Threatconnect Connector

CB v4.2.5.150311.1434

March 11, 2015

Contents

Overview 1
Prerequisites 1
Installation 1
Configuration 2
Operation 2
Troubleshooting 3
Common Issues 3

Overview

Carbon Black provides integration with ThreatConnect by retrieving indicators of compromise (IOCs) from specified communities.

To support this integration Carbon Black provides an out-of-band connector that communicates with the ThreatConnect API.

Prerequisites

Before installing the bridge you will need to have the following:

- The main Carbon Black repository installed
- Your Carbon Black Alliance SSL certificate and key. Usually located in:

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

- A ThreatConnect API key and secret key

Installation

Installation of the ThreatConnect connector should be as simple as using yum to retrieve and install the package from the Carbon Black repo.

1. Verify the machine has Internet connectivity.
2. Configure a yum repo that points to the Carbon Black ThreatConnect Connector repository. Create a new file '/etc/yum.repos.d/threatconnect.repo' with the following contents:

```
[ThreatConnect]
name=ThreatConnect
baseurl=https://yum.carbonblack.com/enterprise/integrations
/threatconnect/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
```

3. Verify that the necessary repos exist
    1. The main Carbon Black repo entry is usually located in /etc/yum.repos.d/CarbonBlack.repo
    2. The ThreatConnect Connector repo entry is usually located in /etc/yum.repos.d/threatconnect.repo
4. Verify the yum configuration and install the ThreatConnect bridge

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

5. Configure the connector as per the **Configuration** section
6. Examine the ThreatConnect Connector logs to verify that the daemon is running normally

```
/var/log/cb/integrations/carbonblack_threatconnect_bridge/
carbonblack_threatconnect_bridge.log
```

7. Log in to the Carbon Black Web UI, navigate to the Alliance Feeds section, click on “Create Feed” and use the url http://127.0.0.1:6100/threatconnect/json for the source. Be sure to modify the IP and port numbers if you selected alternate values during configuration.

Configuration

Edit the configuration file at /etc/cb/integrations/cb_threatconnect_bridge/cb_threatconnect_bridge.conf before starting the ThreatConnect connector.

The following fields should be configured:

1. **api_key** - The secret key for your ThreatConnect API account.
2. **secret_key** or **secret_key_encrypted** - The secret key for your ThreatConnect API account. Set the secret key in plaintext under secret_key or use the /usr/share/cb/cbpasswd utility if you prefer not to store a plaintext password on disk. Note that the cbpasswd utility is a part of the carbonblack server installation. You will need to run the script on the server machine if you installed this connector to a different machine.
3. **feed_retrieval_minutes** - In minutes how often should the connector retrieve the full list of indicators of compromise?
4. **listener_address** - What address should the connector listen for connections on? Default is 127.0.0.1
5. **listener_port** - What address should the connector listen for connections on? Default is 6100
6. Add any additional communities to which you are subscribed under the [sources] section with a unique tag and corresponding API endpoint.

Operation

Starting

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

Stopping

```
/etc/init.d/cb-threatconnect-bridge stop
```

Write to file and exit

```
/usr/share/cb/integrations/carbonblack_threatconnect_bridge/
carbonblack_threatconnect_bridge write <filename_to_write>
```

Viewing diagnostic page With the ThreatConnect Connector daemon running you can connect to

```
http://127.0.0.1:6100/
```

Be sure to change the IP address or port if you modified the default settings.

Viewing the Feed With the ThreatConnect Connector daemon running you can connect to

```
http://127.0.0.1:6100/threatconnect/json
```

Be sure to change the IP address or port if you modified the default settings.

Troubleshooting

- You may have to modify iptables your firewall rules if you want to connect from a remote system
- The connector's log file is located in

  ```
  /var/log/cb/integrations/carbonblack_threatconnect_bridge/
  carbonblack_threatconnect_bridge.log
  ```

- You can try running the program directly along with write mode to see any exceptions that get thrown:

  ```
  /usr/share/cb/integrations/carbonblack_threatconnect_bridge/
  carbonblack_threatconnect_bridge write <filename_to_write>
  ```

- Connection issues arising from failed logins will be written to the log file as well as appended to the "last_sync column" on the connector diagnostic page, located from the same machine at http://localhost:6100/
  sample error: No sync performed ({u'status': u'Failure', u'errorMessage': u'Timestamp out of acceptable time range'})

Common Issues

- Symptom: No sync occurs, "`Timestamp out of acceptable time range`" reported in error logs or connector diagnostic page
  - Diagnosis: The system clock is out of sync. Successful authentication with the ThreatConnect API requires the system clock to be correct.
  - Fix: Set the system clock

    ```
    date --set="2 MAR 2014 18:00:00"
    ```

- Symptom: No sync occurs, "`Signature data did not match expected result`" reported on error logs or connector diagnostic page
  - Diagnosis: The text of the community URL is incorrect. This text is used as part of the ThreatConnect API. Below is one example of a typo that can lead to this error.
    - Incorrect (note the extra percent sign):
      `CommonCommunity = /v1/indicators/?owner=Common%%20Community`

    * Correct:
      `CommonCommunity = /v1/indicators/?owner=Common%20Community`

- Fix: Carefully examine the URL for any typos
- Symptom: No sync occurs, `{u'status': u'Failure', u'errorMessage': u"}` reported in error logs or connector diagnostic page
  - Diagnosis: Your API key or Secret Key is incorrect
  - Fix: Double check that your API key and Secret Key were entered correctly. If you used the cbpasswd utility make sure that all symbols are properly escaped, as bash and other shells may attempt to interpret special characters before sending them into the script.