Log Correlation Engine OPSEC Client 4.5.0 Guide

December 11, 2018

Table of Contents

Introduction

This document describes the 4.5.x OPSEC client that is available for Tenable Inc's **Log Correlation Engine 4.8**.x Please email any comments and suggestions to support@tenable.com.

A working knowledge of Secure Shell (SSH), Log Correlation Engine (LCE), and SecurityCenter operation and architecture is assumed. Familiarity with general log formats from various operating systems, network devices and applications, as well as a basic understanding of Linux/Unix is also assumed.

This document describes the current LCE server (daemon) version of 4.8.x as it is used with the LCE 4.5.x OPSEC Client.

Standards and Conventions

Throughout the documentation, filenames, daemons, and executables are indicated with a **`courier bold`** font such as **`gunzip`**, **`httpd`**, and **`/etc/passwd`**.

Command line options and keywords are also indicated with the **`courier bold`** font. Command line examples may or may not include the command line prompt and output text from the results of the command. Command line examples will display the command being run in **`courier bold`** to indicate what the user typed while the sample output generated by the system will be indicated in `courier` (not bold). Following is an example running of the Linux/Unix **`pwd`** command:

```
# pwd
/opt/lce/
#
```

Important notes and considerations are highlighted with this symbol and grey text boxes.

Tips, examples, and best practices are highlighted with this symbol and white on blue text.

Log Correlation Engine OPSEC Client

The process to setup and configure the LCE OPSEC Client begins with the connection of a Check Point Log Extraction API (LEA) application, such as LCE OPSEC Client, to a Check Point Security Management Server. The steps outlined here apply to the Check Point VPN-1 version R80. The initial environment assumed by this article includes the following:

- A Check Point Security Management Server is installed with the proper license and is running without any configuration issue.
  - The Security Management Server should be accessed using SSH to get what we will call the Configuration Prompt (which is not a regular terminal).
- A Check Point SmartDashboard client is available to configure the Security Management Server. We will refer to the server that hosts the SmartDashboard client as the Dashboard Server.
  - The Dashboard Server must be accessed through a user interface (not SSH), such as Remote Desktop Connection.

- The Check Point OPSEC SDK has been downloaded and extracted on the machine running the LCE OPSEC Client. Upon downloading the OPSEC SDK 6.1 for Linux, the required utilities are in **`OpsecSicUtils_linux50.tar`**. The OPSEC SDK 6.1 for Linux can be found here.

- The LCE OPSEC Client RPM package has been downloaded and a RHEL6 64-bit host is available on which to install the client. The LCE OPSEC Client may also be referred to as the LEA Application, because it uses the OPSEC LEA to pull logs from an OPSEC-compliant device.

The major steps include:

- Setting up an authenticated LEA connection on the Check Point Security Management Server
- Adding the LCE OPSEC Client host in the SmartDashboard
- Creating keys on the Check Point Security Management Server
- Configuring the LEA Server and LCE OPSEC client application

Setting up Authenticated LEA Service on the Check Point Security Management Server

1. Log in to the Security Management Server via SSH as admin (not root).
2. From the configuration prompt (see below), go into expert mode with the command "**`expert`**", which requires a password.
3. Change directory to **`$FWDIR/conf`**.
4. Open the **`fwopsec.conf`** file in an editor such as vi.

```
login as: admin
admin@192.168.2.32's password:
Last login: Sun Oct 12 19:10:41 2014 from 192.168.2.124

? for list of commands
sysconfig for system and products configuration

# expert
Enter expert password:

You are in expert mode now.

# cd $FWDIR/conf
# vi fwopsec.conf
```

5. Edit the file **`fwopsec.conf`**. This file configures the port and authentication settings for the various server types (SAM, LEA, ELA, CPMI, UAA, etc.). For LEA configuration, look for the lines starting with the text "lea_server". Update or add the highlighted settings below to enable an authenticated LEA connection on port 18185:

```
#
# (c) Copyright 1993-2008 Check Point Software Technologies Ltd.
# All rights reserved.
#
```

```
# This is proprietary information of Check Point Software Technologies
# Ltd., which is provided for informational purposes only and for use
# solely in conjunction with the authorized use of Check Point Software
# Technologies Ltd. products.  The viewing and use of this information is
# subject, to the extent appropriate, to the terms and conditions of the
# license agreement that authorizes the use of the relevant product.
#
# This file, by default, has no active entries.
# From this Check Point version, the purpose of this opsec
# configuration file is:
# 1) To present the default configuration of security methods
#    and port numbers of the OPSEC servers within Check Point.
# 2) To allow the administrator to change those defaults.
# 3) To configure non-default security methods for OPSEC client products.
#
# Note:
# To change the security method with old OPSEC server products, use the
# Policy-Editor to edit the backwards-compatibility method of the ufp/cvp
# servers.
#
# The format of a configuration entry is:
#  <opsec-server>  <security-method>  <port-number>
#
# Where:
#  <opsec-server> is one of:
#      sam_server, lea_server, ela_server, cpmi_server, uaa_server.
#  <security-method> is one of:
#      auth_port: The OPSEC server listens to secure connections
#                 on the following port number.
#      port:      The OPSEC server listens to clear connections
#                 on the following port number.
#  <port> is an integer port number:
#      0:   Means that this security method is disabled.
#      >0:  Indicates the port number for this security method.
#
# It is possible that a specific OPSEC server will listen to both
# secure and clear connections (on two different ports!).
# the 'clear' security method may only be used for connections with
# OPSEC products using an OPSEC SDK version 4.1.2 and below.
#
# To change the default setting of an entry:
#   a. Remove the comment sign (#) at the beginning of the line.
#   b. Change the port number.
#
# The VPN-1/FireWall-1 default settings are:
#
#  sam_server  auth_port   18183
#  sam_server       port       0
#
#  lea_server  auth_port   18184
#  lea_server       port       0
#
#  ela_server  auth_port   18187
#  ela_server       port       0
#
#  cpmi_server auth_port   18190
```

```
#
#  uaa_server  auth_port   19191
#  uaa_server       port       0
#
lea_server auth_port 18185
lea_server port 0
lea_server auth_type sslca
```

> Make note of the **`auth_port`** (18185 in this example) and the IP address of the Check Point Security Management Server (192.168.2.32 in this example).
>
> They will be placed into the <fw1-port> and <fw1-server> tags, respectively, when the LCE OPSEC Client policy is created.

The first line, "**`lea_server auth_port`**", turns on LEA authenticated connection on port 18185. The second line, "**`lea_server port 0`**", turns off LEA unauthenticated connection. If these lines are not present in the **`fwopsec.conf`** file, add them. Save the file and exit the editor. After the **`fwopsec.conf`** file is updated, the firewall service must be restarted. Use the following commands to stop, and then start the server:

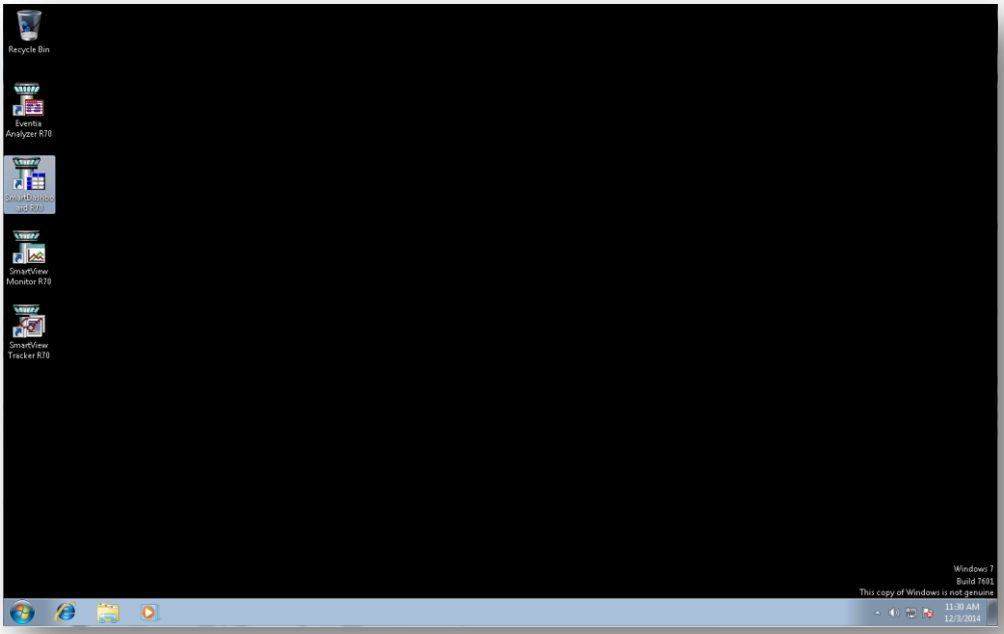
```
# cpstop
SmartPortal: Stopping CPWMD
cpwd_admin:
Process CPWMD terminated
SmartPortal: Stopping CPHTTPD
cpwd_admin:
Process CPHTTPD terminated
Stopping SmartView Monitor daemon ...
SmartView Monitor daemon is not running
Stopping SmartView Monitor kernel ...
Driver is Down.
rtmstop: SmartView Monitor kernel is not loaded

# cpstart
cpstart: Power-Up self tests passed successfully

cpstart: Starting product - SVN Foundation

SVN Foundation: cpWatchDog already running
SVN Foundation: cpd already running
SVN Foundation started

cpstart: Starting product - VPN-1

FireWall-1: starting external VPN module -- OK
FireWall-1: Starting fwd
```

The LEA service is now enabled and running on the Check Point device.

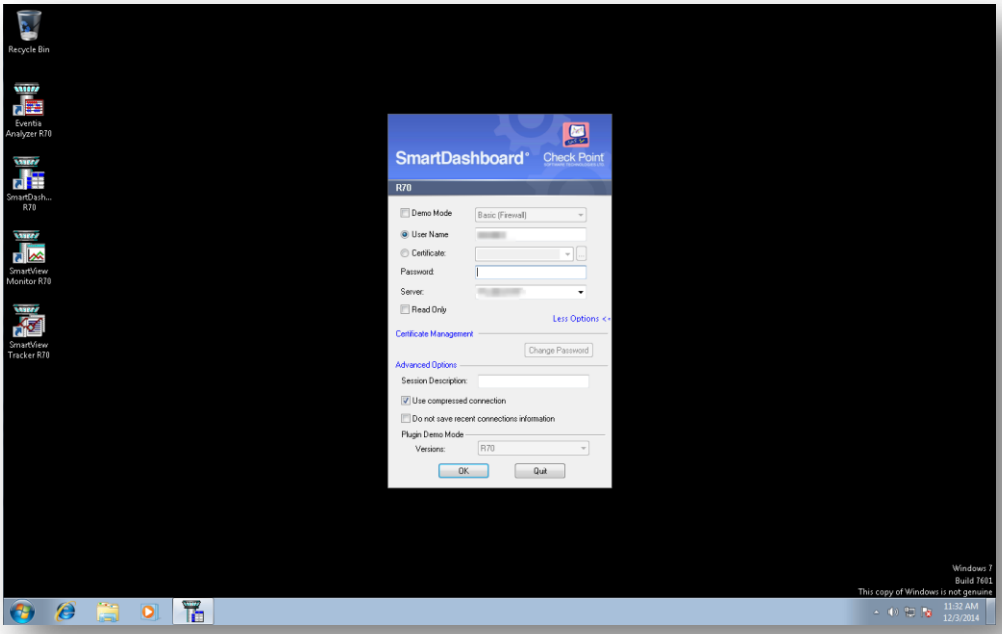
Adding the LEA Application in SmartDashboard

SmartDashboard is an application that manages the security policies and rules for Check Point's VPN-1/FireWall-1 gateways. You will need to add the LCE OPSEC Client information to the SmartDashboard using the steps below:

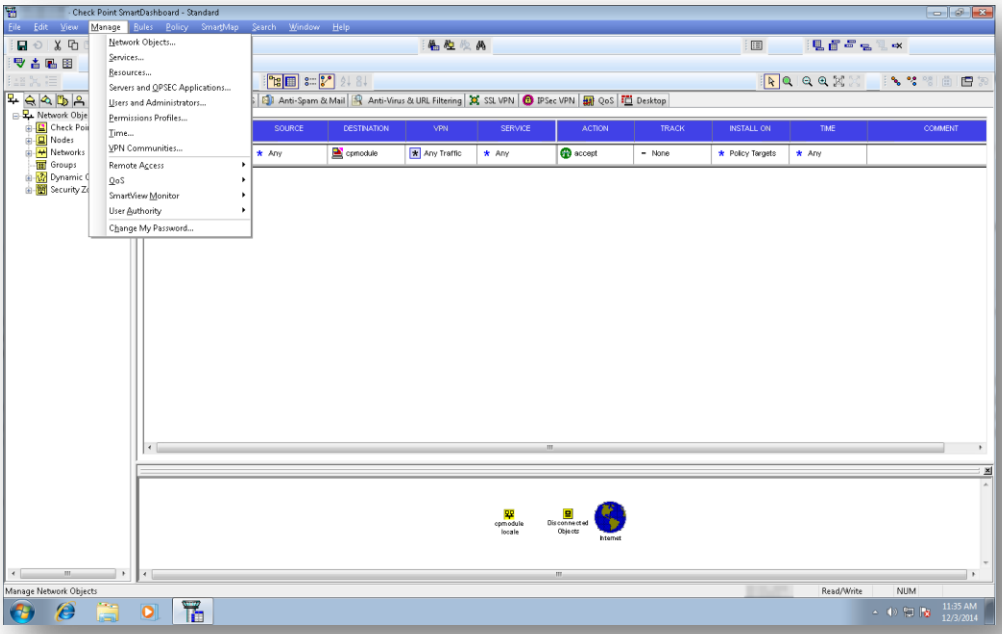
1. Connect to the Dashboard Server's desktop and start the SmartDashboard application.



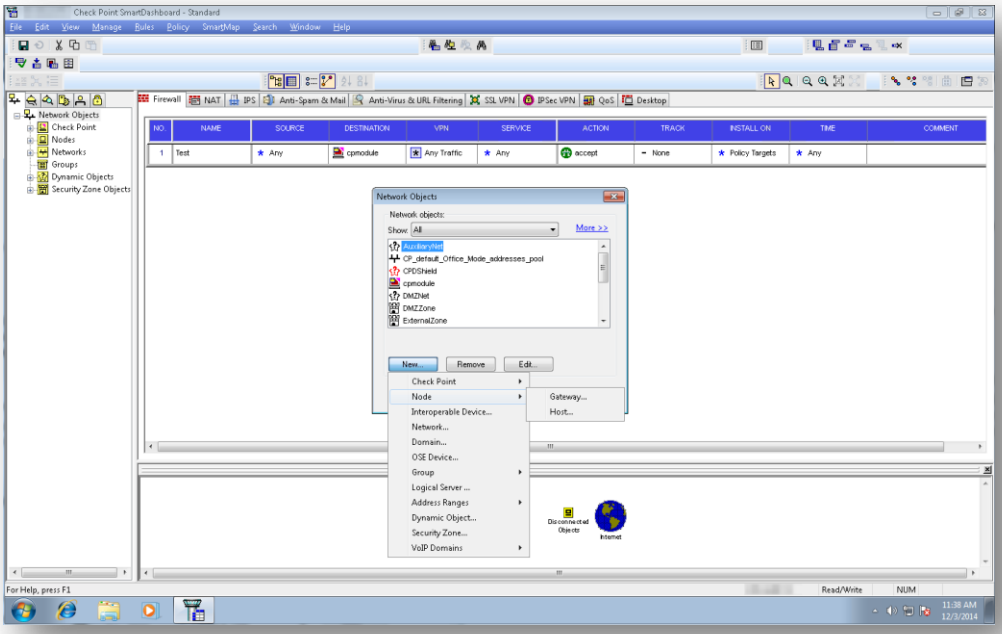
2. Connect to the Security Management Server using the SmartDashboard username. Note: This is not the same user that logs into the Security Management Server via SSH; it is a SmartDashboard administrative user, not a user on the Security Management Server.



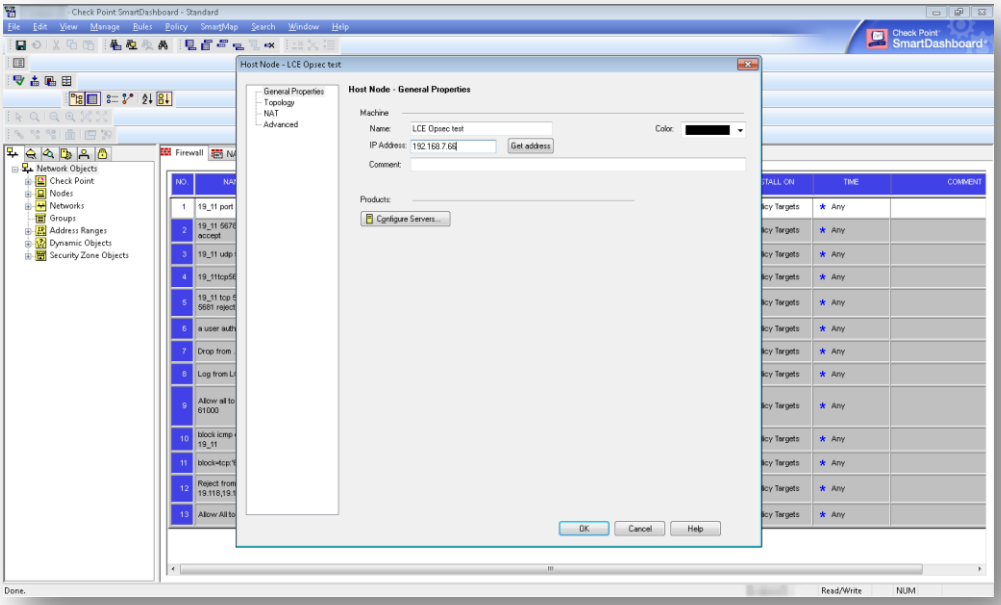
3. Click the menu item Manage -> Network Objects... to bring up the "Network Objects" dialog.



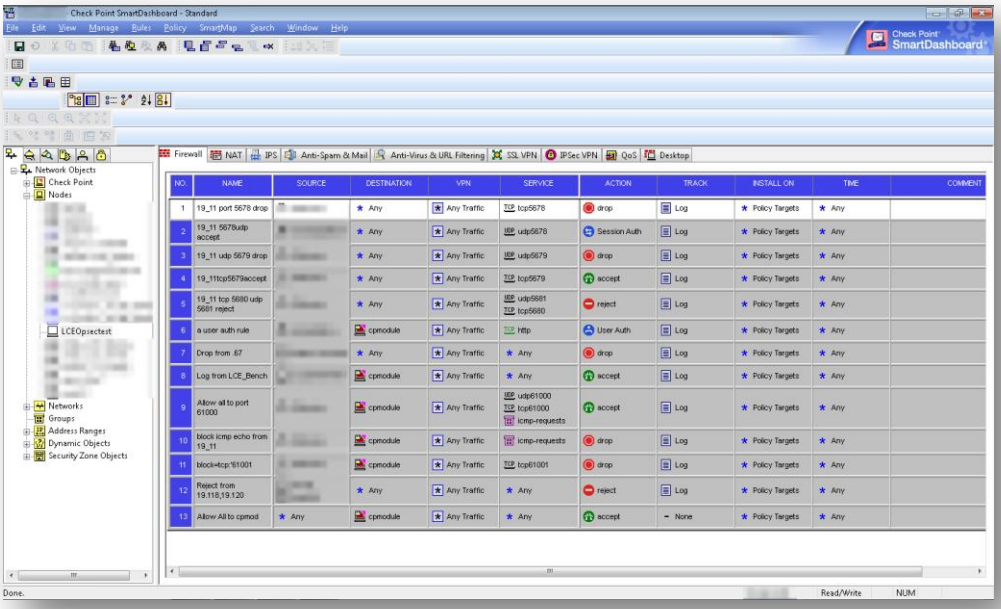
4. In the "Network Objects" dialog, select New... -> Node -> Host... to enter new host node information.



5. In the "New Node" window, fill in the "Name" and "IP Address" using the hostname and IP address of the LCE OPSEC Client host. Click "OK" when done. Select "Close" on the "Network Objects" window.

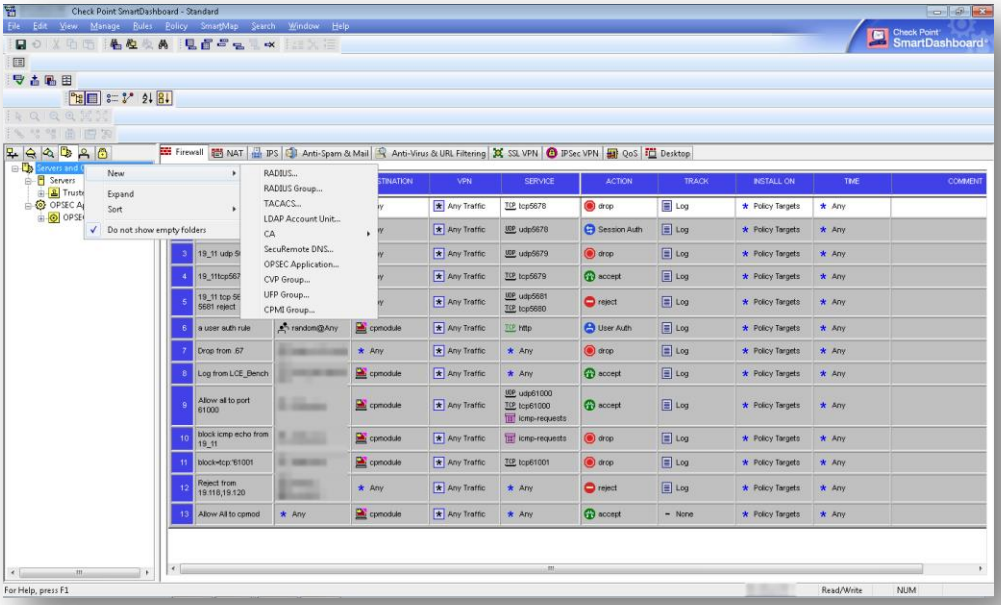


6. In the SmartDashboard locate the "Objects Tree" window. The objects tree can be displayed or hidden using the menu item View -> Objects Tree. Verify the new node is in the Objects Tree tab "Network Objects", under "Nodes".

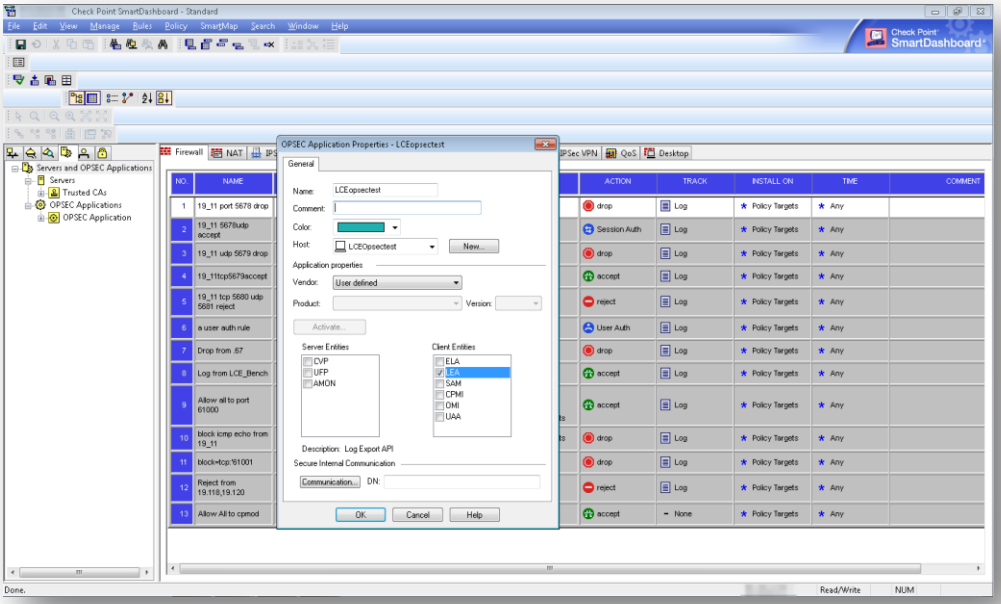


The next series of steps is to tell the Security Management server about the LCE OPSEC Client (as opposed to the LEA server).

1. In the Objects Tree window (see step six directly above), click on the “Servers and OPSEC Applications” tab. In this tab view, locate the item “Servers and OPSEC Applications” -> “OPSEC Application”. Right-click this item and select “New OPSEC Application...”. This will bring up the OPSEC Application Properties dialog.

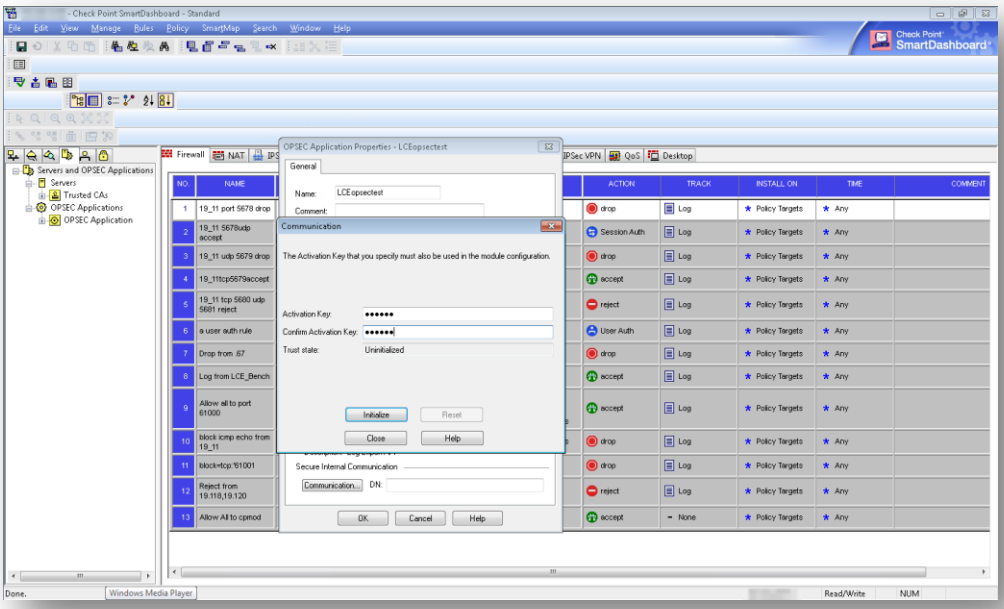


2. Fill in the OPSEC Application Properties dialog (shown below) with information about the LCE OPSEC Client host. Enter the LCE OPSEC Client hostname under “Name”, and choose the correct host from the “Host” drop-down box. Under “Client Entities” select the “LEA” checkbox. This step describes the LCE OPSEC Client to the SmartDashboard and the Security Management Server.



To set up an authenticated connection, the section “Secure Internal Communication” (SIC) must be completed:

1. Click on the “Communication” button of the OPSEC Application Properties dialog shown above. This will bring up the SIC Dialog, which requests an “Activation Key” that is created by the user. This is also referred to as the “SIC Password” and will be used again later on.

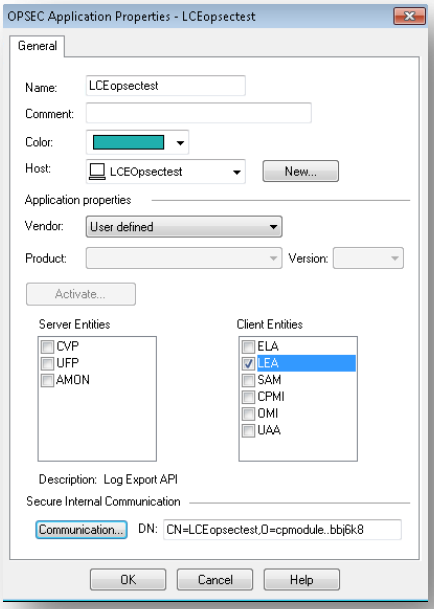


2. In the SIC Dialog (shown above), enter an activation key that is created by the user for the LEA connection.

3. Click “Initialize” to accept the key. The “Trust state” value should show “Initialized but trust not established”. This is OK at this time. The value will change to “Trust established” when the certificate is pulled by the LEA application.

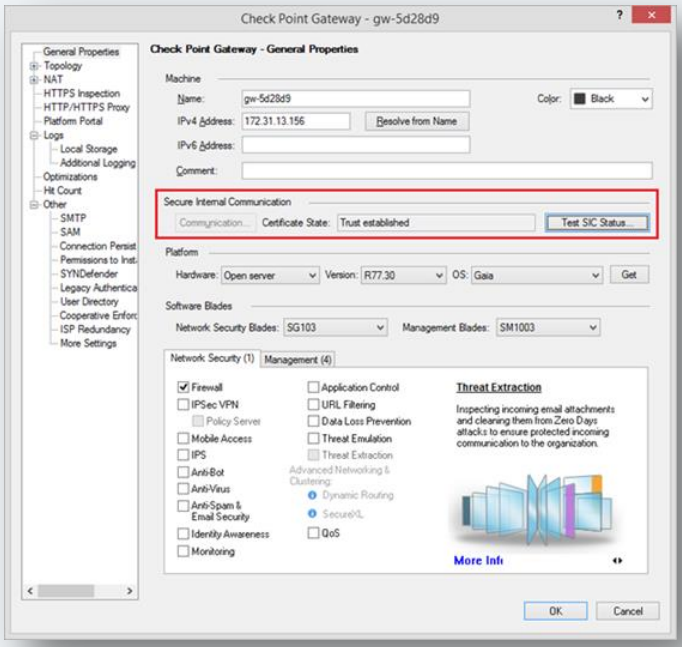
4. Click “Close” to close the “Communication” dialog.

5. Copy down the DN field from the “OPSEC Application Properties” dialog. This value is referred to as the OPSEC client DN and will be used by the LCE OPSEC client in the <opsec-client-dn> policy element.



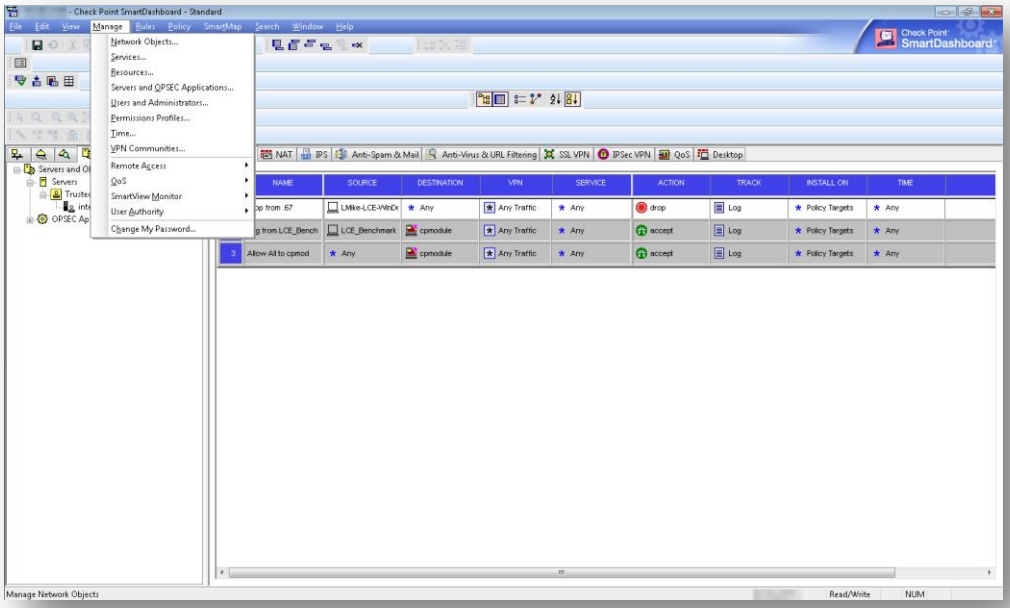
Make note of the "Name" in the "OPSEC Application Properties" window it will be used later in conjunction with the `opsec_pull_cert` command.

If you are using an R80 SmartDashboard, to obtain the SIC DN, in the **Check Point Gateway – General Properties** section, click the **Test SIC Status…** button.

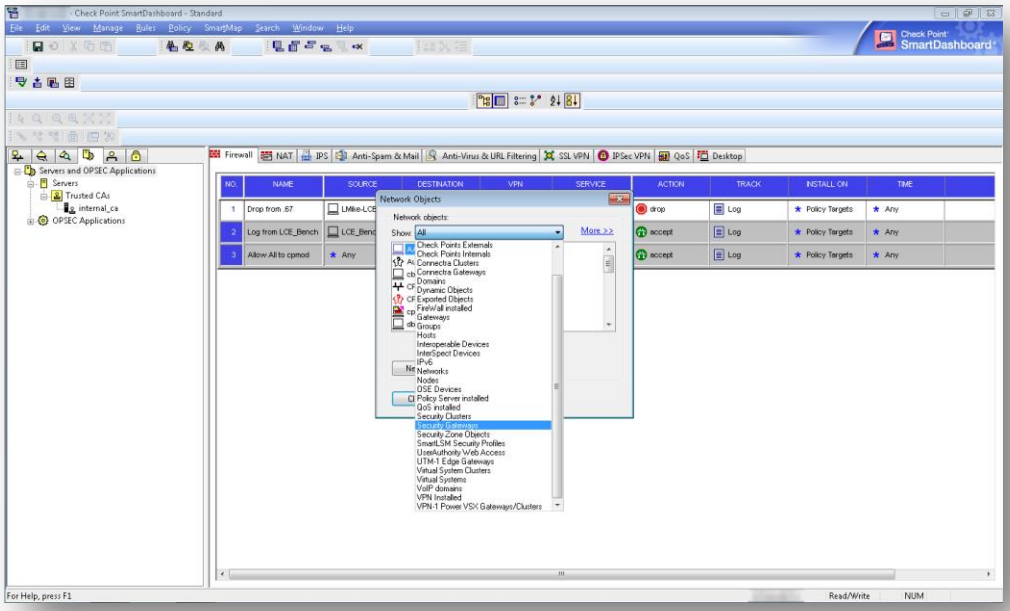


There is another DN needed by the LCE OPSEC Client that is referred to as the OPSEC server DN. This is the DN of the Check Point server. To get this value,

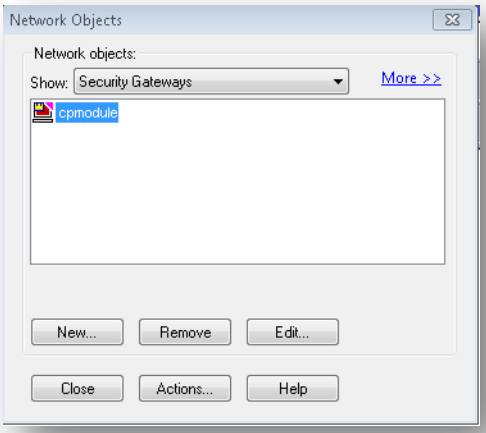
1. Click the menu item Manage -> Network Objects... to bring up the "Network Objects" dialog.



2. In the "Network Objects" dialog, set the drop-down filter labeled "Show" to "Security Gateways" to identify the Security Management Server machine.



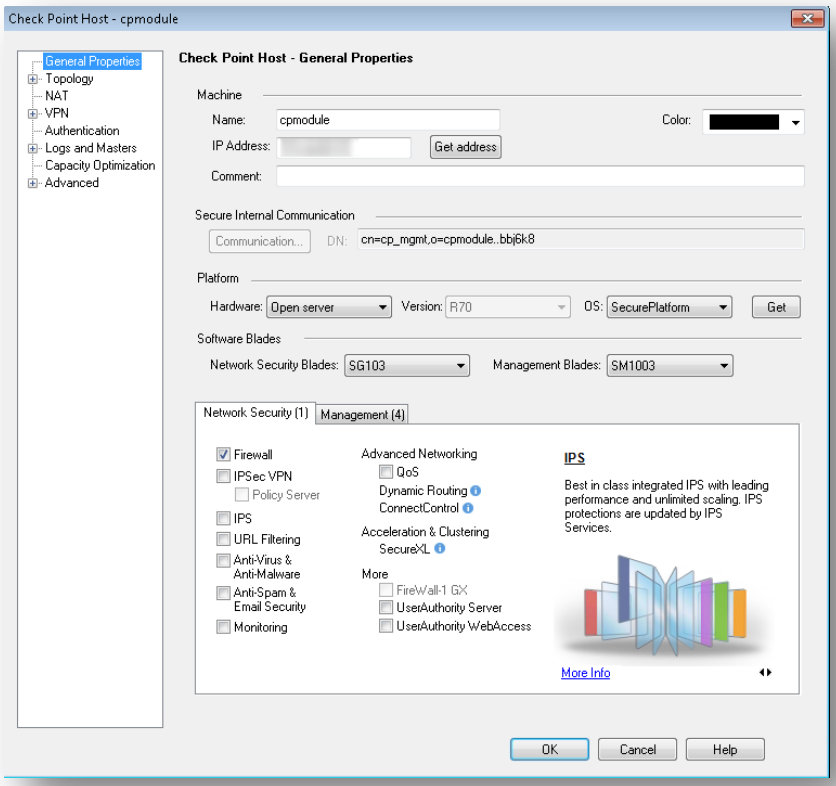
3. Select the Security Management Server machine and click “Edit…” to bring up the “Check Point Host” dialog.



4. In the “Check Point Host” dialog,

    1. Verify that the IP address matches that of the Security Management Server, to ensure you will get the correct DN.

    2. Locate the “Secure Information Communication” (SIC) section and copy down the OPSEC Server DN value to be used later in the LCE OPSEC Client policy file.



3. Click “Cancel”.

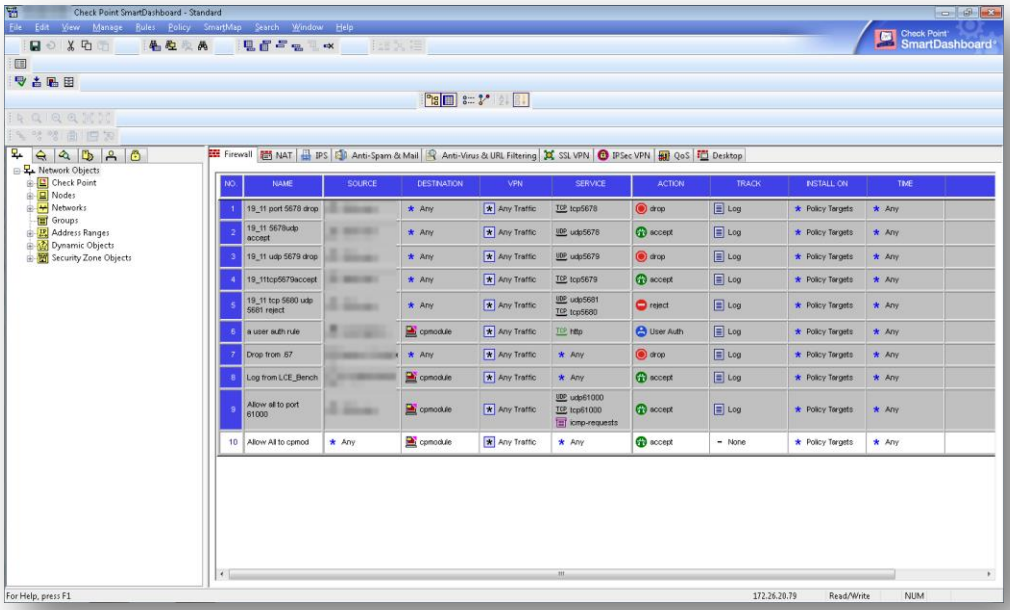
5. Close the Network Objects dialog.

The OPSEC Server DN value isn’t visible in some versions of the Check Point SmartDashboard. Using the **`cpca_client lscert -kind SIC`** command on the Security Management Server will also show the OPSEC Server DN value, as shown below:

```
# expert
Enter expert password:

You are in expert mode now.

# cpca_client lscert -kind SIC
Operation succeeded. rc=0.  1 certs found.

Subject = CN=cpmodule,O=cpmodule..bbj6k8
Status = Valid   Kind = SIC   Serial = 56267
Not_Before: Mon Oct  8 21:51:52 2012   Not_After: Mon Jan 11 20:43:04 2038
```

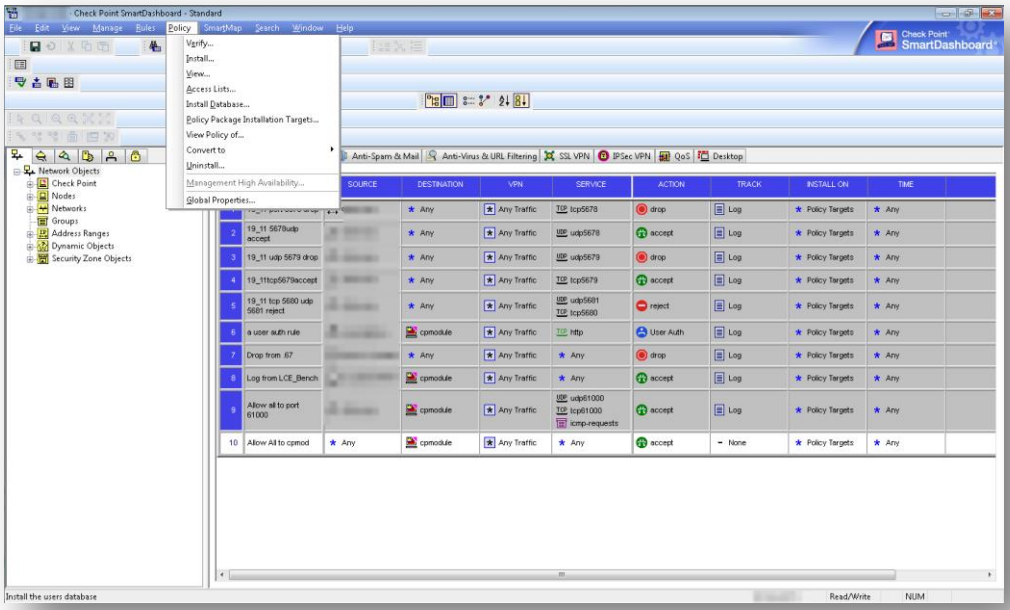
Adding Rules to the SmartDashboard

In the SmartDashboard, add a rule to allow the LEA Server to connect to the Security Management Server on the LEA port (18185 in preceding sections). Add the rule to the Firewall tab of the “Rule Base” window. Below is an example of the rules table on the Firewall tab of the “Rule Base” window. If a rule already exists to allow any traffic to and from the cpmodule, no additional rule should be required. It is recommended that the rule tracking is not set to “Log”, because this would create additional logs for each instance of the LCE OPSEC Client connection to the OPSEC interface.

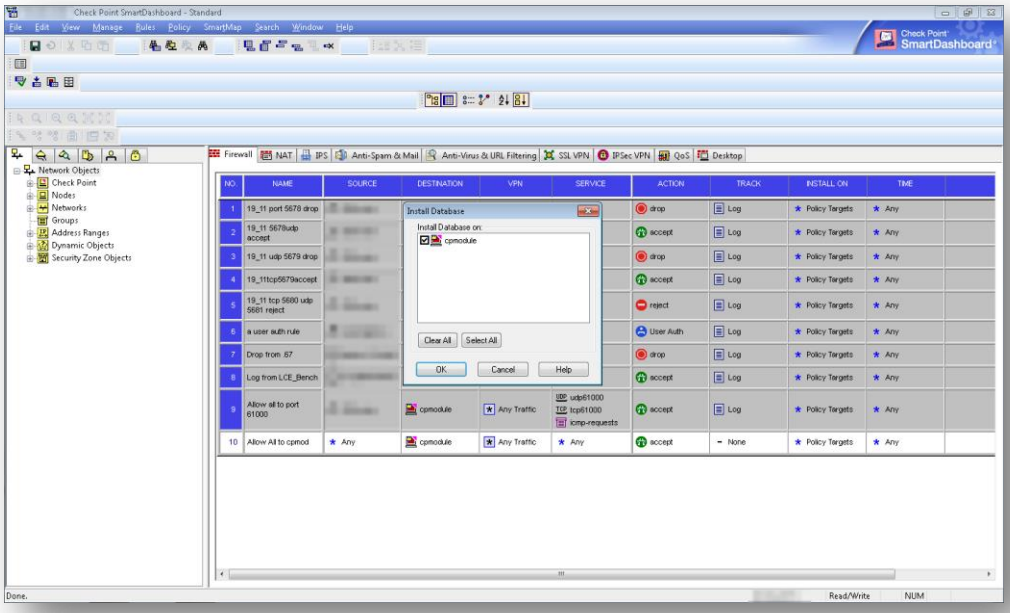


Installing the Policy Database

After adding the LEA application information, select the menu item Policy -> Install Database...



On the "Install Database" dialog, select the firewall being configured and click "OK". This will install the changes to the Security Management Server.



Creating Keys on the Check Point Security Management Server

Connect to the Security Management Server via SSH as the admin user, then complete the following steps:

1. Run the command, **`expert`**. You will be asked for a password.

2. Run the command, **`fw putkey <IP of LCE OPSEC Client>`**
3. Enter and confirm the "Secret Key". This key will be used later when configuring the LCE OPSEC Client environment.

```
# fw putkey 192.168.7.222
Enter secret key:
Again secret key:
```

LCE OPSEC Client Installation

The latest version of the LCE OPSEC client install package can be downloaded from the Tenable Downloads Page.
The dependency needed for the LCE OPSEC Client is shown below:

```
pam-1.1.1-20.el6.i686
```

To install the package use the command shown in the example below:

```
# rpm -ivh lce_opsec-4.5.0-el6.x86_64.rpm
Preparing...                ########################################### [100%]

   1:lce_opsec              ########################################### [100%]
Please run /opt/lce_opsec/set-server-ip.sh to configure your LCE server's IP and port.
```

After the installation completes run the **`set-server-ip.sh`** script to configure the client. The IP address of the LCE server and port that is used to communicate with the LCE server is needed to complete the initial configuration of the LCE OPSEC client. An example is shown below:

```
# /opt/lce_opsec/set-server-ip.sh

Enter the new desired LCE server IP or hostname.
>>
192.168.7.55


Enter the new desired LCE server port [31300].
>>
31300
Updating LCE Server IP from 203.0.113.1 to 192.168.1.155...
Updating LCE Server Port from 31300 to 31300...
Done

Starting LCE OPSEC daemon                                  [  OK  ].
```

Make sure DNS is configured in **`/etc/resolv.conf`** on the host where the LCE OPSEC Client is installed.

LCE OPSEC Client commands:

Option	Command
Install	`# rpm -ivh lce_opsec-4.5.0-el6.x86_64.rpm`
Start	`# service lce_opsec start`
Stop	`# service lce_opsec stop`
Status	`# service lce_opsec status`

Remove	Determine the name of the installed package:
	
	`# rpm -qa | grep lce_opsec`
	`lce_opsec-4.5.0-el6.x86_64`
	
	Remove the installed package:
	
	`# rpm -ev lce_opsec-4.5.0-el6.x86_64`

Configuring the LEA Server and Application

The steps below explain how to extract, locate, and move the files that are needed to create the authentication keys and pull the certificate for the OPSEC API.

1. Log in to a terminal on the LCE OPSEC Client host.
2. Change to the directory where the **`OPSEC_SDK_6_0_SHA_256.linux50.tar`** file is located.
3. Unzip the file **`OPSEC_SDK_6_0.linux30.tar`**.
4. Using the tar command, extract the files contained in **`OpsecSicUtils_linux50.tar.gz`**.

```
# cd /opt
# unzip OPSEC_SDK_6.0_Linux.zip
Archive:  OPSEC_SDK_6.0_Linux.zip
  inflating: OPSEC_SDK_6_0.linux22.tar.gz
inflating: OPSEC_SDK_6_0_SHA_256.linux50.tar.gz
  inflating: RoamAdmin_linux22.tar.gz
  inflating: RoamAdmin_linux30.tar.gz
inflating: OpsecSicUtils_linux50.tar.gz
  inflating: OpsecSicUtils_linux22.tar.gz
# tar -xf OpsecSicUtils_linux50.tar.gz
# cd linux30/
# ls
opsec_pull_cert  opsec_putkey
# cp opsec_pull_cert opsec_putkey /opt/lce_opsec
```

6. Run the following command from the **`/opt/lce`** directory: **`opsec_putkey <IP of Security Management Server>`**

7. Enter the secret key that was created in the previous step with the "**fw putkey**" command. Confirm the secret key. This creates several SSL files that begin with the prefix "CKP".

```
# ./opsec_putkey 192.168.7.44
Please enter secret key:
Please enter secret key again:
 Key for host 192.168.7.44 saved to file
```

8. The two files **CKP_shmem_._sslsess.C** and **CKP_shmem_._sslauthkeys.C** will be used to connect to the Security Management Server.

9. Run the command: **opsec_pull_cert -h <IP of Security Management Server> -p <SIC Password> -n <LEA application name>**. The SIC Password and the "LEA application name" came from the SmartDasboard's "OPSEC Application Properties" dialog when the LEA application was added to SmartDashboard.

```
# ./opsec_pull_cert -h 192.168.7.44-p password –n LCE_OPSEC_test
 The full entity sic name is:
CN=LCE_OPSEC_test,O=cpmodule..bbj6k8
Certificate was created successfully and written to "opsec.p12".
```

If the **opsec_pull_cert** command hangs, you may need to SSH to and log in to the Security Management Server, type "**expert**", and run the "**cpstop**" and "**cpstart**" commands to restart the Check Point module.

```
# cpstop
SmartPortal: Stopping CPWMD
cpwd_admin:
Process CPWMD terminated
SmartPortal: Stopping CPHTTPD
cpwd_admin:
Process CPHTTPD terminated
Stopping SmartView Monitor daemon ...
SmartView Monitor daemon is not running
Stopping SmartView Monitor kernel ...
Driver is Down.
rtmstop: SmartView Monitor kernel is not loaded

# cpstart
cpstart: Power-Up self tests passed successfully

cpstart: Starting product - SVN Foundation

SVN Foundation: cpWatchDog already running
SVN Foundation: cpd already running
```

10. Verify the **opsec.p12** (which was created in the previous step and is the OPSEC certificate file) is present in the **/opt/lce_opsec** directory by performing the following command:

```
# ls -la opsec.p12
-rw-r--r--. 1 root root 2641 Jan 22 13:15 opsec.p12
```

Make note of the name of the OPSEC certificate file as it will be required when the LCE OPSEC Client policy is created.

Configure an LCE OPSEC Client Policy

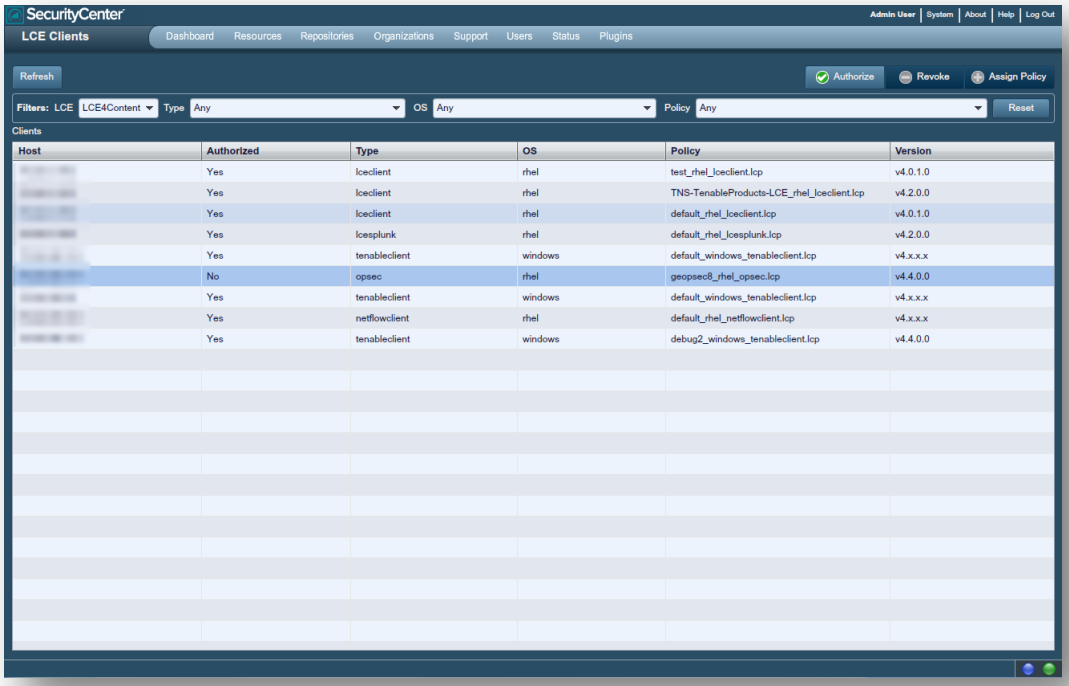
Configure the LCE OPSEC Client policy with the information gathered throughout the configuration process. The following is required to configure a LCE OPSEC Client policy:

1. The IP address and port of the Security Management Server
2. The OPSEC client DN
3. The OPSEC server DN
4. The certificate file name (**`opsec.p12`**)

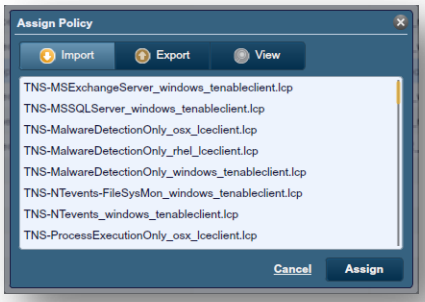
A sample LCE server policy file shown below can be used by changing the **`fw1-server`**, **`opsec-certificate`**, **`opsec-client-dn`**, and **`opsec-server-dn`** values in a text editor.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<options xmlns:xi="http://www.w3.org/2003/XInclude">
    <log-directory>/opt/lce_opsec/logs/</log-directory>
    <fw1-server>192.168.7.44</fw1-server>
    <fw1-port>18185</fw1-port>
    <dateformat>STD</dateformat>
    <authenticated>yes</authenticated>
    <opsec-certificate>opsec.p12</opsec-certificate>
    <opsec-client-dn>CN=LCE_OPSEC_test,O=cpmodule..bbj6k8</opsec-client-dn>
    <opsec-server-dn>cn=cp_mgmt,O=cpmodule..bbj6k8</opsec-server-dn>
    <heartbeat-frequency>300</heartbeat-frequency>
    <statistics-frequency>60</statistics-frequency>
    <compress-events>1</compress-events>
</options>
```

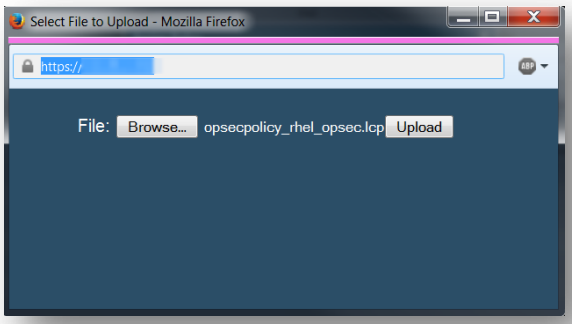
After the information is added to the LCE OPSEC Client policy, save the file as **`opsecpolicy_rhel_opsec.lcp`**, and then upload it to SecurityCenter. To do this, log in to SecurityCenter as the admin user and select “Resources” followed by “LCE Clients”. Select the LCE OPSEC client from the list of clients. Then select “Authorize” if the client hasn’t previously been authorized.



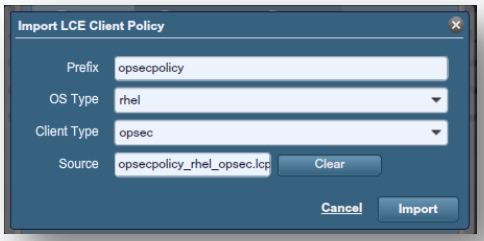
Then choose “Assign Policy” from the “LCE Clients” menu. From the “Assign Policy” window select “Import”.



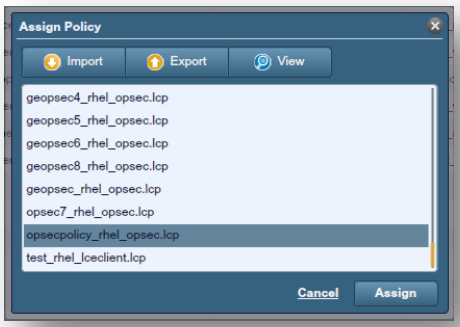
In the “Import LCE Client Policy” dialog box use the prefix “opsecpolicy”, select “rhel” from the “OS type” drop-down menu, and then choose “opsec” from the “Client Type” drop-down menu.Then choose “Browse”, followed by “Browse”, locate the **`opsecpolicy_rhel_opsec.lcp`**, and choose “Upload”.



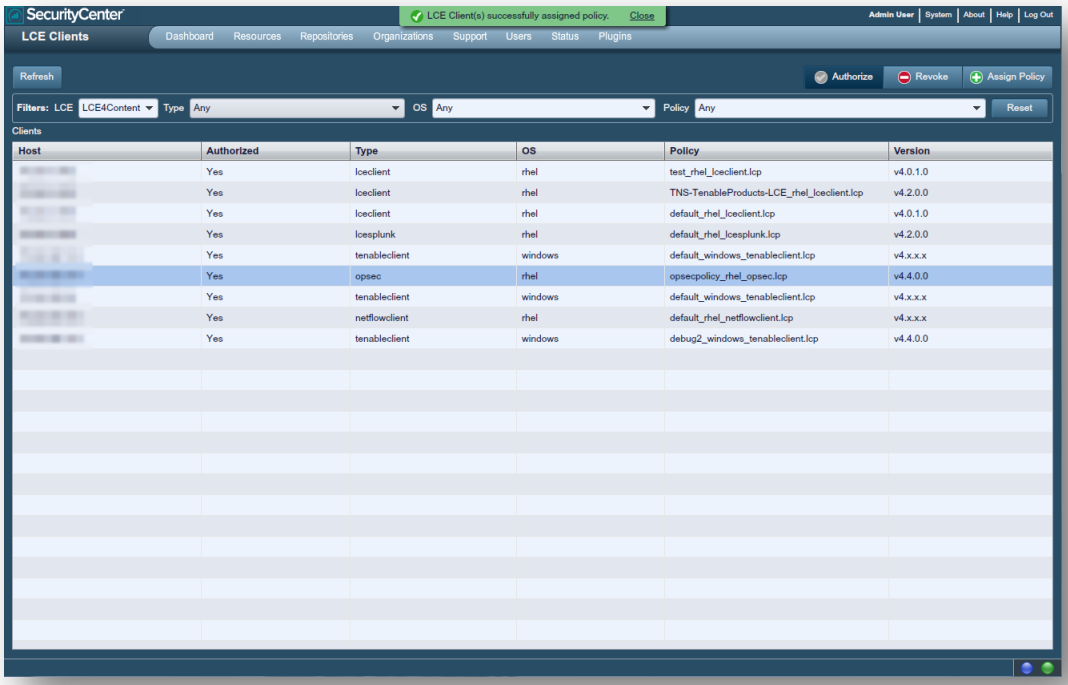
When the **`opsecpolicy_rhel_opsec.lcp`** has been uploaded, choose “Import”.



From the “Assign Policy” menu select the **`opsecpolicy_rhel_opsec.lcp`** and then choose “Assign”.



If the policy was applied successfully, the following should be displayed at the top of the LCE Clients window and the correct policy should be displayed in the "Policy" column for the LCE OPSEC Client.



Policy Parameters

The following is a list of all valid "keys" available for use with the LCE OPSEC Client policies:

Key Name	Description	Example Values
`fw1-server`	The IP address of the Check Point Security Gateway to monitor.	192.168.1.1
`fw1-port`	The port on which the Check Point Security Gateway LEA (Log Export Agent) interface operates.	18185
`opsec-certificate`	The path to the `opsec.p12` file pulled from the Check Point.	/opt/lce_opsec/opsec.p12
`opsec-client-dn`	The distinguished name of the client, provided by the SmartDashboard after adding a LEA entry.	CN=LCE-OPSEC-test,O=cpmodule..bbj6k8
`opsec-server-dn`	The distinguished name of the server, provided by the SmartDashboard.	cn=cp_mgmt,o=cpmodule..bbj6k8
`authenticated`	YES to enable cert authentication using the three options above, or NO to disable it.	YES
`dateformat`	CP, UNIX, or STD (default) - this controls the date format in the output of each log.	STD

	Examples: STD 2014-10-18 19:07:53 UNIX 1413673763 CP 18Oct2014 19:32:10	
syslog-server	Destination IP or hostname, colon, destination port of a remote syslog server to which to send events.	192.168.1.66:514
debug-level	Minimum debugging level that is printed to the log. The options supported are as follows: VERBOSE INFO WARN ERROR NONE.	INFO
log-directory	The path to which to write the LCE OPSEC admin logs.	/opt/lce_opsec/logs/
local-ip-net	The preferred network from which this client will connect to the LCE server. This can be used to control the source network address of the client. Note: if a client policy is assigned with a new local-ip-net that causes the client to show up as connecting from another IP, then using SecurityCenter, the policy for that client will need to be re-assigned to the new client entry for that IP.	192.168.1.0/24
heartbeat-frequency	The number of seconds between each client heartbeat message to the LCE server. If "0", it will not send heartbeats.	A positive integer.
statistics-frequency	The number of minutes between each client host performance statistics report (CPU, Disk Space, and Physical Memory) sent to the LCE server. If "0", it will not send stats.	A positive integer.
monitor-period	Period (in milliseconds) of the monitor loop within the LCE Client.	1000
compress-events	Whether or not to compress events before transmitting them to the LCE server. Marginally saves bandwidth, marginally increases CPU usage.	0 or 1 (0=off, 1=on)
compression-level	This can be used to further define "compress-events". The value can be 0-9 with the highest value offering the most amount of compression. The more compression that is used, the more impact it will have on CPU usage.	0-9
minimum-compression-ratio	The minimum compression ratio is the minimum required ratio of the original data size to the compressed data size. If the ratio satisfies the minimum ratio, compression is used for that subset of events in transit to the LCE server. Otherwise, compression is not used for that subset of events. Lower this value to compress packets more often.	1.0-10.0
minimum-compression-	This defines the size of event packets that will be compressed in bytes. The lower the value, the more often compression	0-1500

input-size	takes place. For example, a lower value of 100 can be used if the goal is to compress more packets.	
event-queue-timeout	The maximum time allowed to pass, in seconds, before the event queue in the client is flushed and all queued events are transmitted to the LCE server. Higher values may cause latency between the LCE OPSEC Client and LCE server.	1-60

LCE Conf Converter

The LCE Conf Converter is a utility to convert LCE configuration files from versions of the LCE Clients prior to 4.0 to new policy files.

The following command run from the command line with no options will display the help file:

```
# /opt/lce/daemons/lce_conf_file_converter
```

There are four valid options to use as described in the table below:

Option	Description
--input-conf-file **-i**	The input configuration file (i.e., **lce_opsec.conf**)
--output-policy-file **-o**	The output policy file (e.g., **my-new-policy.lcp**)
--help **-h**	Display the help menu
--version **-v**	Display version information

Once saved as a policy file, the converted file may be imported to the LCE Client Manager and assigned to the appropriate client(s).

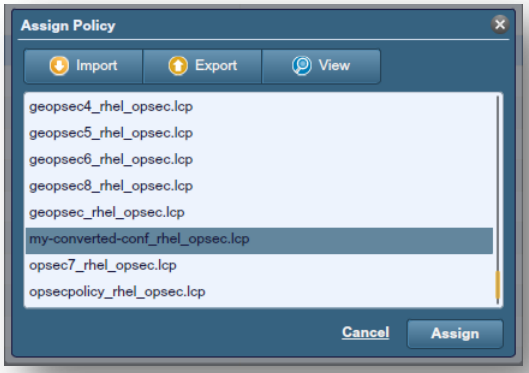
The following is an example of how to convert an **lce_opsec.conf** to a policy file (for RHEL), and add the policy that is created to the LCE:

```
# /opt/lce/daemons/lce_conf_file_converter -i
      /opt/lce_opsec/lce_opsec.conf -o ~/lce_opsec_conf.lcp

Successfully converted /opt/lce_client/lce_opsec.conf to policy
/root/lce_opsec_conf.lcp.

# /opt/lce/daemons/lce_client_manager --import-policy
      ~/lce_opsec_conf.lcp --output-policy my-converted-conf
      --client-type opsec --os-type rhel
      /opt/lce/daemons/policies/my-converted-conf_rhel_opsec.lcp
```

If there is an error, a non-zero error code will be displayed.

The policy that was added to the LCE can be found in SecurityCenter CV by logging in as the Admin user, and selecting "Resources" followed by "LCE Clients". Select the client that requires the policy from the list of clients, and choose "Assign

Policy" to view the available policies. Choose the imported policy, and select "Assign" to apply the policy to the LCE OPSEC Client.

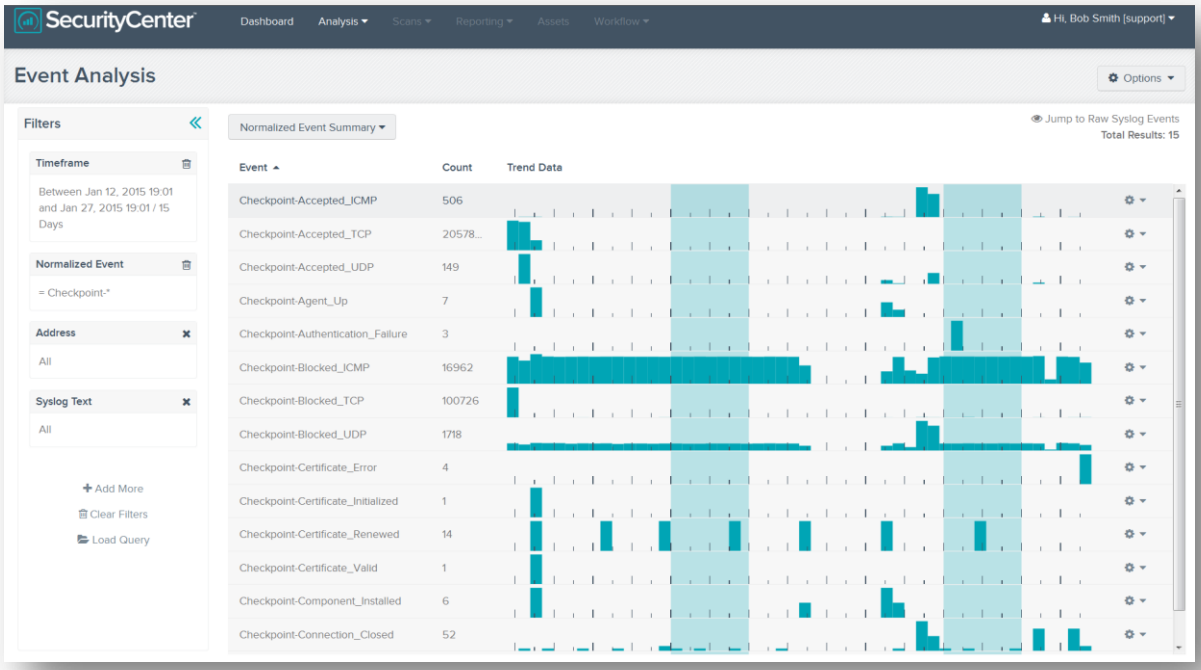


Viewing Check Point Events in SecurityCenter CV

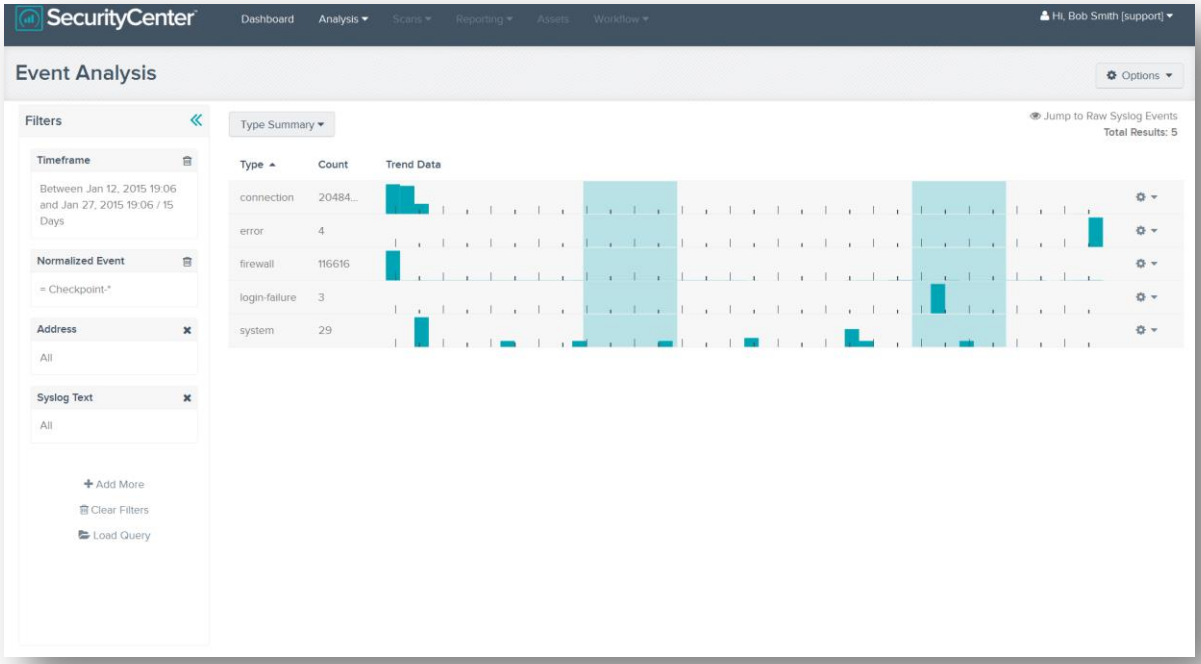
To view Check Point events in SecurityCenter CV, log in as the Security Manager user, or a user with access to the repository where the Check Point events are stored. Then select "Events" from the "Analysis" tab. After the intial screen loads showing all events, the following search can be run for all normalized Check Point events:

```
Checkpoint-*
```

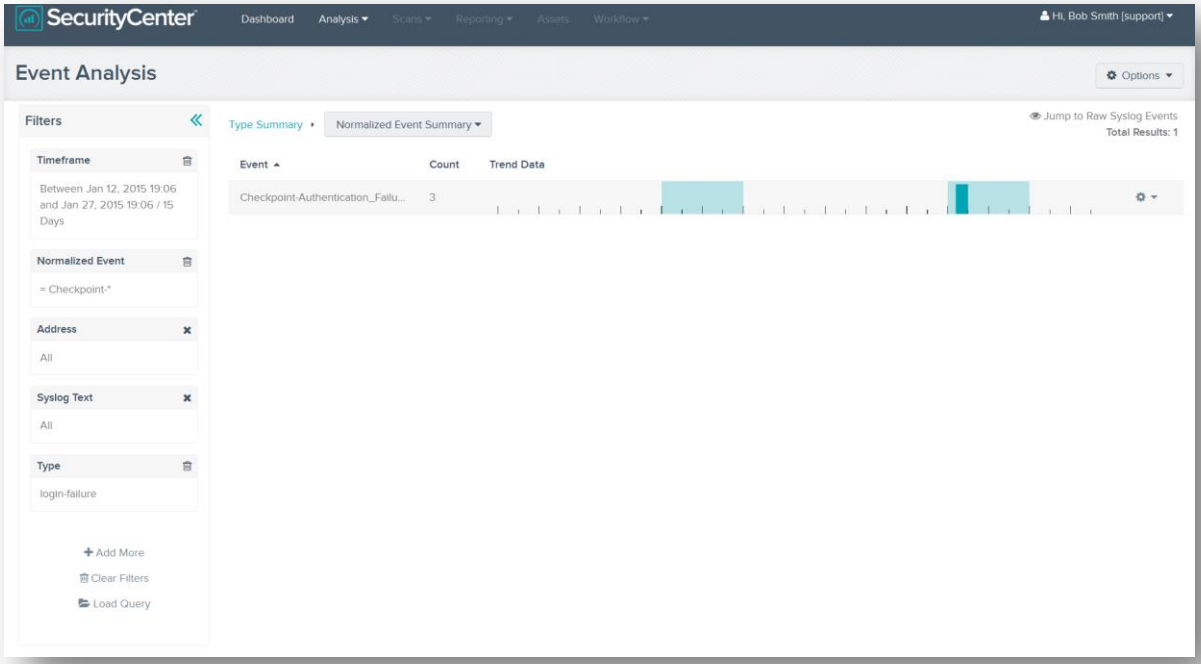
The results are shown below:



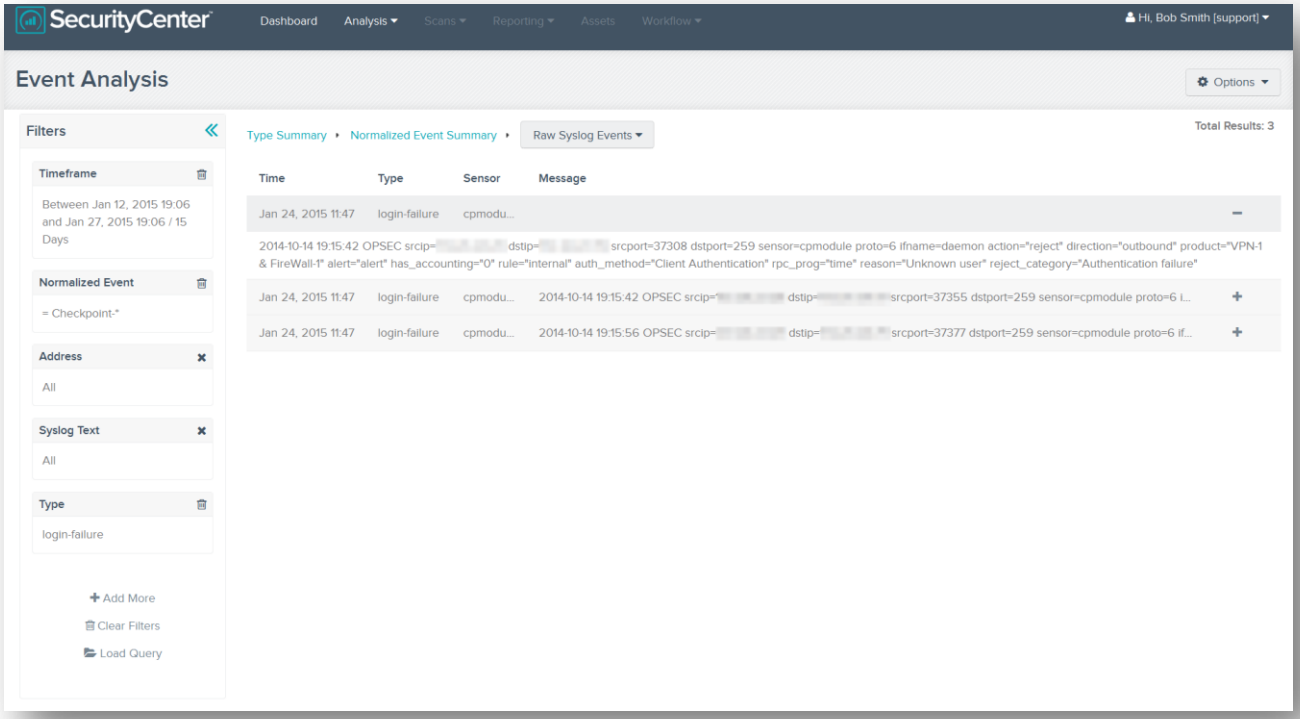
By selecting "Normalized Event Summary" at the top of the "Event Analysis" window a list of filters will be shown. Choose the option "Type Summary" the Check Point data will be organized by type as shown below:



From the "Type Summary" view selecting the number "3" next to the "login-failure" will display the associated normalized event name.



If more detailed information is required about these "login-failure" events, "Jump to Raw Syslog Events" can be chosen from the upper right hand corner of the "Event Analysis" window. In the "Raw Syslog Events" view selecting the plus symbol will display the complete log from the Check Point Firewall as shown below:



For More Information

Tenable has produced a variety of documents detailing the LCE's deployment, configuration, user operation, and overall testing. These documents are listed here:

- Log Correlation Engine 4.2 Architecture Guide – provides a high-level view of LCE architecture and supported platforms/environments.
- Log Correlation Engine 4.4 Administrator and User Guide – describes installation, configuration, and operation of the LCE.
- Log Correlation Engine 4.4 Quick Start Guide – provides basic instructions to quickly install and configure an LCE server. A more detailed description of configuration and management of an LCE server is provided in the "LCE Administration and User Guide" document.
- Log Correlation Engine 4.4 Client Guide – how to configure, operate, and manage the various Linux, Unix, Windows, NetFlow, OPSEC, and other clients.
- Log Correlation Engine 4.4 OPSEC Client Guide – how to configure, operate, and manage the OPSEC Client.
- LCE 4.4 High Availability Large Scale Deployment Guide – details various configuration methods, architecture examples, and hardware specifications for performance and high availability of large scale deployments of Tenable's Log Correlation Engine (LCE).

- LCE Best Practices – Learn how to best leverage the Log Correlation Engine in your enterprise.
- Tenable Event Correlation – outlines various methods of event correlation provided by Tenable products and describes the type of information leveraged by the correlation, and how this can be used to monitor security and compliance on enterprise networks.
- Tenable Products Plugin Families – provides a description and summary of the plugin families for Nessus, Log Correlation Engine, and the Passive Vulnerability Scanner.
- Log Correlation Engine Log Normalization Guide – explanation of the LCE's log parsing syntax with extensive examples of log parsing and manipulating the LCE's `.prm` libraries.
- Log Correlation Engine TASL Reference Guide – explanation of the Tenable Application Scripting Language with extensive examples of a variety of correlation rules.
- Log Correlation Engine 4.0 Statistics Daemon Guide – configuration, operation, and theory of the LCE's statistic daemon used to discover behavioral anomalies.
- Log Correlation Engine 3.6 Large Disk Array Install Guide – configuration, operation, and theory for using the LCE in large disk array environments.
- Example Custom LCE Log Parsing - Minecraft Server Logs – describes how to create a custom log parser using Minecraft as an example.

Documentation is available for all products at docs.tenable.com

There are also some relevant postings at Tenable's blog located at http://www.tenable.com/blog.

For further information, please contact Tenable at support@tenable.com, sales@tenable.com, or visit our web site at http://www.tenable.com/.

Appendix 1: Non-Tenable License Declarations

Below you will find third party software packages that Tenable provides for use with the Log Correlation Engine.

Section 1 (b) (ii) of the Log Correlation Engine License Agreement reads:

(ii) The Software may include code or other intellectual property provided to Tenable by third parties (collectively, "Third Party Components"). Any Third Party Component that is not marked as copyrighted by Tenable is subject to other license terms that are specified in the Documentation. By using the Software, you hereby agree to be bound by such other license terms as specified in the Documentation.

The Log Correlation Engine's Software License Agreement can be found on the machine in the top-level directory for the LCE application, **`/opt/lce`**.

Related 3rd Party and Open-Source Licenses

blowfish.h

This product includes cryptographic software written by Eric Young (eay@mincom.oz.au).
This product includes software written by Tim Hudson (tjh@mincom.oz.au).

crypto/bf/blowfish.h
Copyright (C) 1995-1998 Eric Young (eay@mincom.oz.au)
All rights reserved.

This package is an SSL implementation written by Eric Young (eay@mincom.oz.au).
The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@mincom.oz.au).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the for m of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:
"This product includes cryptographic software written by Eric Young (eay@mincom.oz.au)"
The word 'cryptographic' can be left out if the rouines from the library being used are not cryptographic related : -).
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) y ou must include an acknowledgement:
"This product includes software written by Tim Hudson (tjh@mincom.oz.au)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT

LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

libCURL

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2011, Daniel Stenberg, <daniel@haxx.se>.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright
notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not
be used in advertising or otherwise to promote the sale, use or other dealings
in this Software without prior written authorization of the copyright holder.

OpenSSL

Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in
   the documentation and/or other materials provided with the
   distribution.

3. All advertising materials mentioning features or use of this
   software must display the following acknowledgment:
   "This product includes software developed by the OpenSSL Project
   for use in the OpenSSL Toolkit. (http://www.openssl.org/)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
   endorse or promote products derived from this software without
   prior written permission. For written permission, please contact
   openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:
"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

zlib

(C) 1995-2010 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

Hash functions

'Hash functions' is Copyright 2004-2008 by Paul Hsieh, and distributed under the LGPL 2.1 license.

OpenBSM

OpenBSM is covered by a number of copyrights, with licenses being either two or three clause BSD licenses. Individual file headers should be consulted for specific copyrights on specific components.

libpcap

License: BSD

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

libmcrypt

libmcrypt (part of the mcrypt project) is distributed under the LGPL 2.1 license.

libxml2

Libxml2 is the XML C parser and toolkit developed for the Gnome project (but usable outside of the Gnome platform), it is free software available under the MIT License.