Network Integration Checkpoint

CB v4.2.4.150206.1225

February 06, 2015

Contents

Overview

Carbon Black provides integration with an on-premise Check Point device for correlating Check Point alerts with Carbon Black collected data. More information about Check Point can be found at: http://www.checkpoint.com/

To support this integration, Carbon Black provides an out-of-band bridge that receives alerts from the Check Point device and communicates with the Carbon Black enterprise server.

Prerequisites

```
1. A Carbon Black enterprise server installation >= 4.0

2. Check Point
```

Bridge Installation

1. Configure a yum repo that points to the Carbon Black yum repository that contains the Check Point bridge. Create a new file '/etc/yum.repos.d/CheckPoint.repo' with the following content:

```
[CheckPoint]

name=CheckPoint
baseurl=https://yum.carbonblack.com/enterprise/integrations/checkpoint/x86_64

gpgcheck=0
enabled=1

metadata_expire=60
sslverify=1

sslclientcert=/etc/cb/certs/carbonblack-alliance-client.crt
sslclientkey=/etc/cb/certs/carbonblack-alliance-client.key
```

2. Verify the yum configuration and install the Check Point bridge

```
yum info python-cb-checkpoint-bridge
yum install python-cb-checkpoint-bridge
```

3. Edit the Check Point bridge configuration file

   The Check Point bridge configuration is located here:

```
/etc/cb/integrations/carbonblack_checkpoint_bridge/carbonblack_checkpoint_bridge.conf
```

   Update the *carbonblack_server_url* option to set the URL of the Carbon Black enterprise server.

   Update the *carbonblack_server_token* options to set a Carbon Black enterprise server user api token that has administrative rights on the server.

   The remainder of the options are documented and can be customized if needed to match specific requirements.

   Save the configuration

4. *Do NOT start the bridge yet*

   The initial Check Point configuration must be performed before starting the Carbon Black Check Point Bridge.

Check Point Configuration

The following sections outline the steps necessary to enable the Check Point LEA server and configure the Carbon Black LEA application

Enable the LEA Server

This must be done if the LEA server on the Check Point device has not already been configured.

1. Log onto the terminal access provided by the Check Point device. Depending on configuration, *expert* mode may be required after logging in.
2. Edit the file *$FWDIR/conf/fwopsec.conf and add the following configuration options:

```
lea_server auth_port 18184
lea_server auth_type sslca
```

   Save the configuration.

3. Restart the Check Point FW service

```
cpstop
cpstart
```

Modify rule-set to allow LEA

1. Create a firewall rule to allow connections from the host running the Carbon Black Check Point bridge for the following services:

```
FW1_ica_pull
FW1_lea
```

Register OPSEC LEA client application

1. Using the Check Point SmartDashboard, navigate to *Manage -> Servers and OPSEC Applications*
2. Add a new entry for the Carbon Black OPSEC LEA application

   Specify a name for the application registration (e.g. - *CB_LEA_APP*)

   Specify a new host (e.g. - *CB_Bridge*)

   Ensure that the *LEA* option in *Client entities* is checked

   Click on the *Communication* button, enter a one-time password and select *Initialize*. This will create the certificates needed for secure LEA communications. Close the *Communication* window, and there should be a DN shown. (If not, the DN should appear after closing the application window and opening again)

   Take note of this DN, it will be needed in later steps.

Retrieve OPSEC LEA application certificate

1. On the Carbon Black Check Point bridge host, use the *opsec_pull_cert* to pull the application certificate from the Check Point server. The utility is located here:

   ```
   /usr/share/cb/integrations/carbonblack_checkpoint_bridge/opsec_lea/tools/
   ```

2. Execute the following command to retrieve the certificate

   ```
   opsec_pull_key -h [checkpoint server ip or name] -n [opsec lea application registratio

   Example: opsec_pull_key -h 192.168.1.10 -n CB_LEA_APP -p mypassword
   ```

3. This will create a file in the current directory named *opsec.p12*. Place this file in the following directory:

   ```
   /usr/share/cb/integrations/carbonblack_checkpoint_bridge/opsec_lea/
   ```

Create and retrieve the host authentication key

1. Using the Check Point server terminal access, execute the following command to create the keys:

   ```
   fw putkey -opsec -ssl [ip address of bridge host]
   ```

   This will ask for a secret key that will be used to retrieve the authentication keys

2. Using the Carbon Black Check Point bridge host, execute the following command to retrieve the keys:
   The utility is located here:

   ```
   /usr/share/cb/integrations/carbonblack_checkpoint_bridge/opsec_lea/tools/

   opsec_putkey -ssl -port 18184 [ip address of Check Point server]
   ```

3. This will generate two files in the current directory - *sslauthkeys.C* and *sslsess.C*. Place these files in the following directory:

   ```
   /usr/share/cb/integrations/carbonblack_checkpoint_bridge/opsec_lea/
   ```

Carbon Black OPSEC LEA configuration

1. Using the Carbon Black Check Point bridge host, edit the following file:

   ```
   /usr/share/cb/integrations/carbonblack_checkpoint_bridge/opsec_lea/cb_checkpoint_opsec
   ```

2. Ensure the options are set based on the following example:

   ```
   opsec_sic_name "[DN from application registration]"
   opsec_sslca_file "opsec.p12"
   opsec_shared_local_path "/usr/share/cb/integrations/carbonblack_checkpoint_bridge/opse
   opsec_sic_policy_file "cb_checkpoint_opsec_lea_sic_policy.conf"
   lea_server ip [Check Point server ip address]
   lea_server auth_port 18184
   lea_server auth_type sslca
   lea_server opsec_entity_name "[DN of Check Point server]"
   ```

   The server DN can be obtained by double clicking on the main Check Point server object, and looking under Test SIC Status

3. Save any changes to the configuration file

Start the Check Point Bridge

1. Start the bridge service

   ```
   /etc/init.d/cb-checkpoint-bridge start
   ```

2. Examine the Check Point bridge log to verify the service is running normally

   ```
   /var/log/cb/integrations/carbonblack_checkpoint_bridge/carbonblack_checkpoint_bridge.l
   ```

Check Point Feed

Once the service is running, the Check Point feed can be added to the Alliance feeds on the enterprise server. Add a new feed and specify the following URL:

```
http://[bridge host]:[listener_port from bridge config]/checkpoint/json

Example: http://127.0.0.1:7000/checkpoint/json
```