration begins. You will be directed how to perform additional configuration during the connector installation process.

For the **sslca** connection type, the following step is required during the SmartConnector installation process:

- Pull the certificate from the Management Server.

For the **ssl_opsec** connection type, the following step is required during the SmartConnector installation process:

- Establish an authentication key.

Configure Clear Connection

To configure the connector to use the clear connection type:

1 Navigate to the Check Point firewall configuration directory. For example, on the Management Server where the LEA Server is running:

```
/$FWDIR/conf
```

This example assumes UNIX. On Windows platforms, the directory might be `\$FWDIR\FW1\conf`, `\$FWDIR\conf`, or `\$FWDIR\5.0\conf`.

2 Edit the `fwopsec.conf` file to contain the following lines.

```
lea_server port 18184
lea_server auth_port 0
```

Save and exit the file.

3 Restart the firewall by issuing the following commands:

```
cpstop
cpstart
```

During SmartConnector installation, you will be asked to enter installation parameter values. Select **clear** as the connection type and enter the values for **server_ip** and **server_port**.

Ignore other parameters such as **opsec_sic_name**, **opsec_sslca_file**, and **opsec_entity_sic_name**. You can leave these fields empty.

Configure sslca or ssl_opsec Connection

The connection type recommended by Check Point is **sslca**. The following configuration steps apply to both **sslca** and **ssl_opsec** connection types.

1 Navigate to the Check Point FW-1 configuration directory. For example, on the Management Server where the LEA Server is running:

```
/$FWDIR/conf
```

This example assumes UNIX. On Windows platforms, the directory might be `\$FWDIR\FW1\conf`, `\$FWDIR\conf`, or `\$FWDIR\5.0\conf`.

2 For the **sslca** connection type, edit the `fwopsec.conf` file to contain the following lines. (If **sslca** is the default connection type, this step may not be necessary.)

```
lea_server auth_port 18184
lea_server auth_type sslca
lea_server port 0
```

For **ssl_opsec** connection, edit the `fwopsec.conf` file to contain the following lines:

```
lea_server auth_port 18184
lea_server auth_type ssl_opsec
lea_server port 0
```

Save and exit the file.

3 Restart the firewall by issuing the following commands:

```
cpstop
cpstart
```

Continue with "Create a New Application Object" and "Obtain the OPSEC SIC Name and OPSEC Entity SIC Name" and complete those sections before starting connector installation.

Create a New Application Object

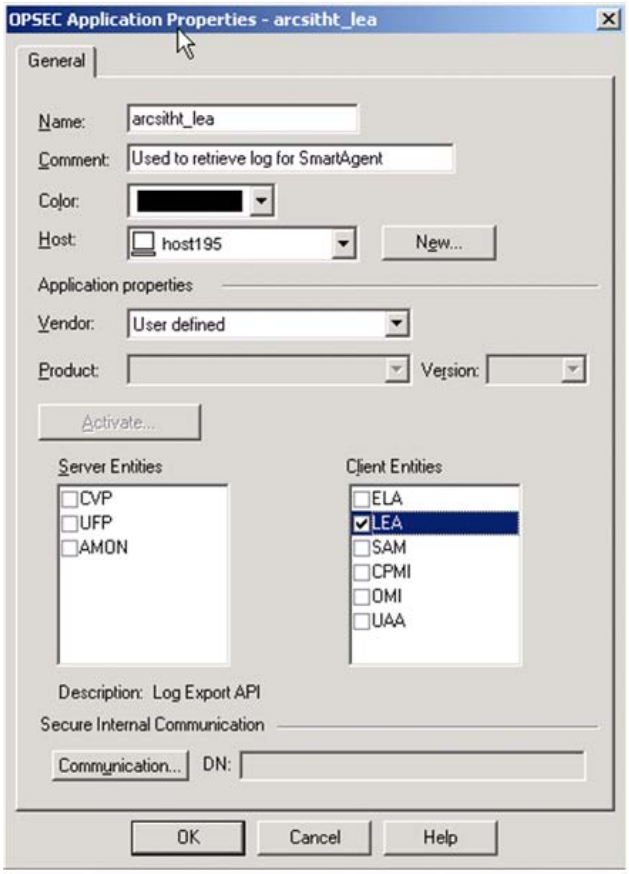
In this section, for the **sslca** and **ssl_opsec** connection types, you will create a new OPSEC Application object for the LEA client. Because the SmartConnector works with several versions of Check Point, specific menu names may vary.

1 Open the Check Point Policy Editor or SmartDashboard.

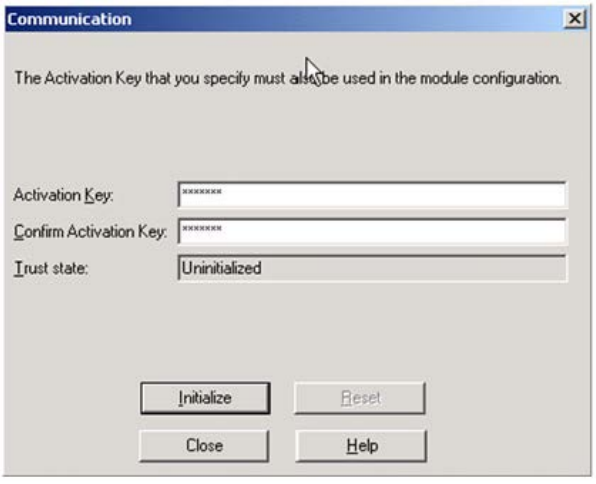
2 From the **Manage** menu, select **OPSEC Applications** (or **Servers and OPSEC Applications**).

3 In the **OPSEC Applications** window, click **New** and select **OPSEC Application** (or **OPSEC Application Properties**).

4 In the **OPSEC Application Properties** window, enter a **Name** for the object, such as `arcsight_lea` (shown in the following example), and any appropriate **Comment**. Note that names cannot contain blank spaces. Select the SmartConnector host in the **Host** field (`host195` in the following example). If the host for the SmartConnector is not in the host list, click **New** and add it (use the **Manage** menu: **Manage -> Network Objects -> New -> Node -> Host**). Select **LEA** from the **Client Entities** section.



5 Click the **Communication** button at the bottom of the window; the **Communication** window is displayed.



Enter an **Activation Key**. Confirm the key by re-entering it in the **Confirm Activation Key** field. This activation key will be used by application `opsec_pull_cert` when the application is executed on the SmartConnector side to pull the certificate of the new OPSEC Application object you created.

6 Click **Initialize**; then, click **Close**.

Continue configuration with "Obtain the OPSEC SIC Name and OPSEC Entity SIC Name" before starting SmartConnector installation. This section is required for **sslca** and **ssl_opsec** connection types.

Obtain the OPSEC SIC Name and OPSEC Entity SIC Name

For the **sslca** and **ssl_opsec** connection types, the next steps are to obtain the server OPSEC SIC Name (opsec_sic_name) and the client OPSEC Entity SIC Name (opsec_entity_sic_name) required during connector installation and configuration.

To display all the SIC names, use the Check Point Command Line Interface:

1 Use ssh to connect to the Check Point CLI:

```
ssh admin@<ip_address>
```

where <ip_address> is the IP address of the Check Point device.

2 Enter the following command:

```
[cpmodule]# cpca_client
```

The following is displayed:

```
Usage: cpca_client [-d]
  create_cert [-p <ca_port>] -n "CN=<common name>" -f  <PKCS12 filename>
[-w <password>] [-k <SIC|USER|IKE|ADMIN_PKG>]
  revoke_cert [-p <ca_port> -n "CN=<common name>"
  init_certs [-p <ca_port>] -i input_file -o output_file
  get_crldp [-p <ca_port>]
  set_mgmt_tool on|off|add|remove|clean|print  [-p <ca_port>]  [-no_ssl]
{ [-a <administrator DN>] [-u <user DN>] [-c <custom user DN>] }
  get_pubkey [-p <ca_port>] output_file
  lscert [-dn substr] [-stat Pending|Valid|Revoked|Expired|Renewed] [-
kind SIC|IKE|User|LDAP] [-ser ser] [-dp dp]
```

3 Enter the following command.

```
[cpmodule]# cpca_client lscert -kind SIC
```

The following information is displayed. In this example, the server SIC Name is associated with the OPSEC Client (`CN=arclea,O=cpmodule..3wmgw`), and the SIC Entity Name is associated with the Server Object (`CN=cp_mgmt,O=cpmodule..3wmzgw`). Make note of these SIC names as they will be required during the connector installation process.

```
Operation succeeded. rc=0.  5 certs found.

Subject = CN=arclea,O=cpmodule..3wmzgw
Status = Valid   Kind = SIC   Serial = 38239   DP = 0
Not_Before: Fri Jun 10 13:43:07 2011   Not_After: Thu Jun  9 13:43:07
```

```
2016

Subject = CN=cpmodule,O=cpmodule..3wmzgw
Status = Valid   Kind = SIC   Serial = 92823   DP = 0
Not_Before: Thu Jun  9 16:18:13 2011   Not_After: Wed Jun  8 16:18:13
2016

Subject = CN=esmlea,O=cpmodule..3wmzgw
Status = Valid   Kind = SIC   Serial = 91560   DP = 0
Not_Before: Fri Jun 10 14:11:48 2011   Not_After: Thu Jun  9 14:11:48
2016

Subject = CN=cp_mgmt,O=cpmodule..3wmzgw
Status = Valid   Kind = SIC   Serial = 83018   DP = 0
Not_Before: Thu Jun  9 16:18:08 2011   Not_After: Wed Jun  8 16:18:08
2016

Subject = CN=calea,O=cpmodule..3wmzgw
Status = Pending   Kind = SIC   Serial = 44911
Not_Before: N/A   Not_After: Fri Jun 10 10:54:20 2016
```

You can now continue with "Install the SmartConnector."

- For **sslca** connections, after the connector core software is installed, the connector will need to pull the certificate to gain access to the Management Server.
- For only the **ssl_opsec** connection type, an authentication key also must be established after the core connector software is installed.

Follow the procedures in "Installing the SmartConnector" to complete these steps at the appropriate point during the installation process.

The Check Point OPSEC NG connector should retrieve all kinds of events that are generated by the Check Point product, such as VPN-1, FW-1, FloodGate-1, SecureClient, SmartDefense, FW-1 GX, Voice over IP, or third-party products. So, customization of the SmartConnector parser file to map new fields may be required.

Installing PAM Package for CentOS and RHEL OS

This procedure is to be performed before installation of the core component software.

If the Check Point connector will be running on CentOS (6.5, 6.6, 7.0 or 7.1) or RHEL OS (6.5 or 7.0), install the Pluggable Authentication Modules (PAM) package before installing the CheckPoint connector. Otherwise, you may get an error message when executing `opsec_pull_cert` on the LEA Client side. PAM is a system of libraries that handles the authentication tasks of applications and services. The library provides a stable API for applications to defer to for authentication tasks.

To install the PAM package:

1 Install PAM: `yum install pam`

2 Install libpam.so.0: `yum install libpam.so.0`

3 Create a soft link to `libpam.so.0` in one of the following ways:.

```
ln -s /usr/lib/libpam.so.0
<arcsight_home>/bin/agent/checkpoint/OPSECAD/linux/libpam.so.0
```

or

```
ln -s /usr/lib/libpam.so.0
<arcsight_home>/bin/agent/checkpoint/OPSECNG/linux/libpam.so.0
```

depending on which LEA client is used.

Pull the Certificate - sslca

This procedure is to be followed after installation of core component software. See step 3 in "Installing the SmartConnector."

If you selected a certificate authentication connection type of **sslca**, pull the certificate from the VPN-1/FW-1 Management Server after installing the core connector software (see step 3 in "Installing the SmartConnector").

To pull the certificate, run the **opsec_pull_cert** application from the command line on the SmartConnector that needs the certificate (that is, on the LEA Client).

In these steps, we are assuming the ArcSight SmartConnector is installed on a Windows machine and under the folder `c:\$ARCSIGHT_HOME`. Navigate to the following directory:

```
C:\$ARCSIGHT_HOME\current\bin\agent\checkpoint\OPSECAD\win32\
```

Enter the following command to pull the authentication certificate:

```
opsec_pull_cert -h <lea_server1's IP Address>
                -n <lea_client application name>
                -p <password used when creating the lea_client>
                -o <name of the output file; by default, opsec1.p12>
```

Copy the output file to the `C:\$ARCSIGHT_HOME\current\user\agent\checkpoint` directory.

If a SmartConnector has been installed on the Linux platform and `'libcpc++-lib6.1-2.so.3'` is not on the system, follow the procedure in "Change the LD_LIBRARY_PATH Variable" and change the environment variable LD_LIBRARY_PATH before you execute `'opsec_pull_cert'` on the LEA Client side.

After executing the opsec_pull_cert command, install the Check Point firewall policy to allow the communication between the connector host and the target Check Point firewall.

The correct path for the sslca file as required by the SmartConnector is:

```
$ARCSIGHT_HOME/current/user/agent/checkpoint
```

When configuring the connector, the filename is all that is required because the connector, by default, is looking in the directory specified above for the filename entered in the connector's parameter entry table, hence the full path is not required.

When setting up the connector on the Connector Appliance, this is accomplished with the Send Container command, but when doing it from the command line for software connectors, change directories to `$ARCSIGHT_HOME/current/bin/agent/checkpoint/OPSECAD/<os>`, where `<os>` is linux or win32, and then run the **opsec_pull_cert** command.

Here is an example from the Connector Appliance when it issued the opsec_pull_cert via Send Container command (notice it specifies the full paths when pulling and where the .p12 is stored).

```
/opt/arcsight/connector_1/current/bin/agent/checkpoint/OPSECAD/linux/opse
c_pull_cert -h 10.0.101.185 -n calea -p arcsight -o
/opt/arcsight/connector_1/current/user/agent/checkpoint/calea.opsec.p12 -
od
/opt/arcsight/connector_1/current/user/agent/checkpoint/calea.opsec.sic
```

For command line troubleshooting, and in order to avoid entering a long path name, the certificate is just pulled in this directory, then copied to the `$ARCSIGHT_HOME/current/user/agent/checkpoint` directory. This way, if you are testing the lea_client from `$ARCSIGHT_HOME/current/bin/agent/checkpoint/OPSEC/<os>`, where `<os>` is linux or win32, the command line might look like:

```
lea_client -m online -t sslca -h 10.0.101.185 -p 18184 -s
CN=arclea,O=cpmodule..3wmzgw -f opsec.p12 -e
CN=cp_mgmt,O=cpmodule..3wmzgw
```

If the opsec.p12 file was found only in the `$ARCSIGHT_HOME/current/user/agent/checkpoint` directory, the command line might look like:

```
lea_client -m online -t sslca -h 10.0.101.185 -p 18184 -s CN-
arclea,O=cpmodule..3wmzgw -f
/home/connector/current/user/agent/checkpoint/opsec.p12 -e
CN=cp_mgmt,O=cpmodule..3wmzgw
```

Change the LD_LIBRARY_PATH Variable

Assume that the ArcSight SmartConnector is installed under the folder `/opt/arcsight/chkpoint`.

```
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/opt/arcsight/chkpoint
/current/bin/agent/checkpoint/OPSECAD/linux
```

Or, if the LD_LIBRARY_PATH environment variable has not yet been created:

```
LD_LIBRARY_PATH=/opt/arcsight/chkpoint/current/bin/agent/checkpoint
/OPSECAD/linux
```

Otherwise, you may get an error message such as `./opsec_pull_cert error while loading shared libraries: libcpc++-libc6.1-2.so.3: cannot open shared object file: No such file or directory.`

For **ssl_opsec** connection type, continue with "Establish an Authentication Key" before returning to step 4 in "Installing the SmartConnector."

Establish an Authentication Key – ssl_opsec Only

This procedure is to be followed after installation of core component software. See step 3 in "Installing the SmartConnector."

If you selected **Authenticated and encrypted connection (ssl_opsec)** as the connection type, establish an authentication key by performing the following operations on both the LEA Server (the Check Point Management Server) and the LEA Client (the ArcSight SmartConnector) machines.

If the LEA Client and LEA Server are both on the same machine, this procedure is unnecessary.

On the LEA Server side:

The LEA Server is the machine where the Check Point Management Server is installed. In the following steps, assume that the Management Server is installed under the C: drive on a Windows machine.

Navigate to the following directory:

```
C:\$FWDIR\FW1\bin
```

Enter the following command:

```
fw putkey -opsec -ssl <LEA Client IP Address>=>
```

When the fw command is executed, you are prompted to enter a password for establishing the new authentication key. You will be prompted to enter the same password when you execute the following **opsec_putkey** command from the LEA Client.

After executing put_keys commands, install the Check Point firewall policy to allow the communication between the connector host and the target Check Point firewall.

On LEA Client side:

The LEA Client is the machine where the ArcSight SmartConnector is installed. In the following steps, assume that it is also a Windows machine and the ArcSight SmartConnector is installed under `C:\$ARCSIGHT_HOME`. Navigate to the following directory:

```
C:\$ARCSIGHT_HOME\current\bin\agent\checkpoint\OPSECAD\win32
```

Enter the following command:

```
opsec_putkey -ssl <LEA Server IP Address>
```

When the **opsec_putkey** command listed above is executed, you are prompted to enter an activation key. Enter the same one you entered when you executed the `fw` command from the LEA Server.

After executing put_keys commands, install the Check Point firewall policy to allow the communication between the connector host and the target Check Point firewall.

Copy the output file `sslauthkeys.C` to the `C:\$ARCSIGHT_HOME\current\user\agent\checkpoint` directory.

If a SmartConnector has been installed on a Linux platform with a connection type other than clear and `libcpc++-libc6.1-2.so.3` is not in the system, perform the procedure in the following subsection to change the environment variable LD_LIBRARY_PATH.

Change the LD_LIBRARY_PATH Variable

The following configuration will not persist across reboots; to make this configuration persist, add these lines to your respective profile. Performing this step is not required for the **clear** connection type.

Assume that the ArcSight SmartConnector is installed under the folder `/export/home/arcsight` and execute the following command:

```
export  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/export/home/arcsight/current
/bin/agent/checkpoint/OPSECAD/linux
```

Or, if the LD_LIBRARY_PATH environment variable has not yet been created, execute this command:

```
export LD_LIBRARY_PATH=/export/home/arcsight/current/bin/agent/checkpoint
/OPSECAD/linux
```

Otherwise, you may get an error message such as `./opsec_putkey: error while loading shared libraries: libcpc++-libc6.1-2.so.3: cannot open shared object file: No such file or directory`.

1 On **lea_server1** and **lea_server2**, respectively, execute the following **fw putkey** command with the **welcome1** and **welcome2** passwords:

```
C:\$FWDIR\FW1\bin>fw putkey -opsec -ssl <lea_client's IP Address>
```

2 On **lea_client**, execute the **opsec_putkey** command twice, using the passwords **welcome1** and **welcome2** respectively.

```
C:\>cd C:\$ARCSIGHT_HOME\current\user\agent\checkpoint
C:\$ARCSIGHT_HOME\current\user\agent\checkpoint>
C:\$ARCSIGHT_HOME\current\bin\agent\checkpoint\OPSECAD\win32>opsec_putkey
-ssl <lea_server1's IP address>
C:\$ARCSIGHT_HOME\current\user\agent\checkpoint>C:\$ARCSIGHT_HOME\current
\bin\agent\checkpoint\OPSECAD\win32>opsec_putkey -ssl <lea_server2's IP
address>
```

After executing put_keys commands, install the Check Point firewall policy to allow the communication between the connector host and the target Check Point firewall.

Configure Provider-1/SiteManager-1 to Accept OPSEC Connections

In Check Point Provider-1/SiteManager-1, logs are generated by a customer's network modules, Customer Management Add-ons (CMAs), and the Multi-Domain Server (MDS). Logs can be stored locally on modules, or a remote machine can be deployed to handle log repository storage. The module sends logs to a log server, which collects and stores them. In Provider-1, the log server is by default

the CMA. For complete information about configuring Provider-1/SiteManager-1, see Check Point product documentation.

1 Edit the `fwopsec.conf` file to add lines specifying port and authentication port information.

2 Restart the Management Server.

If your OPSEC client connects to a single OPSEC server, Check Point recommends using the default method of **sslca** (when possible). This is preferred, as changes to `fwopsec.conf` and `sic_policy.conf` are not recommended and are not supported during upgrades.

Clear Connection Type

For debugging purposes, you can still enable clear as a connection method by modifying `$FWDIR/conf/fwopsec.conf` using the following setting:

```
lea_server auth_port 0
lea_server port 18184
```

Restart the Management Server after modifying `fwopsec.conf`.

ssl_opsec Connection Type

To use an auth_type method of **ssl_opsec**:

1 Verify that the `$FWDIR/conf/fwopsec.conf` file on the CMA is set to the correct values for the OPSEC client connection. For example:

```
lea_server auth_port 18184
lea_server auth_type ssl_opsec
lea_server port 0
```

2 Set the environment to that of the CMA by running `mdsenv <cma>`.

3 Change the `$CPDIR/conf/sic_policy.conf` file on the CMA. In the **Inbound rules** section, change the OPSEC client connection from:

```
# OPSEC: backward compatibility services
# (ssl is supported upon request, not by default)
#LEA:
ANY ; ANY ; 18184 ; fwn1_opsec ; fwn1_opsec, fwn1, local_ipcheck
```

to:

```
# OPSEC: backward compatibility services
# (ssl is supported upon request, not by default)
#LEA:
ANY ; ANY ; 18184 ; ssl_opsec; ssl, fwn1, local_ipcheck
```

4 Stop and start the CMA to enable the change to take effect.

5 Use putkeys on the OPSEC client and OPSEC server:

```
fw putkey on the CMA
opsec_putkey on the LEA client host
```

After executing put_keys commands, install the Check Point firewall policy to allow the communication between the connector host and the target Check Point firewall.

The OPSEC client now should be able to connect to the OPSEC server.

Provider-1 Supplemental Information

Check Point uses a Policy database in the Provider-1 environment that should allow lea_client communication to CLM OPSEC objects once a User Database Policy has been created and pushed down to the other CLM entity objects.

The User Database Policy is **not** the same as the Firewall Policy and, even if the Firewall policy has been pushed down to the firewalls themselves, the User Database must be pushed (installed) as a separate action.

Check Point Technical Support confirmed and explained that the certificate authority runs only on the CMA. However, the lea_client will be able to connect through sslca to any given CLM once a Check Point Database Policy is created and all the objects are pushed down to the CLM hosts managed by that CMA.

Therefore, for each CMA environment, in the Check Point SmartDashboard, from the main menu select **Policy** and then select **Install Database...**. The certificates, hosts, and objects will be pushed down to the CLMs managed by the CMA. This step is required even if the CMA and CLM are in fact hosted on the **same** server (as might be the case in a smaller deployment).

If you subsequently add or change any of the OPSEC Policy Objects, you will need to repeat the previous step.

To configure the connector for Provider-1 in an MLM environment to use sslca (the preferred connection method), make sure you use a unique Application Object Name for each CMA for which you need to pull a certificate; otherwise the connector will not be able to pull the certificate.

In the following example, there are two CMAs (each with multiple CLM log servers), one in Singapore (10.1.1.2) and the other in Sydney (10.1.2.4). The OPSEC Application Objects are named:

Arcsight-lea-sin
Arcsight-lea-syd

Each Communication Trust needs to be established, but for each CLM host that resides in the different CMA region, you also must push the database policies down to the CLM objects (`Policy -> Install DB`).

On the connector side, you will pull both certificates from each CMA.

From the `$ARCSIGHT_HOME\bin\agent\checkpoint\OPSECAD\Win32` directory, enter the following commands:

```
opsec_pull_cert -h 10.1.1.2 -n Arcsight-lea-sin -p test1234 -o lea-
sin.p12
opsec_pull_cert -h 10.1.1.2 -n Arcsight-lea-syd -p test1234 -o lea-
syd.p12
```

Once the certificates (lea-sin.p12 and lea-syd.p12) have been pulled and copied into the `$ARCSIGHT_HOME\current\user\agent\checkpoint` directory, the lea_client should be able to establish the sslca session to each respective CLM host being managed and contain the log files the connector processes.

Before configuring the connector properties (though `runagentsetup` or `arcsight agentsetup`), and to make troubleshooting easier, always test from the command line after pulling the certificates and copying them to the `$ARCSIGHT_HOME\current\user\agent\checkpoint` directory. If additional debugging details are needed or required by Support, before running the lea_client from the command line, optionally set the following the lea_client:

```
SET OPSEC_DEBUG_LEVEL=3 (for Linux, export OPSEC_DEBUG_LEVEL=3)
```

In the lea_client configuration, remember to specify the CLM's IP address, the correct sslca cert file, and the CLM's DN Communication string (SIC Entity name).

The following is an example command line session. Notice the host being used is the IP address of a CLM in Singapore as well as the CLM's SIC Entity name (DN Communication string).

```
D:\arcsightagent\SIN-Provider-1-
CLM\current\bin\agent\checkpoint\OPSECNG\win32>
lea_client -m online -t sslca -h 10.1.1.9 -p 18184 -s CN=Arcsight-lea-
sin,O=Singapore_IPTel-Primary..4bsqqj -f lea-sin.p12 -e
CN=SingCLMhost.O=Singapore_IPTel-Primary..4bsqqj
```

If there are no typographical errors, you can then connect to each CLM host managed by the CMA where the Policy DB was created and pull that CLM's logs.

You can then run the connector setup and specify each CMA and CLM in the connector setup wizard and have the wizard validate the connections for each respective host. Thus, effectively specifying multiple sslca clients and even using the same sslca file (CMA host from which the certificate was pulled and the managed CLMs will use the same certificate file.)

Finally, the following is an example of using multiple lines showing each CMA/CLM configuration. (Notice we are using the same opsec_sslca file for the CMA and the CLM in Singapore.)

```
Server_ip | server_port | opsec_sic_name | opsec_sslca_file |
opsec_entity_sic_name
10.1.1.9 18184 CN=Arcsight-lea-sin,O=Singapore-1..4bsqqj lea-sin.p12
CN=SingCLMhost,O=Singapore-1..4bsqqj
10.1.1.8 18184 CN=Arcsight-lea-sin,O=Singapore_CMA..4bsqqj lea-sin.p12
CN=cp_mgmt,O=Singapore_IPTel-Primary..4bsqqj
10.1.2.4 18184 CN=Arcsight-lea-syd.O=Sydney_Provider..rwxyg2 lea-syd.p12
CN=SydCLMhost,O=Sidney_Provider..rwxyg2
```

If the lea_client debug results appear as follows it usually indicates a problem with the lea_client command line parameters or, if the settings are absolutely correct, could indicate a Check Point target system is not functioning correctly.

```
The sessions are terminated by remote lea server[10.1.1.9], which cause
the lea client continues to establish and only establish the ended
sessions. Please also check the communication between the lea client and
the lea server!
MSG06 Create OPSEC LEA client
MSG08 Created OPSEC LEA server entity
MSG26 Called LeaStartHandler : session( 36969952 )
MSG10 Created security log session
MSG26 Called LeaStartHandler : session( 37052080 )
MSG12 Created audit log session
MSG29 Re-established session. SINFO server[0]ip= 10.1.1.9
security_log_session=36969952 audit_log_session=37052080
MSG25 Ended session : One security/audit log session( 36969952 ) ends <
err = 8 Comm is not connected/Unable to connect>
Session Ended Reason: Unknown reason.
```

Install the SmartConnector

The following sections provide instructions for installing and configuring your selected SmartConnector.

Notes:

- When running the connector on Windows platforms, the Microsoft Visual Studio C++ redistributable dll is needed for projects built with Visual Studio. You can download this redistributable from the Microsoft website: http://www.microsoft.com/en-us/download/details.aspx?id=30679#. Download x86 edition for 32-bit platforms.

- Check Point does not offer OPSEC in a 64-bit binary; for 64-bit platforms, use the SmartConnector for Check Point Syslog.

Prepare to Install Connector

Before you install any SmartConnectors, make sure that the ArcSight products with which the connectors will communicate have already been installed correctly (such as ArcSight ESM or ArcSight Logger).

For complete product information, read the *Administrator's Guide* as well as the *Installation and Configuration* guide for your ArcSight product before installing a new SmartConnector. If you are adding a connector to the ArcSight Management Center, see the *ArcSight Management Center Administrator's Guide* for instructions, and start the installation procedure at "Set Global Parameters (optional)" or "Select Connector and Add Parameter Information."

Before installing the SmartConnector, be sure the following are available:

- Local access to the machine where the SmartConnector is to be installed

- Administrator passwords

Install Core Software

Unless specified otherwise at the beginning of this guide, this SmartConnector can be installed on all ArcSight supported platforms; for the complete list, see the *SmartConnector Product and Platform Support* document, available from the HPE SSO and Protect 724 sites.

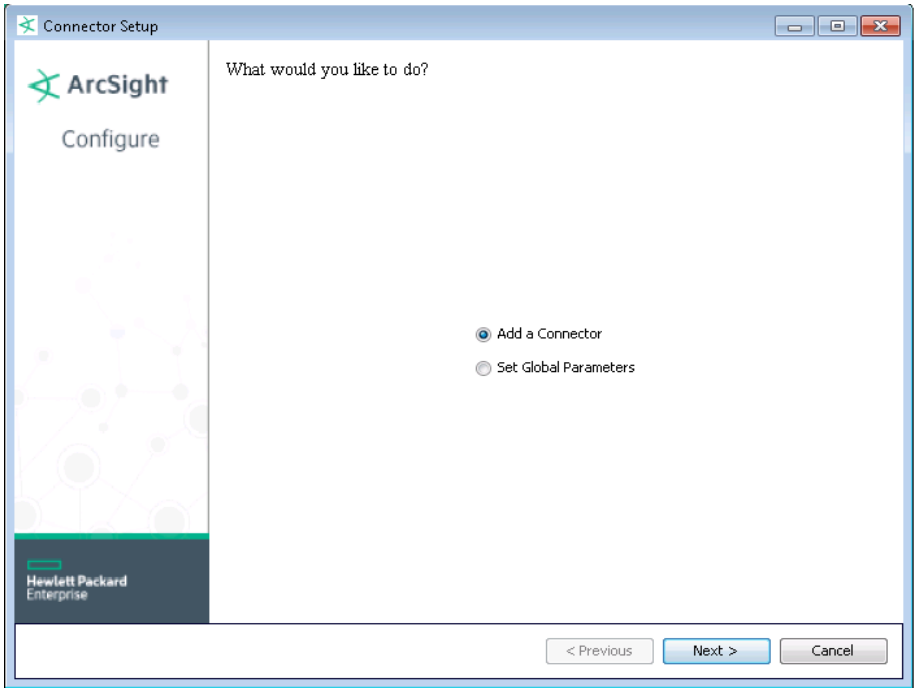
1 Download the SmartConnector executable for your operating system from the HPE SSO site.

2 Start the SmartConnector installation and configuration wizard by running the executable.

Follow the wizard through the following folder selection tasks and installation of the core connector software:

Introduction
Choose Install Folder
Choose Shortcut Folder
Pre-Installation Summary
Installing...

3 When the installation of SmartConnector core component software is finished, the following window is displayed:



A At this point, for **clear** connections, continue with step 4.

B For **sslca** and **ssl_opsec** connections, click **Cancel** to exit the configuration wizard.

C Go to "Pull the Certificate – sslca" and follow the procedure documented. Then continue with step 3E.

D For the **ssl_opsec** connection type, continue with the "Establish an Authentication Key – ssl_opsec Only" procedure, then continue with step 3E.

E From `$ARCSIGHT_HOME/current/bin`, enter `arcsight connectorsetup` to return to the SmartConnector Configuration Wizard. When queried whether to enter Wizard mode, click **Yes**.

Set Global Parameters (optional)

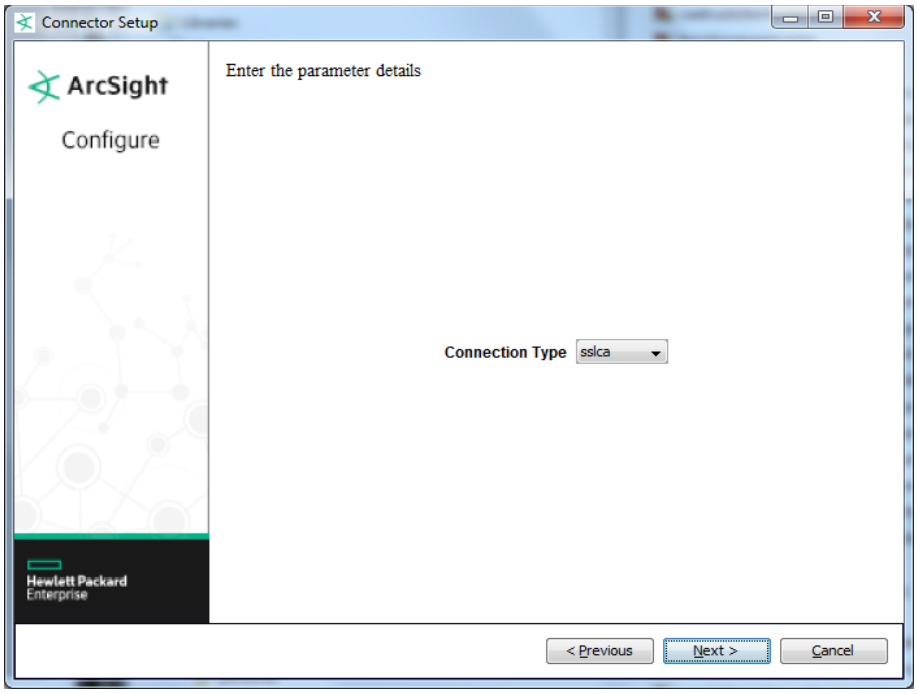
If you choose to perform any of the operations shown in the following table, do so before adding your connector. You can set the following parameters:

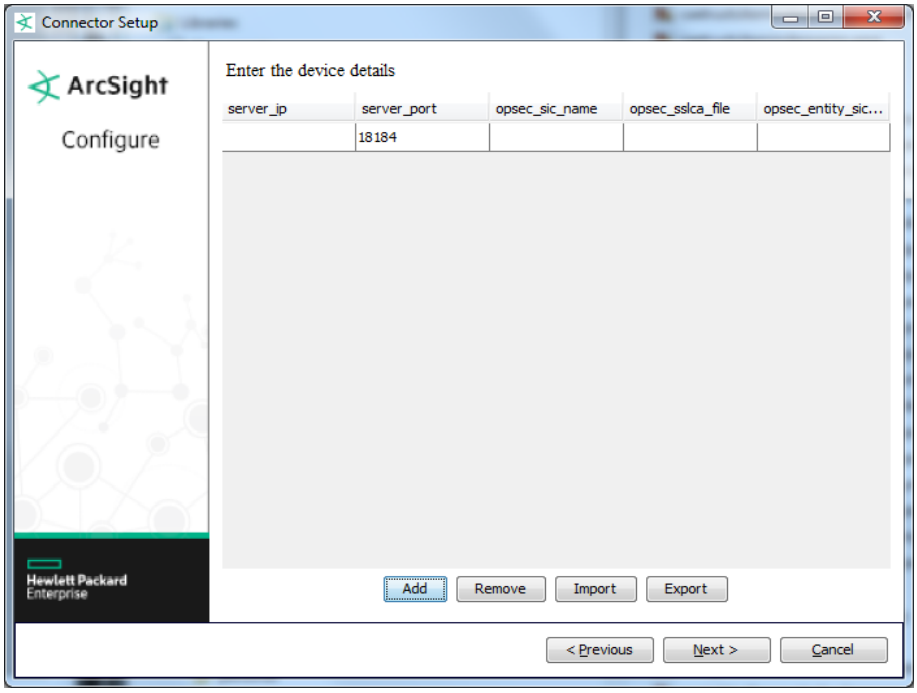
Global Parameter	Setting
Set FIPS mode	Set to 'Enable' to enable FIPS compliant mode. To enable FIPS Suite B Mode, see the SmartConnector User Guide under "Modifying Connector Parameters" for instructions. Initially, this value is set to 'Disable'.
Set Remote Management	Set to 'Enable' to enable remote management from ArcSight Management Center. When queried by the remote management device, the values you specify here for enabling remote management and the port number will be used. Initially, this value is set to 'Disable'.
Remote management listener port	The remote management device will listen to the port specified in this field. The default port number is 9001.
Preferred IP Version	IPv6 is not supported for this connector..

After making your selections, click **Next**. A summary screen is displayed. Review the summary of your selections and click **Next**. Click **Continue** to return to the "Add a Connector" window. Continue the installation procedure with "Select Connector and Add Parameter Information."

Select Connector and Add Parameter Information

1. Select **Add a Connector** and click **Next**. If applicable, you can enable FIPS mode and enable remote management later in the wizard after SmartConnector configuration.
2. Select **Check Point OPSEC NG** and click **Next**.
3. Enter the required SmartConnector parameters to configure the SmartConnector, then click **Next**.


Connector Setup
ArcSight
Configure
Enter the parameter details
Connection Type
sslca
Hewlett Packard Enterprise
< Previous
Next >
Cancel


Connector Setup
ArcSight
Configure
Enter the device details
server_ip
server_port
opsec_sic_name
opsec_sslca_file
opsec_entity_sic...
18184
Add
Remove
Import
Export
Hewlett Packard Enterprise
< Previous
Next >
Cancel

Parameter	Description
Connection Types	Accept the default connection type of 'sslca', which is recommended, or select 'ssl_opsec,' or 'clear' as the connection type. For SSL-based authentication with data encrypted using a 3DES key, use 'sslca.' After authentication, data transferred between the OPSEC application and the firewall is encrypted using a 3DES key when you select 'ssl_opsec.' Select 'clear' to transfer data without any restrictions. After the parameters are entered, the SmartConnector configuration tool verifies whether it can connect the LEA Server from the LEA Client (the SmartConnector). If the verification fails, messages are displayed on the configuration window to assist in fixing the problem. Send the connector log files (under '\logs' folder) to HP Customer Support if you still cannot fix the problem. There must be consistency between the connection type selected when the SmartConnector was installed and the type specified in the 'fwopsec.conf' file because the SmartConnector is certified by Check Point. For Check Point OPSEC NG, the default connection is 'sslca.'
server_ip	Enter the IP address of the Check Point Management Server or the address where the LEA Server resides. USE THE IP ADDRESS OF THE MANAGEMENT SERVER, NOT ITS HOST NAME. Sometimes a dedicated Check Point Log Server handles the events from Check Point Management Servers. In this case, the Log Server device is the one to which the SmartConnector should talk.
server_port	Enter the number of the port to which the LEA Server is listening and that will be used for the communication with the LEA Client (the default value is 18184). Use tools such as 'telnet' or 'netstat' to verify whether Check Point Management Server or firewall is listening on the port specified for this SmartConnector.
opsec_sic_name	sslca and ssl_opsec connections: SIC means Secure Internal Communication and SIC Names are the Distinguished Name for Check Point modules or components. The OPSEC SIC Name is the SIC Name of the OPSEC application (LEA client) and the OPSEC Entity SIC Name is the SIC Name of the OPSEC server (in this case, the LEA server). See "Obtain the OPSEC SIC Name and OPSEC Entity SIC Name".
opsec_sscla_file	Name of the certificate file. This is used only for the 'sslca' connection type. Note: Only the file name is needed if the file is located at: $ARCSIGHT_HOME/current/user/agent/checkpoint. If not, the full file path for the sslca file is required.
opsec_entity_sic	Management Server's SIC names; used for 'ssl_opsec' and 'sslca' connection types. See the "Obtain the OPSEC SIC Name and OPSEC Entity SIC Name" section.

Note that the parameters that are required differ depending upon the connection type you select. The default connection type for Check Point OPSEC NG is SSLCA. For the CLEAR connection type, only the 'server_ip' and 'server_port' parameters are required. 'opsec_sic_name' and 'opsec_entity_sic_name' are required for SSLCA and SSL_OPSEC connections. 'opsec_sslca_file' is required only for SSLCA connections.

Select a Destination

1. The next window asks for the destination type; select a destination and click **Next**. For information about the destinations listed, see the *ArcSight SmartConnector User Guide*.

2. Enter values for the destination. For the ArcSight Manager destination, the values you enter for **User** and **Password** should be the same ArcSight user name and password you created during the ArcSight Manager installation. Click **Next**.

3. Enter a name for the SmartConnector and provide other information identifying the connector's use in your environment. Click **Next**. The connector starts the registration process.

4 If you have selected ArcSight Manager as the destination, the certificate import window for the ArcSight Manager is displayed. Select **Import the certificate to the connector from destination** and click **Next**. (If you select **Do not import the certificate to connector from destination**, the connector installation will end.) The certificate is imported and the **Add connector Summary** window is displayed.

Complete Installation and Configuration

1 Review the **Add Connector Summary** and click **Next**. If the summary is incorrect, click **Previous** to make changes.

2 The wizard now prompts you to choose whether you want to run the SmartConnector as a stand-alone process or as a service. If you choose to run the connector as a stand-alone process, select **Leave as a standalone application**, click **Next**, and continue with step 5.

3 If you chose to run the connector as a service, with **Install as a service** selected, click **Next**. The wizard prompts you to define service parameters. Enter values for **Service Internal Name** and **Service Display Name** and select **Yes** or **No** for **Start the service automatically**. The **Install Service Summary** window is displayed when you click **Next**.

4 Click **Next** on the summary window.

5 To complete the installation, choose **Exit** and Click **Next**.

For instructions about upgrading the connector or modifying parameters, see the *SmartConnector User Guide*.

Run the SmartConnector

SmartConnectors can be installed and run in stand-alone mode, on Windows platforms as a Windows service, or on UNIX platforms as a UNIX daemon, depending upon the platform supported. On Windows platforms, SmartConnectors also can be run using shortcuts and optional Start menu entries.

If the connector is installed in stand-alone mode, it must be started manually and is not automatically active when a host is restarted. If installed as a service or daemon, the connector runs automatically when the host is restarted. For information about connectors running as services or daemons, see the *ArcSight SmartConnector User Guide*.

To run all SmartConnectors installed in stand-alone mode on a particular host, open a command window, go to `$ARCSIGHT_HOME\current\bin` and run: `arcsight connectors`

To view the SmartConnector log, read the file `$ARCSIGHT_HOME\current\logs\agent.log`; to stop all SmartConnectors, enter `Ctrl+C` in the command window.

Device Event Mapping to ArcSight Fields

The following section lists the mappings of ArcSight data fields to the device's specific event definitions. See the *ArcSight Console User's Guide* for more information about the ArcSight data fields.

CheckPoint OPSEC NG Advanced Security Log Mappings

ArcSight ESM Field	Device-Specific Field
Application Protocol	fs_proto
Base Event Count	event_count
Bytes In	One of (client_inbound_bytes, server_inbound_bytes)
Bytes Out	One of (client_outbound_bytes, server_outbound_bytes)
Destination Address	One of (dst, Dst)
Destination Host Name	One of(endpoint_addr, Destination DHCP Hostname, Destination DNS Hostname, NetBIOS Destination Hostname)
Destination Mac Address	MAC Destination Address
Destination Port	d_port
Destination Service Name	One of (service, Service_name, app_name, service_id)
Destination Translated Address	xlatedst
Destination Translated Port	xlatedport
Destination User ID	user_uri
Destination User Name	One of (orig_to, user, vpn_user, uname4domain, d_name, to)
Device Action	One of (action, scan result, spyware_action, alert)
Device Address	One of (orig, endpoint_ip)
Device Custom Date 1	One of (local_time, (start_time, elapsed))
Device Custom Date 2	subs_exp (Subs Expired)
Device Custom IPv6 Address 2	Source IPv6 Address
Device Custom IPv6 Address 3	Destination IPv6 Address
Device Custom Number 1	One of (Payload, __hourMinuteSecondsToSeconds(elapsed))
Device Custom Number 2	One of (icmp_type, ICMP Type)
Device Custom Number 3	One of (icmp_code, ICMP Code)
Device Custom String 1	One of (virus_name, virus name, spyware_name, both (rule, rule_name))
Device Custom String 2	One of (Sensor Mode, integrity_av_invoke_type, category, categories)
Device Custom String 3	One of (Source OS, user_group, policy_id_tag)
Device Custom String 4	One of (treatment_result, spyware_status, Destination OS, scan result, rule_uid)
Device Custom String 5	One of (Vlan ID, bytes)
Device Custom String 6	One of (policy_name, policy_id_tag, policy)
Device Direction	i/f_dir
Device Event Category	One of (event_type, 'SecurityLog')
Device Event Class ID	One of(Both('authcrypt', one of (ICMP, reason, reason:, sys_msgs, message_info, IKE log:, encryption_failure, decryption failure, decryption_failure, success reason:, Reason:, Instruction:, wam_result, req_result,Attack Info, cluster_info, sync_info, description, request_info, Validation log:, message, GTP message info, System Alert message, log_unification_error)),one of (attack, action, spyware_action, alert, integrity_av_event, message, description, short_desc, dynamic object))
Device Facility	One of (orig_name, product_family)
Device Host Name	One of (cat_server, peer_gateway, endpoint_addr, Hostname)
Device Inbound Interface	i/f_name (when i/f_dir is inbound)
Device Outbound Interface	i/f_name (when i/f_dir is outbound)
Device Product	One of (product, policy_id_tag,"product=(variable"),"No Product")
Device Receipt Time	time

ArcSight ESM Field	Device-Specific Field
Device Severity	One of (Attack Impact, Attack Nature, Severity, severity)
Device Vendor	'Check Point'
Device Version	One of (version, sig_ver)
End Time	start_time, elapsed
Event Outcome	One of (Attack Assessment, status)
External ID	uuid
File Name	One of (file_name, file name)
File Type	file_type
Message	One of (integrity_av_event, spyware_type, ICMP, reason, reason:, sys_msgs, message_info, IKE log:, encryption_failure, decryption failure, decryption_failure, success reason:, Reason:, Instruction:, wam_result, req_result, Attack Info, cluster_info, sync_info, request_info, Validation log:, message, GTP message info, System Alert message, log_unification_error, fde_details, action_comment, description, Description, fw_message, Internal_CA:, comment, failure_impact, log_sys_message)
Name	One of (attack, action, alert, integrity_av_event, spyware_name, message, event_type, description, Description, activity, short_desc, sync, sync info)
Request Context	HTTP Referer
Request Cookies	Cookie
Request Protocol	RPC Service Number
Request URL	One of (resource, url, URL, redirect_url, data origin)
Source Address	One of (src, Src)
Source Host Name	One of (end_user_address, NetBIOS Source Hostname, Source DHCP Hostname, Source DNS Hostname)
Source Mac Address	MAC Source Address
Source Port	s_port
Source Service Name	update_src
Source Translated Address	One of (xlatesrc, VPN internal source IP)
Source Translated Port	xlatesport
Source User Name	One of (orig_from, from, Email Address)
Start Time	start_time
Transport Protocol	proto

Check Point OPSEC Advanced Audit Log Mappings

ArcSight ESM Field	Device-Specific Field
Category Outcome	Audit Status ('Success=/Success', 'Failure=/Failure')
Destination Host Name	Machine
Destination User Name	Administrator
Device Action	action
Device Address	orig
Device Custom String 1	session_id
Device Custom String 2	Subject
Device Custom String 3	policy_id_tag
Device Custom String 5	ObjectName
Device Custom String 6	Policy Name
Device Direction	i/f_dir

ArcSight ESM Field	Device-Specific Field
Device Event Category	'AuditLog'
Device Event Class ID	Operation
Device Host Name	EndpointName
Device Inbound Interface	i/f name (when i/f_dir is inbound)
Device Outbound Interface	i/f name (when i/f_dir is outbound)
Device Product	One of (product, policy_id_tag)
Device Receipt Time	time
Device Service Name	app_name
Device Vendor	'Check Point'
External ID	Uid
Message	One of (Both (TCP packet out of state, tcp_flags), FieldsChanges, Additional Info)
Name	Operation
Source Address	client_ip

Check Point OPSEC NG Application Control Module Mappings

ArcSight ESM Field	Device-Specific Field
Bytes In	received_bytes
Bytes Out	sent_bytes
Destination Host Name	dst_machine_name
Destination User ID	UserCheck_incident_uid
Destination User Name	One of(dst_user_name,UserCheck)
Device Custom Date 1	browse_time
Device Custom String 1	app_rule_name
Device Custom String 2	app_category
Device Custom String 3	app_rule_id
Device Custom String 4	frequency
Device Custom String 5	user_status
Device Custom String 6	UserCheck_Confirmation_Level
Device Outbound Interface	UserCheck_Interaction_name
Event Outcome	Update Status
File ID	snid
Message	portal_message
Request Client Application	web_client_type
Source Host Name	src_machine_name
Source User Name	src_user_name

Check Point OPSEC NG Data Leakage Protection Module Mappings

ArcSight ESM Field	Device-Specific Field
Application Protocol	dlp_transport
Destination User Name	dlp_recipients
Device Custom String 1	dlp_rule_name

ArcSight ESM Field	Device-Specific Field
Device Custom String 3	incident_extension
Device Custom String 6	scan direction
Device Event Category	dlp_categories
External ID	dlp_rule_uid
File Name	dlp_file_name
Message	One of (dlp_violation_description, dlp_action_reason, portal_message, information)

Check Point OPSEC NG Anti-bot (Anti Malware) Module Mappings

ArcSight ESM Field	Device-Specific Field
Bytes In	received_bytes
Bytes Out	sent_bytes
Destination Host Name	dst_machine_name
Destination User Name	dst_user_name
Device Custom String 1	malware_rule_name
Device Custom String 2	Protection Type
Device Custom String 3	protection_id
Device Custom String 4	Protection Name
Device Custom String 6	scan direction
Device Event Class ID	One of(action, description, Description, short_desc, long_desc, Scan Summary)
Event Outcome	Update Status
File ID	snid
Name	One of(action, description, Description, short_desc, long_desc, Scan Summary)
Source Host Name	src_machine_name
Source User Name	src_user_name

Check Point OPSEC NG Identity Awareness Module Mappings

ArcSight ESM Field	Device-Specific Field
Device Custom String 1	connectivity_state
Device Custom String 2	identity_src
Device Custom String 3	identity_type
Device Custom String 4	termination_reason
Device Custom String 5	auth_method
Device Custom String 6	src_user_group
Device Event Category	ctrl_category
Device Version	client_version (R77)
File Path	src_machine_group (R77)
Message	One of (description,description ,error_description)
Name	auth_status
Reason	termination_reason (R77)
Request Client Application	client_name (R77)

ArcSight ESM Field	Device-Specific Field
Request Context	origin_sic_name (R77)
Source Host Name	src_machine_name
Source NT Domain	domain_name (R77)
Source User Name	src_user_name
Source User Privileges	roles

Check Point OPSEC NG URL Filtering Module Mappings

ArcSight ESM Field	Device-Specific Field
Bytes In	received_bytes
Bytes Out	sent_bytes
Destination Host Name	dst_machine_name
Destination User ID	UserCheck_incident_uid
Destination User Name	One of(dst_user_name,UserCheck)
Device Custom Date 1	browse_time
Device Custom Number 1	limit_requested
Device Custom Number 2	limit_applied
Device Custom String 1	app_rule_name
Device Custom String 2	app_category
Device Custom String 3	app_rule_id
Device Custom String 4	user_status
Device Custom String 5	Update Status
Device Custom String 6	UserCheck_Confirmation_Level
Device Outbound Interface	UserCheck_Interaction_name
Event Outcome	update status
File ID	snid
Message	portal_message
Request Client Application	web_client_type
Source Host Name	src_machine_name
Source User Name	src_user_name

Check Point OPSEC NG Anti-spam and Email Module Mappings

ArcSight ESM Field	Device-Specific Field
Destination Host Name	dst_machine_name
Destination User Name	dst_user_name
Device Custom Number 1	email_recipients_num
Device Custom Number 2	Log delay
Device Custom String 1	email_id
Device Custom String 2	email_message_id
Device Custom String 3	email_spool_id
Device Custom String 4	email_control
Device Custom String 5	email_session_id
Device Event Class ID	One of(action,alert,email_control)

ArcSight ESM Field	Device-Specific Field
Message	email_control_analysis
Name	One of(action,alert,email_control)
Source Host Name	src_machine_name
Source User Name	src_user_name

Check Point OPSEC NG IPS Module Mappings

ArcSight ESM Field	Device-Specific Field
Destination Host Name	dst_machine_name
Destination User Name	dst_user_name
Device Custom Floating Point 1	Update Version
Device Custom Number 1	Update Version
Device Custom Number 2	during_sec
Device Custom Number 3	fragments_dropped
Device Custom String 1	voip_log_type
Device Custom String 2	Protection Type
Device Custom String 3	protection_id
Device Custom String 4	TCP flags
Device Custom String 5	content_type
Device Custom String 6	Protection Name
File ID	snid
Message	One of(Message,Summary)
Request Client Application	web_client_type
Source Host Name	src_machine_name
Source User Name	src_user_name

Additional Notes

This section describes some common tasks you may need to understand when installing and configuring the SmartConnector.

Verifying Check Point1 Lets the Connector Box Pass Through

ArcSight recommends installing the SmartConnector on a machine other than the Check Point Management Server or host. Make sure that the firewall does not block the SmartConnector from retrieving events by checking the rules on the Check Point Policy Editor or Check Point SmartDashboard.

Making Sure to Set Rules to Track Events

For rules on Check Point Policy Editor or SmartDashboard, select the **Log** option for the rules whose events you want the SmartConnector to retrieve. Otherwise, you may not see any Check Point FW-1 events coming through to the ArcSight Console, even if there are new events appearing in the Check Point LogViewer or SmartTracker.

Adapting HF1 or Later HotFix Patches for Check Point FP3

If Check Point Feature Pack 3's automatic log rotation function is used, make sure that the Check Point Feature Pack HotFix1 or later patch has been applied. Otherwise, there is a known bug in which the Check Point Management Server stops sending events to third-party applications after the log is switched.

Making Sure the C/C++ lea_client in UNIX has Adequate Privilege

As previously described, the **lea_client** is a C/C++ application included in SmartConnector that is dedicated to collecting Check Point firewall events. The application is located in the folder:

```
<$ARCSIGHT_HOME>\current\bin\agent\checkpoint\[OPSEC|OPSECNG]\[platform\]
```

When you start the connector on a UNIX machine, especially when running the connector as a daemon, make sure that the current user account has the correct execution permission for this executable file.

Troubleshooting

I receive the error message "The Program can't start because MSVCR110.dll is missing from your computer."

The Microsoft Visual Studio C++ redistributable dll is needed for projects built with Visual Studio. You can download this redistributable from the Microsoft website: http://www.microsoft.com/en-us/download/details.aspx?id=30679#.

Download x86 edition for 32-bit platforms.

How do I resolve an incorrect IP address connection error when attempting to pull the certificate?

For certificates to be pulled from the Check Point by a firewall module from a management module, the CPD process on the management module must be listening on TCP port 18210. Ensure the TCP port is specified correctly.

Check Point Connector Stops Receiving Events After a Period of Time

Obtain the following hotfix from Check Point for this issue:

Check Point Hotfix SK98588 "Log Server stops forwarding logs to LEA clients: SmartEvent, SmartReporter, OPSEC clients"

Check Point OPSEC NG connector fails to connect to LEA Server due to missing dll files needed for lea_client.exe

If you receive an error message that the connector cannot connect to the LEA server, the `msvcp71.dll` and `msvcr71.dll` files cannot be found.

To rectify this problem, download the Visual C++ Redistributable 2012 Updated 4 in 32-bit (vcredist_x86.exe) from https://www.microsoft.com/en-us/download/details.aspx?id=30679. This file

automatically extracts and copies the correct 32-bit msvcr110.dll and supporting files to be able to run the lea_client.exe.

Restart the connector.

If the dll files are still not found, copy them into the same directory in which `lea_server.exe` is located `($ARCSIGHT_HOME\current\bin\agent\checkpoint\OPSECAD\win32)`.

For more information, follow this link for Microsoft's recommendations for resolving this error: http://answers.microsoft.com/en-us/windows/forum/windows_7-performance/err-msvcr71dll-missing-reinstall-win-fix/af0ef1f9-e0bd-4947-9660-deeff6149680.

When upgrading from a previous version to the current SmartConnector version, the Check Point service stopped running; how can I fix this error?

In some instances after upgrade, the Check Point service may stop running and indicate that the MSVCR75.dll or MSVCP75.dll file was not found. If you receive this type of error, you can manually download the missing DLL file or files to `c:\WINDOWS\system32` for supported Windows 32-bit systems to fix this problem. If the dll files are still not found, copy them into the same directory in which `lea_server.exe` is located `($ARCSIGHT_HOME\current\bin\agent\checkpoint\OPSECAD\win32)`.

How do I resolve the error "./opsec_pull_cert: error while loading shared libraries: libpam.so.0: cannot open shared object file: No such file or directory"?

To resolve this problem, download the missing library to `$ARCSIGHT_HOME/ArcSightSmartConnectors/current/bin/agent/checkpoint/OPSECAD/linux` or add the path to the location of this library to the `LD_LIBRARY_PATH`.

Fixing Error: Error while loading shared libraries: libcpc++-libc6.1-2.so.3

If you get this error message when executing applications such as **opsec_putkey** or **opsec_pull_cert** in Linux environments, you can fix the problem by changing or creating the environment variable LD_LIBRARY_PATH, as follows. This example assumes that the ArcSight SmartConnector is installed in the folder /opt/arcsight/chkpoint.

```
export
LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/opt/arcsight/chkpoint/current/bin/agent/
checkpoint/OPSECAD/linux
```

Or, if the LD_LIBRARY_PATH does not yet exist:

```
export
LD_LIBRARY_PATH=/opt/arcsight/chkpoint/current/bin/agent/checkpoint/OPSECAD/
linux
```

Executing lea_client Under OPSEC Debug Mode

If you do not see events in the ArcSight Console from a properly installed SmartConnector for Check Point OPSEC NG, there may be problems with the connection between the LEA client (the SmartConnector) and the LEA server (Check Point).

For example, in the SmartConnector command line window or in the `agent.out.wrapper.log` or `agent.log` files, you may see an error message such as `SG14 End OPSEC opsec_mainloop`. In this case, you can trace the reason for the communication failure by executing the lea_client binary under OPSEC debug mode. For this example, assume that the ArcSight SmartConnector is installed on Windows in the folder `C:\$ARCSIGHT_HOME` and that the LEA server is listening on port 18184 with connection type `clear`. Run in debug mode by executing these commands:

```
C:\>cd C:\$ARCSIGHT_HOME\current\bin\agent\checkpoint\OPSECAD\win32
C:\$ARCSIGHT_HOME\current\bin\agent\checkpoint\OPSECAD\win32>set
OPSEC_DEBUG_LEVEL=3
C:\$ARCSIGHT_HOME\current\bin\agent\checkpoint\OPSECAD\win32>lea_client -d
on -m online -t clear -h 'LEA server IP Address' -p 18184
```

Let the LEA client run for several minutes, and then press Ctrl+C to stop it. Send debug output to ArcSight Customer Support.

The export command generates the output as standard error, not standard output; therefore, to capture this, do not use `output.txt` as the output will be empty. Instead, to capture these to file, use the following:

```
CSH: command> & output.txt
BSH: command> output.txt
Windows: command 2> output.txt
```

When the lea_client cannot connect to the lea server

Be sure to contact Check Point to obtain the latest Hotfix.

When lea_client cannot connect to the lea server and is receiving some debug messages such as:

```
client: got LOGTRACK for session
Destroying session (bfa838) id 5 (ent=bd8ad8)
reason=PEER_ENDED
```

or

```
Destroying session (bfa838) id 5 (ent=bd8ad8)
reason=END_BY_APPLICATION
```

Consider fixing the problem by cleaning up the log files under `<CheckPoint Product Installation Directory>`\log. There may be too many log files.