VMware Carbon Black Cloud Sensor Installation Guide

Modified on 08 MAR 2021

VMware Carbon Black Cloud

You can find the most up-to-date technical documentation on the VMware website at:

https://docs.vmware.com/

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Contents

Preface

The *VMware Carbon Black Cloud Sensor Installation Guide* provides installation and configuration instructions for the VMware Carbon Black Cloud™ sensors.

You can install a Carbon Black Cloud sensor on Windows, macOS, and Linux endpoints, and on endpoints in VDI environments. The sensor provides data from the endpoints to Carbon Black Cloud analytics. You can also secure VMware workloads and Kubernetes cluster workloads by using the Carbon Black Cloud.

Intended Audience

This documentation provides sensor installation, upgrade, and uninstall instructions for administrators, incident responders, and others who will operate the VMware Carbon Black Cloud.

Staff who manage Carbon Black Cloud activities should be familiar with operating systems, web applications, installed software, desktop infrastructure (especially in-house procedures for software roll-outs, patch management, and anti-virus software maintenance), and the effects of unwanted software.

Getting Started

1

You can install a Carbon Black Cloud sensor on Windows, macOS, and Linux endpoints, and on endpoints in VDI environments. The sensor provides data from the endpoints to Carbon Black Cloud analytics.

The following instructions describe how to install sensors on endpoints. To install and manage sensors on workloads, see Chapter 8 Managing Sensors for VM Workloads.

Method 1: Invite Users to Install Sensors on Endpoints

- Invited users receive an email that contains an installation code; each invited user installs the sensor directly on an endpoint. This method is not available for Linux sensors.
- This method is useful for installing sensors to a small number of endpoints.

Method 2: Install the Sensor on the Endpoint by using the Command Line or Software Distribution Tools

- The command line method allows for small-scale deployments and testing.
- A scripted or automated method installs the sensor by using software distribution tools. This method is useful when installing sensors across a large number of endpoints.

This chapter includes the following topics:

Before you Install Sensors on Endpoints

Make sure that endpoints meet the operating environment requirements for the Carbon Black Cloud products that you have purchased.

- Enterprise EDR Operating Environment Requirements
- Container Essentials Operating Environment Requirements

Note Some sensor names contain the product name “CB Defense.” This is correct: the same sensors apply for all Carbon Black Cloud products.

About Sensor Groups and Policy Assignments

Each sensor is assigned a policy that determines what policy rules apply to the sensor.

By default, each new sensor is assigned the Standard policy unless one of the following conditions applies:

- You define an alternate policy during a command line installation.
- You have previously created sensor groups,the installed sensor matches a sensor group’s criteria, and the target policy is not the Standard policy.

All the sensors in the sensor group receive an automatic assignment to a policy, which is based on the metadata that is associated with the sensor and the criteria that you define. This capability requires the following (or later) sensor versions:

- Windows sensors v3.1
- macOS sensors v3.2
- Linux sensors v2.5

You cannot define the policy during a direct user installation; however, you can change the policy to which a sensor is assigned after its installation.

Note Policy assignments do not apply to the Audit and Remediation Standalone product.

Local Scan Settings

The local scan feature is only available for Windows sensors 2.0 and later. It is not available for the Audit and Remediation Standalone product, Linux sensors, or macOS sensors.

For more information about Local Scan Settings for Windows, see Local Scan Settings and the AV Signature Pack.

Method 1: Invite Users to Install Sensors on Endpoints

This method is useful when you have a small number of sensors to install, or when software distribution tools are not available. This method is not available for Linux sensors.

The installation code will expire after seven (7) days.

Important The user on the endpoint must have administrator privileges to install the sensor.

Note With the release of the Windows 3.6 sensor, you can supply either the installation code or the company code to install the sensor.

To Invite Users to install Sensors

You can invite users to install sensors on their endpoints.

Procedure

1. Sign in to the Carbon Black Cloud console.
2. On the navigation bar, click **Inventory** and then click **Endpoints**.
3. Click **Sensor Options** and then click **Send installation request .**
4. Add a single user or multiple users. To add multiple users, type a comma-separated list of email addresses and then click **Send**.

Results

Users receive an email invitation that contains the installer download link and a unique single use installation code. The installation code expires after one week. If the installation code expires, follow the procedure To Send a new Installation Code

The users should follow the instructions in the email to install the sensor. In the email, end users will click on the appropriate OS installer link to download the sensor.

Note We recommend that you inform users in advance that you're sending the email invitation. In the advance notification, tell the users which version to download (32-bit or 64-bit). The 32-bit variant of the sensor does not run on a 64-bit version of Windows. Instruct the users that they should copy/paste the installation code into a plain text editor, and then copy/paste that entry into the installer. Copy/pasting the installation code directly from the console does not always work properly.

To Send a new Installation Code

If installation codes have expired, you can follow these steps to send new installation codes to users.

Procedure

1. Sign in to the Carbon Black Cloud console.
2. On the navigation bar, click **Inventory** and then click **Endpoints**.
3. Search for and select the sensors that have expired installation codes.
4. Click **Take Action** and click **Send new installation code**.

Method 2: Install the Sensor on the Endpoint by using the Command Line or Software Distribution Tools

You can install sensors on the command line, or by using a scripted or automated method such as Group Policy or systems management tools.

The latter method is useful when you are installing sensors across a large number of endpoints.

Note Sensors automatically try to detect proxy settings during initial installation. This should be tested. If the automatic detection does not succeed, you must define the parameters to include the proxy IP address and port in the MSI command line. See Configure a Proxy.

Follow these procedures in the order listed:

1 To Obtain a Company Registration Code

2 To Download Sensor Kits

3 To Install Sensors on Endpoints

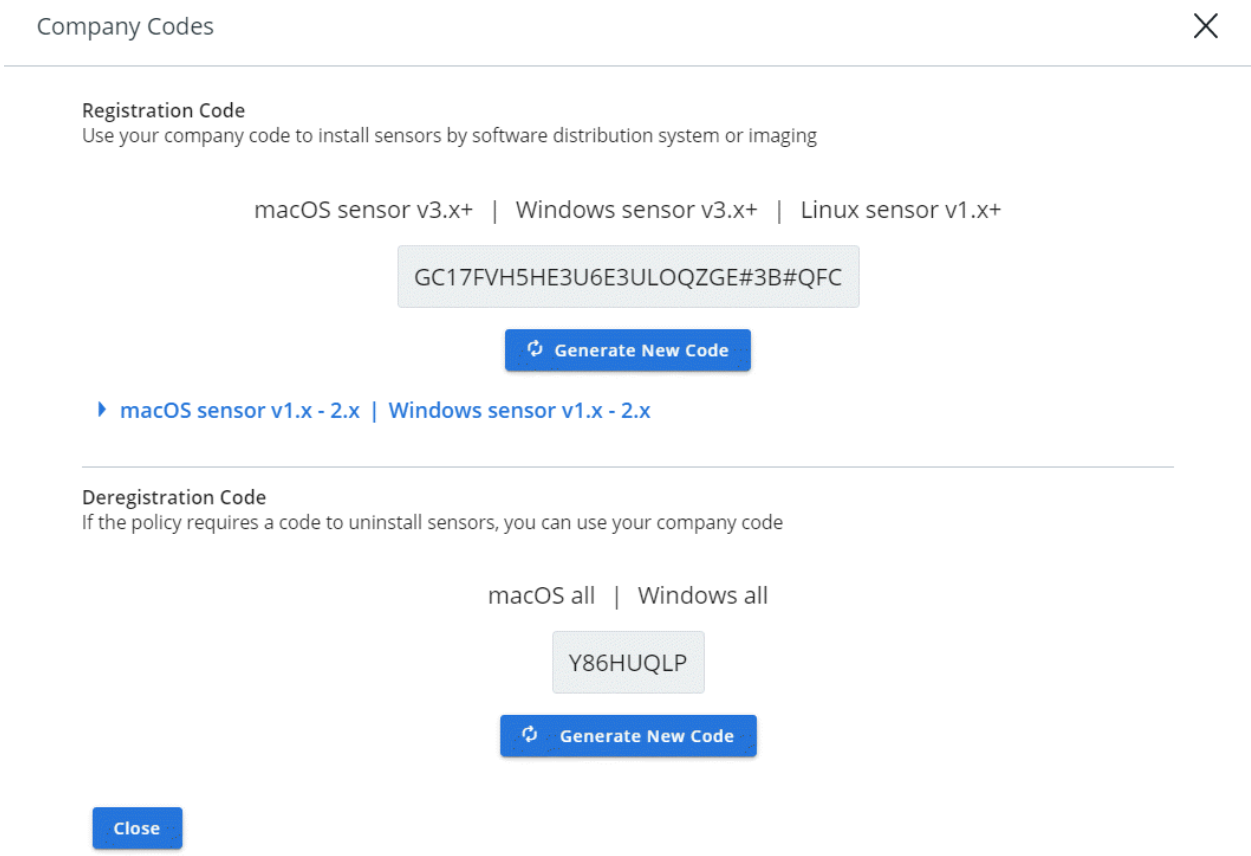
To Obtain a Company Registration Code

A company registration code is required to register new sensors.

Procedure

1. Sign in to the Carbon Black Cloud console.
2. On the navigation bar, click **Inventory** and then click **Endpoints**.
3. Click **Sensor Options** and click **Company codes**.
4. If the company code has not already been assigned, under **Registration Codes , click the Generate New Code** button.

Results



You can also generate a company deregistration code to be required for uninstalling sensors directly at the endpoints.

Take note of the generated codes so that you can supply them during the installation. We recommend that you copy/paste the codes into a plain text editor and then copy/paste them from that source.

Note For 3.0 and later Windows or macOS sensor versions, the length of the company registration code is extended.

Use the company registration code that is specified as 3.0 to install all 3.0 and later Windows and macOS sensors, and use the 1.x — 2.x code to update Windows or macOS sensors prior to version 3.0. The process of supplying the code during sensor install remains the same. You must update any software distribution tools or any existing installation scripts to use the extended codes.

Use the code that is specified for 3.0 and later sensors to install Linux sensors.

You can change the company registration code. If you install sensors using a specific company registration code and then change the code and install sensors using the new code, the old sensors will continue to operate. Installed sensors are unaffected. Only new installation packages must use the new code.

To Download Sensor Kits

You must download a sensor kit that matches the operating system of the endpoint.

Procedure

1. Sign in to the Carbon Black Cloud console.
2. On the navigation bar, click **Inventory** and then click **Endpoints**.
3. Click **Sensor Options** and click **Download sensor kits**.
4. Select the appropriate sensor kit version and click the link to download it.

To Install Sensors on Endpoints

Sensor installation on endpoints varies by operating system and environment.

See the following sections for specific sensor installation instructions:

- Chapter 2 Installing Linux Sensors on Endpoints
- Chapter 3 Installing macOS Sensors on Endpoints
- Chapter 4 Installing Sensors on Endpoints in a VDI Environment
- Chapter 5 Installing Windows Sensors on Endpoints

Installing Linux Sensors on Endpoints

2

This section describes how to install Linux sensors on the command line.

Important Before you begin the processes described here, read Chapter 1 Getting Started. It contains highly relevant information to help you succeed in your sensor installation.

Before you can install sensors, you must perform the following steps:

To Obtain a Company Registration Code

To Download Sensor Kits

The sensor kit is a .tgz with the format `cb-psc-sensor-<DISTRO>-<BUILD-NUMBER>.tgz`.

With the release of the Carbon Black Cloud v2.5.0 Linux sensor, Audit and Remediation and Enterprise EDR are supported on the Linux platform. The Carbon Black Cloud Linux sensor is highly modularized. It can support independent runtime enablement of Enterprise EDR and Audit and Remediation. You can manually customize the installer package to install only desired features. To install Audit and Remediation only, see Customizing the Carbon Black Cloud Linux feature selection.

To configure a proxy for a Linux installation, see Configure a Proxy for Linux.

Note If the company registration code contains special characters (!, #, *, $, etc.) and is not quoted, the installation will immediately terminate. Double quotation marks are not an acceptable substitute to single quotes.

This chapter includes the following topics:

To Unpack the Agent

The first step in installing a Linux sensor on an endpoint is to unpack the agent.

Procedure

1. Create a root-owned temporary install directory on the endpoint; do not use a shared folder such as `/tmp` or `/var/tmp`:

   ```
   $ mkdir cb-psc-install
   ```

2. Extract the contents of the installer package into the temporary directory you created. Replace `cb-psc-sensor-<DISTRO>-<BUILD-NUMBER>.tgz` with the filename of the installer package.

   ```
   $ tar -C cb-psc-install -zxf cb-psc-sensor-<DISTRO>-<BUILD-NUMBER>.tgz
   ```

Prerequisites for Linux 4.4+ Kernels for Linux Sensor Versions 2.10+

Prior to installing the sensor, the underlying BPF implementation requires the Linux kernel headers for the active kernel to be installed.

You can check the running kernel version by running the following command: `$ uname -r`

For CentOS, RHEL, Oracle RHCK or Amazon Linux

- To check whether the kernel headers are installed (any user can run this):

  ```
  $ yum list kernel-devel-$(uname -r)
  ```

- To install the necessary kernel headers:

  ```
  $ sudo yum install -y kernel-devel-$(uname -r)
  ```

- When properly installed, the required kernel headers are located under

  ```
  $ /usr/src/kernels/$(uname -r)/include/
  ```

If the kernel headers package cannot be found

Linux distributions regularly update the kernel package and might not keep the old kernel headers package in their package repos. If this happens, the easiest solution is to update the system to the latest kernel and then rerun the kernel headers install command.

To update the kernel to the latest version and install kernel headers, run the following commands (this requires a reboot):

```
$ sudo yum update kernel kernel-devel
```

```
$ reboot
```

For Oracle UEK

- To check whether the kernel headers are installed (any user can run this):

```
$ yum list kernel-uek-devel-$(uname -r)
```

- To install the necessary kernel headers:

  ```
  $ sudo yum install -y kernel-uek-devel-$(uname -r)
  ```

- When properly installed, the required kernel headers are located under

  ```
  $ /usr/src/kernels/$(uname -r)/include/
  ```

For SUSE or OpenSUSE

- To check whether the kernel headers are installed (any user can run this):

  ```
  $ zypper se -s kernel-default-devel | grep $(uname -r | sed "s/-default//") $ zypper se -s kernel-devel | grep $(uname -r | sed "s/-default//")
  ```

- The output should be like the following, where the `i+` signifies that the package is installed. If the left-hand column is `v` or is blank, the package must be installed.

  ```
  $ i+ kernel-default-devel | package | 4.12.14-lp150.12.25.1 | x86_64 | openSUSE-Leap-15.0-Update
  ```

- To install the necessary kernel headers:

  ```
  $ zypper install --oldpackage kernel-default-devel=$(uname -r | sed "s/-default//")
  ```

  ```
  $ zypper install --oldpackage kernel-devel=$(uname -r | sed "s/-default//")
  ```

- When properly installed, the required kernel headers are located under

  ```
  $/usr/src/linux-$(uname -r) | sed "s/-default//")/include/ | grep -f
  ```

For Ubuntu

- To check whether the kernel headers are installed (any user can run this):

  ```
  apt list linux-headers-$(uname -r)
  ```

- To install the necessary kernel headers:

  ```
  sudo apt install linux-headers-$(uname -r)
  ```

- When properly installed, the required kernel headers are located under

  ```
  /usr/src/linux-headers-$(uname -r)/include/
  ```

To Install a Linux Sensor on a Single Endpoint

You can install a Linux sensor on a single endpoint by following this procedure.

Procedure

1 Extract the contents of the installer package into a temporary directory.

2 Install and register the sensor by running the following command; replace '<COMPANY_CODE>' with your company registration code: `Sudo cb-psc-install/install.sh '<COMPANY_CODE>'`

To Install a Linux Sensor on an Endpoint using the RPM/DPKG Installer

You can install a Linux sensor on an endpoint by using this method.

Procedure

1 Extract the contents of the installer package into a temporary directory.

2 Install the RPM/DEB package.

RPM:

```
$ sudo rpm -i cb-psc-install/cb-psc-sensor-<BUILD-NUMBER>.x86_64.rpm
```

DEB:

```
$ sudo dpkg -i cb-psc-install/cb-psc-sensor-<BUILD-NUMBER>.x86_64.deb
```

3 Install the blades.

```
$ sudo cb-psc-install/blades/bladesUnpack.sh
```

4 Update the `cfg.ini` file with the v3.x+ company registration code.

```
$ sudo /opt/carbonblack/psc/bin/cbagentd -d '<COMPANY_CODE>'
```

5 `Start the agent.`

For CentOS/RHEL 6:

```
$ service cbagentd start
```

For all other distributions:

```
$ systemctl start cbagentd
```

To Install a Linux Sensor on an Endpoint that Automatically Registers the First Time it is Started

By using this method, the Linux sensor registers the first time it starts up.

Procedure

1 Extract the contents of the installer package into a temporary directory.

2 Use the `install.sh` script to install the agent, but do not provide a company code. `$ sudo cb-psc-install/install.sh`

3 Update the `cfg.ini` file with the v3.x+ company code. `$ sudo /opt/carbonblack/psc/bin/cbagentd -d '<COMPANY_CODE>'`

Results

Note The sensor is configured to register when the sensor starts up. This can occur on the next system boot or by restarting the agent. When the agent starts, the sensor will register itself with the Carbon Black Cloud backend.

Installing macOS Sensors on Endpoints

3

This section introduces ways to install macOS sensors on endpoints.

Important Before you begin the processes described here, read Chapter 1 Getting Started. It contains highly relevant information to help you succeed in your sensor installation.

Before you can install sensors, you must perform the following steps:

To Obtain a Company Registration Code

To Download Sensor Kits

This chapter includes the following topics:

macOS v3.1 Sensor on High Sierra and Later

For macOS v3.1 sensor installations on macOS 10.13, High Sierra requires initial KEXT approval of the product kernel extension by administrative policy or user.

This requirement is enforced by Apple. It applies to all third-party products that have a driver component. The sensor requires KEXT approval regardless of the previous KEXT approval status.

Note For macOS Big Sur, user KEXT approval is only part of the requirement; MDM approval is required for KEXT deployment on macOS Big Sur.

Carbon Black recommends that you pre-configure High Sierra endpoints with pre-approved drivers by using MDM policy, netboot, or pre-configured images. This approach simplifies sensor installation, especially during a command line installation. A CLI message occurs during the install, and requires the – `kext` flag to skip and finish the install.

If drivers are not pre-approved before sensor installation, the behavior is as follows:

Command line installation: Installation finalizes and returns success, but logs a warning to installation logs. Because drivers cannot load, the sensor enters Bypass state and reports this state to the cloud. After KEXT is approved, the sensor recovers within one hour and enters the full protection state.

Direct installation is handled similarly to a command line installation, with two differences: (1) sensor installation displays a dialog message that requests the user to approve the KEXT by using system preferences; (2) installer stalls for up 10 minutes to give the user the opportunity to approve the KEXT.

To Identify Devices with Sensors that do not support the Operating System or are KEXT Not-approved

These search procedures show you which sensors are running on unsupported operating systems or are not KEXT approved.

Procedure

1. Sign in to the Carbon Black Cloud Console.
2. On the navigation bar, click **Inventory** and then click **Endpoints**.
3. Change the **Status** filter to **All**, and type the following search query:
   `sensorStates:UNSUPPORTED_OS`
4. Use the following search query to help identify devices with sensors that do support the operating system, but with sensor KEXT or System Extension not approved:
   `sensorStates:DRIVER_LOAD_NOT_GRANTED`

Approving the Kernel Extension for macOS Sensor Version 3.1+

Carbon Black recommends submitting the applicable Carbon Black Defense KEXT IDs for approval by MDM before install or upgrade of macOS sensor version 3.0 or above.

However, if the KEXT is not pre-approved by MDM, you can approve KEXTs manually upon install or upgrade, unless you are installing on macOS Big Sur.

See CB Defense: How to approve KEXT on JAMF.

To Manually Approve the KEXT

This procedure lets you manually approve the KEXT.

Procedure

1. Install the sensor on the endpoint.
2. In **System Preferences**, in the **Security & Privacy** pane, click the **General** tab.
3. Authenticate as Administrator.

4 Click the **Allow** button for **System software from developer "Carbon Black" was prevented from loading**. The installer will finish running and load the sensor.

To approve the KEXT via MDM

Specify the Apple Team ID and KEXT bundle in your configuration profile.

Apple Team ID: `7AGZNQ2S2T`

KEXT Bundle ID: `com.carbonblack.defense.kext`

See How to approve Mac Sensor 3.0 KEXT for Install/Upgrade and Apple Technical Note TN245.

Security Enhancements in macOS 10.14.5+

As part of user data security enhancements in macOS 10.14.5 and above, you must approve access to protected user and application data.

This requirement is in addition to kernel extension approval, and does not replace that process.

Applications can be granted access to app data (such as photos, contacts, and calendars), protected services and devices (such as the microphone or camera), or user data (such as mail, cookies, and Safari history) via the **Security & Privacy System Preferences** pane. Access is granted by enabling individual access, or by allowing full disk access. For the macOS sensor to operate at full functionality on an endpoint that is running macOS 10.14.5+, the sensor must have full disk access.

To be completely effective, the macOS sensor must be granted access to protected user and application data. This can be done manually on each endpoint, or for quicker and more consistent endpoint management, these settings can be managed through the creation and deployment of a mobile device management (MDM) profile.

To Manually Grant the pre-3.5.1 Sensor Full Disk Access

This procedure grants full disk access to pre-3.5.1 sensor versions.

1 Go to the **Security & Privacy System Preferences** section and click the **Privacy** tab.

2 After you are authenticated as an Administrator, scroll down to the **Full Disk Access** section and click the **Plus (+)** button to add an application. Select **/Applications/Confer.app**.

3 Restart Confer.app. After the restart, `com.carbonblack.defense.ui` will appear in the allowed applications list.

To manually grant the 3.5.1 or later sensor full disk access

This procedure grants full disk access to 3.5.1 or later sensor versions.

1 Open the Finder and go to `/Applications/VMware Carbon Black Cloud/`.

2 Right-click `repmgr.bundle` and select **Show Package Contents**. Go to `repmgr.bundle/Contents/macOS` and drag the `repmgr` executable into the **Full Disk Access Privacy** tab.

3 Repeat the previous steps for `uninstall.bundle`, and `LiveQuery.bundle` (the executable is called osqueryi): `uninstall.bundle/Contents/macOS/uninstall`

`LiveQuery.bundle/Contents/macOS/osqueryi`

To Grant the Sensor Full Disk Access via MDM

The easiest way to distribute the necessary Privacy Preference payload is to upload the MDM-privacyconfig.mobileconfig file, which is in the mounted DMG of the installer in the docs folder.

The following steps recreate the mobileconfig in your MDM.

These instructions were created using Apple documentation and were validated in Jamf PRO and WorkspaceONE UEM using sensor version 3.5.0.30. Field names, values, and functionality vary depending on the MDM framework or sensor version.

Granting an application full disk access is accomplished via a Privacy Preferences payload. The Carbon Black Cloud Sensor requires five identifiers in this Privacy payload.

The fields should be completed exactly as follows. Copy and paste for accuracy.

Identifier: com.vmware.carbonblack.cloud.daemon

Identifier Type: Bundle ID

Code Requirement:

```
identifier "com.vmware.carbonblack.cloud.daemon" and anchor apple generic
            and certificate 1[field.1.2.840.113635.100.6.2.6] /* exists */ and
            certificate leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and
            certificate leaf[subject.OU] = "7AGZNQ2S2T"
```

App or Service: SystemPolicyAllFiles

Access: Allow

Identifier: com.vmware.carbonblack.cloud.osqueryi

Identifier Type: Bundle ID

Code Requirement:

```
identifier "com.vmware.carbonblack.cloud.osqueryi" and anchor apple generic
            and certificate 1[field.1.2.840.113635.100.6.2.6] /* exists */ and
            certificate leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and
            certificate leaf[subject.OU] = "7AGZNQ2S2T"
```

App or Service: SystemPolicyAllFiles

Access: Allow

Identifier: com.vmware.carbonblack.cloud.se-agent.extension

Identifier Type: Bundle ID

Code Requirement:

```
identifier "com.vmware.carbonblack.cloud.se-agent.extension" and anchor apple generic
            and certificate 1[field.1.2.840.113635.100.6.2.6] /* exists */and
            certificate leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and
            certificate leaf[subject.OU] = "7AGZNQ2S2T"
```

App or Service: SystemPolicyAllFiles

Access: Allow

Identifier: com.vmware.carbonblack.cloud.uninstall

Identifier Type: Bundle ID

Code Requirement:

```
identifier "com.vmware.carbonblack.cloud.uninstall" and anchor apple generic
            and certificate 1[field.1.2.840.113635.100.6.2.6] /* exists */ and
            certificate leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and
            certificate leaf[subject.OU] = "7AGZNQ2S2T"
```

App or Service: SystemPolicyAllFiles

Access: Allow

Identifier: com.vmware.carbonblack.cloud.uninstallerui

Identifier Type: Bundle ID

Code Requirement:

```
identifier "com.vmware.carbonblack.cloud.uninstallerui" and anchor apple
            generic and certificate 1[field.1.2.840.113635.100.6.2.6] /* exists */ and
            certificate leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and
            certificate leaf[subject.OU] = "7AGZNQ2S2T"
```

App or Service: SystemPolicyAllFiles

Access: Allow

Support for macOS 10.15 Catalina

Beginning with macOS 10.15, a system reboot is required for newly-installed KEXTs to load. Factor this reboot requirement into your deployment work flow.

Endpoints that require a reboot report that state on the **Dashboard** or **Endpoints** page; search for `sensorStates:DRIVER_INIT_REBOOT_REQUIRED` on the **Endpoints** page to find 10.15 devices that are in bypass mode and require a reboot.

macOS Sensor for Big Sur

Carbon Black Cloud sensor version 3.5.1.x provides initial support for macOS Big Sur. The following sections describe prerequisites and installation procedures for Big Sur.

Configuring MDM for the macOS Sensor for Big Sur

MDM configurations are required to mass-deploy the VMware Carbon Black Cloud sensor for Big Sur.

Detailed instructions explain how to install all necessary MDM configurations for the unattended install of the sensor:

- KEXT approval
- System Extension approval
- Network Extension approval
- Privacy Preferences (Full Disk Access) approval

Current MDM instructions are available in the sensor's mounted DMG. Your mount point might be slightly different than what is shown here:

`/Volumes/CBCloud-3.5.1.19/docs/MDM-instructions.txt`

Approving the KEXT via MDM for Big Sur

The easiest way to distribute the necessary MDM payload to approve the KEXT is to upload the `MDM-KEXT-approval.mobileconfig` file, which is located in the mounted DMG of the installer in the docs folder.

You can also recreate the attached mobileconfig in your MDM tool by specifying the Apple Team ID and the KEXT Bundle ID in your Kernel Extension configuration profile:

- Apple Team ID: `7AGZNQ2S2T`
- KEXT Bundle ID: `com.carbonblack.defense.kext`

To allow the KEXT to load on MacOS Big Sur, the OS either requires a local action from an admin to approve the KEXT after install or a customized reboot command from your MDM to rebuild the Kernel Cache.

Your MDM must support custom XML to use the following method. If your MDM provider does not support custom XML, use the local approval method to run the KEXT.

The easiest way to distribute the necessary MDM command is to upload the `MDM-KEXT-reboot-command.xml` file, which is found in the mounted DMG of the installer in the docs folder. This XML file should be uploaded as a Custom Command and sent to endpoints after KEXT install. The target machine will reboot without warning; this distribution method is a temporary workflow until MDM providers update their reboot protocols to support `RebuildKernelCache`. This command is here:

```
<dict>
   <key>RebuildKernelCache</key>
   <true/>
   <key>KextPaths</key>
   <string>/Library/Extensions/CbDefenseSensor.kext</string>
   <key>RequestType</key>
   <string>RestartDevice</string>
</dict>
```

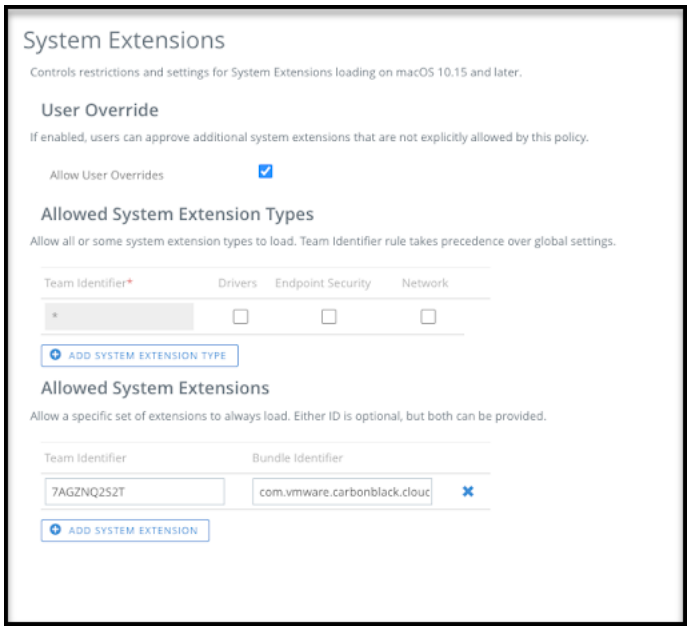
Approving the System Extension via MDM for Big Sur

The manual steps to create the correct mobileconfig in your MDM are listed here.

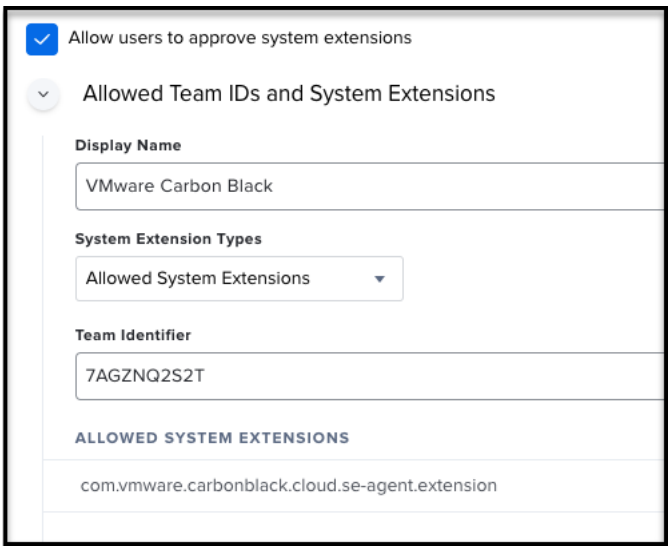
Specify the Apple Team ID and System Extension bundle Identifier in your Allowed System Extension configuration profile:

- System Extension Types: `Allowed System Extensions`
- Apple Team ID: `7AGZNQ2S2T`
- System Extension Bundle ID: `com.vmware.carbonblack.cloud.se-agent.extension`

The Workspace One configuration should look like the following:



The JAMF configuration should look like the following:



Approving the Network Extension Component of the System Extension via MDM for Big Sur

You can grant the System Extension the ability to Filter Network Content via a Web Content Filter configuration profile.

Note These instructions were created using Apple documentation and ProfileCreator (https://github.com/ProfileCreator/ProfileCreator). Field names, values, and functionality vary depending on the MDM framework or sensor version.

After creating this profile, the profile should be signed to enable distribution via MDM.

The fields should be completed exactly as follows. Copy and paste for accuracy.

In the General payload:

- **Payload Scope**: `System`

In the Web Content Filter payload:

- **Filter Type**: `Plug-In`
- **Plug-In Bundle ID**: `com.vmware.carbonblack.cloud.se-agent`
- Check **Enable Socket Filtering**
  - **Filter Data Provider System Extension Bundle ID (macOS)**: `com.vmware.carbonblack.cloud.se-agent.extension`
  - **Filter Data Provider Designated Requirement (macOS)**: identifier `"com.vmware.carbonblack.cloud.se-agent.extension" and anchor apple generic and certificate 1[field.1.2.840.113635.100.6.2.6] /* exists */ and certificate leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and certificate leaf[subject.OU] = "7AGZNQ2S2T"`

- Check **Enable Packet Filtering (macOS)**
  - **Filter Packet Provider System Extension Bundle ID (macOS)**: `com.vmware.carbonblack.cloud.se-agent.extension`
  - **Filter Packet Provider Designated Requirement (macOS)**: `identifier "com.vmware.carbonblack.cloud.se-agent.extension" and anchor apple generic and certificate 1[field.1.2.840.113635.100.6.2.6] /* exists */ and certificate leaf[field.1.2.840.113635.100.6.1.13] /* exists */ and certificate leaf[subject.OU] = "7AGZNQ2S2T"`

Unattended Fresh Install of a System Extension Sensor on Big Sur with MDM Configurations

MDM configurations are required to mass-deploy the System Extension sensor on Big Sur.

See Configuring MDM for the macOS Sensor for Big Sur.

Run the `cbcloud_install_unattended.sh` script. Sensor installation can be automated over many endpoints by using the following command. Your mount point may be slightly different:

```
sudo /Volumes/CBCloud-3.5.1.19/docs/cbcloud_install_unattended.sh -i /Volumes/
CBCloud-3.5.1.19/CBCloud\ Install.pkg -c [Company Registration Code] -e
```

Attended Fresh Install of a System Extension Sensor on Big Sur (no MDM Configurations)

On macOS 11, the attended installer defaults to installing a System Extension sensor.

To install into KEXT, see Approving the KEXT via MDM for Big Sur.

To perform an attended install of the System Extension sensor

1 Mount the `CBCloud.dmg` and double-click **CBCloud Install**.

2 Follow the prompts through the attended installation.

3 During the **Running package scripts** steps, two approvals must be made: **VMware CBCloud**: Go to the **System Preferences > Security & Privacy > General** tab. Authenticate as an administrator and click the **Allow** button next to the following message: **System software from application "VMware CBCloud" was blocked from loading** . Allow **'VMware CBCloud' Would Like to Filter Content**.

4 The installation should finish. However, the System Extension must be granted Full Disk Access to function completely. Go to **System Preferences > Security & Privacy > Privacy tab > Full Disk Access**. Authenticate as an administrator and select `com.vmware.carbonblack.cloud.se-agent.extension.systemextension`.

5 The remaining sensor executables must be granted Full Disk Access. Open the Finder and go to `/Applications/VMware Carbon Black Cloud/`.

6 Right-click `repmgr.bundle` and select **Show Package Contents**. Go to `repmgr.bundle/Contents/macOS` and drag the `repmgr` executable into the **Full Disk Access Privacy** tab.

7 Repeat the previous steps for `uninstall.bundle`, and `LiveQuery.bundle` (the executable is called osqueryi):

```
uninstall.bundle/Contents/macOS/uninstall
LiveQuery.bundle/Contents/macOS/osqueryi
```

Installing a KEXT-enabled Sensor on Big Sur

On macOS 11, the attended installer defaults to installing a System Extension sensor. To install into KEXT mode, we recommend using the `cbcloud_install_unattended.sh` unattended install script, which is found in the mounted DMG of the sensor installer in the docs folder.

A new `–k` flag in `cbcloud_install_unattended.sh` signifies a KEXT sensor install. This flag also works during an upgrade. See macOS Command Line Parameters.

For Kernel Extensions (legacy System Extensions) to run on macOS Big Sur, Apple has added two new restrictions for new installs or upgrades:

- Kernel Extensions must be pre-approved via MDM.
- Kernel Extensions must be approved manually, and the OS requires a reboot after install.

To install a KEXT-enabled Sensor on Big Sur

1 Run the cbcloud_install_unattended.sh script. Your mount point may be slightly different than what is shown here: `sudo /Volumes/CBCloud–3.5.1.19/docs/cbcloud_install_unattended.sh –i /Volumes/CBCloud–3.5.1.19/CBCloud\ Install.pkg –c [Company Registration Code] –k`

2 Before the install finishes, a window appears stating that a System Extension has been updated. Approve this prompt in the **Security & Privacy** pane of **System Preferences**, or follow these steps to automate this KEXT approval through MDM: Approving the KEXT via MDM for Big Sur. The installation might report a failure here if the user did not approve KEXT in time. The install can still be completed despite this reported failure.

3 Restart the OS to finish installing the new KEXT.

4 If Full Disk Access has not already been granted through MDM, see step 5 of the Attended Fresh Install of a System Extension Sensor on Big Sur (no MDM Configurations) to finish granting the sensor executables Full Disk Access.

Validating a Healthy System Extension Sensor through RepCLI on Big Sur

Follow this procedure to validate a healthy System Extension Sensor on Big Sur.

1 Run the following command: `sudo /Applications/VMware\ Carbon\ Black\ Cloud/repcli.bundle/Contents/macOS/repcli status`

2 Expected results:

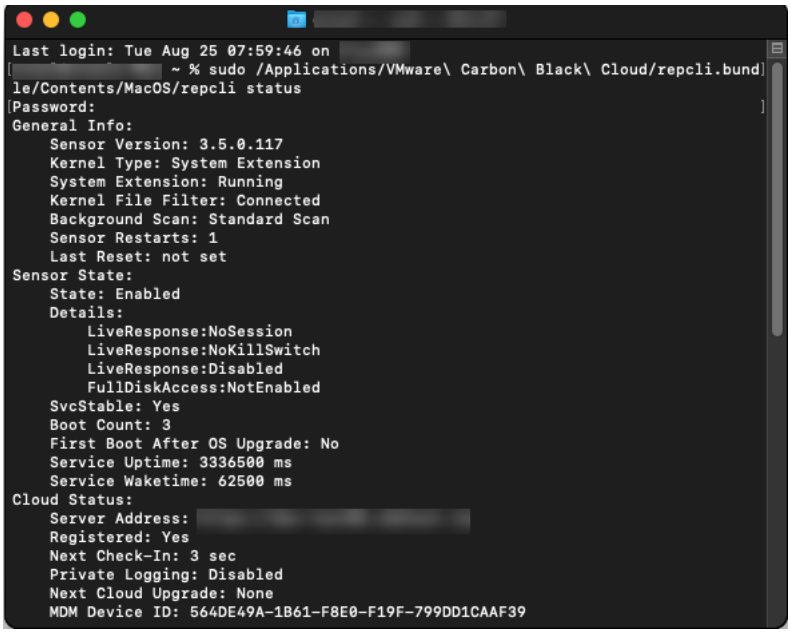
   a General Info

      - Kernel Type: System Extension

- System Extension: Running
- Kernel File Filter: Connected

b Sensor State

- State: Enabled
- SvcStable: Yes (Might take a few minutes after install to reach SvcStable)
- Cloud Status:
- Registered: Yes



Upgrading the macOS Sensor on Big Sur

On Big Sur, to switch between kernel types during an upgrade, run the `cbcloud_install_unattended.sh` script with either the `–k` or `–e` flag.

The `–k` flag will force a Kernel Extension sensor. The `–e` flag will force a System Extension sensor. See macOS Command Line Parameters.

Your mount point might be slightly different than what is shown here: `sudo /Volumes/CBCloud–3.5.1.19/docs/cbcloud_install_unattended.sh –i /Volumes/CBCloud–3.5.1.19/CBCloud\ Install.pkg –c [Company Registration Code] –k`

Toggling between Kernel Extension and System Extension in Big Sur

We highly recommend that you perform the toggle command after you have configured MDM for both Kernel Extension and System Extension in Big Sur.

Without an MDM configuration, see Attended Fresh Install of a System Extension Sensor on Big Sur (no MDM Configurations).

Your organization's deregistration code is required to run the toggle command. In the VMware Carbon Black Cloud console, go to **Inventory > Endpoints > Sensor Options > View Company Codes**. In this context, the code does not uninstall anything and is used as an administrative code to enable the repCLI tool.

To toggle from System Extension to Kernel Extension

1 Run the following command: `sudo /Applications/VMware\ Carbon\ Black\ Cloud/repcli.bundle/Contents/macOS/repcli setsensorkext [deregistration code]`

2 The KEXT must be manually approved. A window will appear stating that a System Extension has been updated. Approve this prompt in the **Security & Privacy** pane of **System Preferences**.

3 Restart the OS.

To toggle from Kernel Extension to System Extension

1 Run the following command: `sudo /Applications/VMware\ Carbon\ Black\ Cloud/repcli.bundle/Contents/macOS/repcli setsensorsysext [deregistration code]`

2 Restart the OS.

Uninstall of the Sensor on Big Sur

In Big Sur, you can uninstall the sensor through the Carbon Black Cloud console, or you can automate an uninstall over many endpoints.

To perform an unattended uninstall of the sensor on Big Sur

1 Run the following command: `sudo /Applications/VMware\ Carbon\ Black\ Cloud/uninstall.bundle/Contents/MacOS/uninstall –y`

2 If the **Require code to uninstall sensor** option is enabled, run the following command: `sudo /Applications/VMware\ Carbon\ Black\ Cloud/uninstall.bundle/Contents/MacOS/uninstall –y –c <Uninstall Code>`

3 If the sensor was installed in KEXT mode, you must reboot the endpoint to fully remove the unloaded KEXT.

To perform an attended uninstall of the sensor on Big Sur

1 Mount the `CBCloud.dmg` and double-click **CBCloud Uninstall**.

2 Proceed through the attended uninstallation prompts. You must authenticate as admin.

3 If the sensor was installed in KEXT mode, you must reboot the endpoint to fully remove the unloaded KEXT.

macOS Sensor Command Line Install

The `CB Defense Install.pkg` and `cbdefense_install_unattended.sh` scripts are part of the macOS sensor release and are embedded in the CB Defense DMG. Both files are required for command line installations on macOS endpoints.

Note Instructions on how to create custom packages for software distribution tools is beyond the scope of this article. Carbon Black provides generic instructions on how to install the `CB Defense Install.pkg` payload on the command line, with the help of the `cbdefense_install_unattended.sh` utility script. You can adapt these instructions to a software distribution tool.

macOS utility script

The utility script can be used in the following ways:

- As-is (passed command line options to customize the install process).
- Modified to hard-code the install options and simplify the installation.
- Used as an example or guide on how to create a custom script.

A common installation method is to use the utility script as-is, push the script and the PKG payload onto the target device (both files can be bundled in a custom package), and then execute the utility script.

To Extract and Prepare the macOS Install Files

Before you can install sensors, you must extract and prepare the macOS install files.

Procedure

1. Click **CB Defense DMG** or mount it by using system tools. DMG is mounted to the `/Volumes/CBCloud–X.X.X.X` directory (where X.X.X.X refers to the sensor version).
2. Alternatively, use the `hdiutil` command to mount the downloaded sensor release disk image; for example: `hdiutil attach /path/to/CBCloud_Installer_mac_X.X.X.X`
3. Extract the `CBCloud Install.pkg` file from the mounted volume `/Volumes/CBCloud–X.X.X.X` directory. The `.pkg` file is the sensor installer payload.
4. Extract the `cbcloud_install_unattended.sh` utility script from the `/Volumes/CBCloud–X.X.X.X/docs/` directory.
5. The mounted volume can be unmounted because it is not needed for the remainder of the steps. You can unmount it by using `Finder`, or by running the following command: `hdiutil eject /Volumes/CBCloud–X.X.X.X`
6. Use the extracted `CBCloud Install.pkg` and `cbcloud_install_unattended.sh` files to create a custom package that is compatible with your software distribution tool, or install the two files directly onto the target macOS device.

Results

Note `cbcloud_install_unattended.sh` and the `CBCloud Install.pkg` payload must be extracted from the same major and minor version of released DMG file to ensure compatibility between the utility and the installer payload. If the two files do not originate from the same release, the installation might fail.

Typically, the extracted `cbcloud_install_unattended.sh` and the `CBCloud Install.pkg` files are pushed to the target server endpoints. They can be used to create a custom installation bundle that is compatible with a specific software distribution tool.

Note You must always wrap the company registration code in single quotation marks. Double quotation marks are not an acceptable substitute to single quotes.

To Perform a macOS Sensor Command Line Installation

Follow this procedure to install a macOS sensor from the command line.

Procedure

1. Extract the files from a sensor release DMG file.
2. Optionally, create a custom wrapper package bundle that is compatible with the selected software distribution tool. The custom package embeds the `CBCloud Install.pkg` file, together with a utility to set up options and start the sensor PKG installation.
3. Install the sensor installer on the endpoint by using the supported options.

macOS Command Line Parameters

The following common command line parameters are supported by the `cbdefense_install_unattended.sh` utility script. The parameters are passed on to the installer.

Note The `-c` and `-i` parameters are the only required options for a command line installation. Parameter values must always be enclosed in single quotes.

To view all command line parameters, run the command together with the `–h` parameter.

Parameter	Required or Optional	Description
`–c COMPANY CODE`	Required	Company registration code.
`–i PKG_FILE`	Required	Absolute path to the PKG installer payload.
`–d`	Optional	Enter bypass mode (disabled protection) immediately after installation. You can enable protection at a later time. This mode is only recommended for test situations.
`–g POLICY_NAME`	Optional	Specify a policy to which the sensor will be added.

Parameter	Required or Optional	Description
–p PROXY_SERVER:PORT	Optional	Preferred Proxy server and port; for example: -p '10.5.6.7:54443' Multiple proxy servers can be provided and separated by semi-colons; for example: - p '10.5.6.8:54443;10.5.6.7:54443' If a proxy server/port are not specified but are required, the sensor will attempt proxy auto-detection. See Configure a Proxy.
–x PROXY_USER:PASSWORD	Optional	Proxy credentials to use for the proxy server, if required. These apply whether the proxy server is auto-detected or specified. Example: - `x 'proxy_user:proxy_password'` If proxy credentials are not specified, but are required by the proxy server, the macOS sensor will attempt to detect and use proxy credentials that are stored in the keychain that match the detected or specified proxy server.
–h		Displays all command line options, including advanced options that are not documented here. Refer to the built in help in the `cbdefense_install_unattended.sh` utility script for currently supported installation options.
–e	Optional	Forces System Extension install on macOS Big Sur (the sensor will default to this mode on Big Sur and does not need to be explicitly specified).
–k	Optional	Forces Kernel Extension install on macOS Big Sur (pre-approvals must be in place).

Obfuscation of command line inputs

Endpoint users might input sensitive data into the command line. The obfuscation of command line inputs protects against unauthorized users accessing the data in plain text in the sensor `.log` files and the sensor databases. You can obfuscate command line inputs by using the following argument in the unattended install script: `––enable–hide–command–lines=1`

The setting enables the obfuscation of command line input in sensor `.log` files and databases. The data in the Carbon Black Cloud console is not obfuscated.

macOS Command Line Install Examples

Review the following examples for macOS command line installations.

The following commands should be on a single line.

The following examples assume that the required files are installed to the target device `/tmp/` directory.

To run a command line install with required parameters

```
sudo /tmp/cbcloud_install_unattended.sh –i '/tmp/CBCloud Install.pkg' –c 'XYZ'
```

To specify a policy for the sensor

```
sudo /tmp/cbcloud_install_unattended.sh –i '/tmp/CBCloud Install.pkg' –c 'XYZ' –g 'Monitored'
```

To Address the Extension Warning Post-install

A post-install warning can occur after installing a macOS sensor. This procedure resolves the problem.

Procedure

1. Download the installer: macOS.
2. When prompted to approve the CB Defense kernel extension, click **OK**.
3. When the **System Extension Blocked** message appears, click **Open Security Preferences**.
4. Click **Allow** next to **System software from developer "Carbon Black, Inc." was blocked**.
5. Double-click the Carbon Black icon and copy/paste the installation code from a text editor.

macOS Services, Utilities, and Uninstaller

Review the macOS sensor services, utilities, and uninstaller files that reside on the endpoints.

macOS installed services for 3.5.0 and lower

- Sensor Driver Bundle: `/System/Library/Extensions/CBDefenseSensor.kext`
- Sensor Service: `/Applications/Confer.app/Contents/MacOS/repmgr`
- Sensor UI: `/Applications/Confer.app/Contents/MacOS/CBDefense`

macOS installed services for 3.5.1 and higher

- Sensor Driver Bundle: `/Applications/VMware Carbon Black Cloud/`
- Sensor data directories: `/Library/Application Support/com.vmware.carbonblack.cloud/`
- Sensor Service: `/Applications/VMware Carbon Black Cloud/repmgr.bundle/Contents/MacOS/repmgr`
- Sensor UI: `/Applications/VMware Carbon Black Cloud/CBCloudUI.bundle/Contents/MacOS/CBCloudUI`

macOS installed utilities

- Uninstaller helper: `/VMware Carbon Black Cloud/uninstall.bundle/Contents/MacOS/uninstall`
- Upgrade helper: `/VMware Carbon Black Cloud/UpgradeHelper.bundle/Contents/MacOS/UpgradeHelper`
- RepCLI: `/VMware Carbon Black Cloud/repcli.bundle/Contents/MacOS/repcli`

macos uninstaller

- 3.X: `/Applications/Confer.app/uninstall`
- 1.X sensor: `/Applications/Confer.app/uninstall.sh`

- CLI_USERS=sid #Required, needed to interact with sensor locally

Installing Sensors on Endpoints in a VDI Environment

4

This section describes how to install sensors through the command line or software distribution tools in a Virtual Desktop Infrastructure (VDI) environment.

Important Before you begin the processes described here, read Chapter 1 Getting Started. It contains highly relevant information to help you succeed in your sensor installation.

Before you install sensors, perform the following steps:

To Obtain a Company Registration Code

To Download Sensor Kits

Review the following requirements and implement the recommended best practices. If the best practices included here are not the preferred method for deployment, an alternative configuration that uses the OFFLINE_INSTALL switch is also supported for Windows sensors v3.5 and above. This is useful for organizations who want to create a primary image and clone it to offline computers. See CB Defense: How to Perform Offline Installation of Sensor and Windows Sensor Supported Commands.

For firewall and proxy information, see Chapter 11 Configuring Carbon Black Cloud Communications.

This chapter includes the following topics:

- VDI Requirements
- Non-Persistent VDI
- Persistent VDI Install
- Persistent VDI and Non-Persistent VDI Mixture
- Horizon Linked-Clones and VMware Carbon Black Cloud 3.6 Sensor Deployment Best Practices

VDI Requirements

Before you install sensors in a VDI environment, confirm that your environment meets the minimum requirements.

- Carbon Black Cloud sensors: Windows sensor v3.5+
- Supported
  - Install layer: OS

    VDI method: Linked clones, full clones
- Preview Support:
  - Install layer: OS

    VDI method: Instant clones.
- Not Supported
  - Install layer: Application

    VDI method: Application volumes, application layering

Important

- Support for Instant Clones is enabled by leveraging the Carbon Black Cloud 3.6+ sensor in conjunction with Horizon version 7.13+.
- Carbon Black Cloud sensor 3.5.x, and Horizon <=7.12 enable support in a Preview Capacity. Carbon Black Cloud sensor <=3.4.x are not supported.
- See VMware Horizon KB - Interoperability of VMware Carbon Black and Horizon for further detail.

Note The sensor on the primary image must be installed from the command line with RepCLI execution enabled and without the VDI=1 parameter. VDI=1 is deprecated in sensor versions 3.4+. See Windows Sensor Supported Commands

Non-Persistent VDI

This section describes concepts and settings to consider when are you are deploying non-persistent VDIs.

VDI Policy Settings

Carbon Black Cloud console administrators should create specific policies to manage VDI endpoints.

After a policy is applied to the primary image, all clones inherit this policy unless otherwise directed by membership in sensor groups.

For more information about sensor groups, see the *VMware Carbon Black Cloud User Guide*.

If the primary image has no assigned policy, clones are assigned to the Virtual Desktops policy (if it exists) or the Standard policy, in that order.

The following policy settings are recommended for non-persistent VDIs.

General Tab

- **Name** – Virtual Desktops . Note that “Virtual Desktops” was previously a prescribed policy name. Now, you can put VDI clones into any policy name and support clones in different policies. This allows you to segregate non-persisted clones from persistent/physical machines and have different settings for each type.
- **Description** – This policy is optimized for VDI endpoints. Special considerations improve performance and provide a strong base of reputation, behavioral, and targeted prevention.
- **Target Value** – Medium

Sensor Tab

- **Display sensor message in system tray** - Enable this setting and add a message similar to this sample text: "Virtual Desktops Policy - Contact someone@example.com with any questions and concerns. Provide context regarding the issue and any available replication steps."

Prevention Tab - Permissions

- **Bypass rules (exclusions)** – Policy-level bypass rules help achieve stability in a VDI environment.

  Each organization must understand the trade-offs between performance and security. VDI vendors recommend the use of exclusions. It is recommended that you review your specific vendor’s VDI best practices before you implement a solution. Work with stakeholders to review risks and benefits (performance versus visibility) and apply the bypass rules as needed.

  VMware Carbon Black provides exclusions for supported methods as examples. Additionally, please review the applications that are installed in the VDI environment and apply any required bypass rules. For additional assistance, contact your VMware Carbon Black Technical Representative.

  Bypass references: Permission Allow | Allow & Log | Bypass (also known as AV Exclusions)

- CB Defense: How to Create Policy Blocking & Isolation and Permissions Exclusions
- CB Defense: How to Set up Exclusions for AV Products
- CB Defense: How to Use Wildcards in Policy Rules

The following examples are based on public documentation for Citrix and VMware solutions:

Citrix bypass rules best practices

```
**\Program Files*\Citrix\**,
**\AppData\Local\Temp\Citrix\HDXRTConnector\*\*.txt,
**\*.vdiskcache,
**\System32\spoolsv.exe
```

Note Additional bypass rules might be required. For example, some organizations do not want to bypass `winlogon.exe`. This is a Citrix recommendation for any AV solution because a common problem with VDIs that use AV is longer login times. This bypass rule helps restore the expected experience.

VMware bypass rules best practices

```
**\Program Files\VMware\**,
**\SnapVolumesTemp**,
**\SVROOT**,
**\SoftwareDistribution\DataStore**,
**\System32\Spool\Printers**,
**\ProgramData\VMware\VDM\Logs**,
**\AppData\VMware\**
```

Note Additional bypass rules might be needed; see VMware - Antivirus Considerations in a VMware Horizon 7 Environment.

Prevention

Blocking and Isolation

Best practices recommend applying Blocking and Isolation rules to address specific attack surfaces. To get started, we recommend that you duplicate the Standard policy rules to the Virtual Desktops policy. To learn how to modify rules, apply methodologies and practices that are in the VMware Carbon Black User Exchange:

- Threat Research
- Endpoint Standard: Achieving Good, Better and Best Policies
- Training & Certification

Local Scan tab

- **On Access File Scan Mode** – Disabled
- **Allow Signature Updates** – Disabled

It is a best practice is to disable **Allow Signature Updates**. The local scan feature adds network overhead and augments resource utilization. The Carbon Black Cloud can pull reputation and enforce policy in real time from the Cloud because most VDI environments maintain 99% uptime.

However, you can install the signature pack to the primary image. This installation avoids the performance penalty of running updates on each clone but allows the clones to have some offline protection. Malware that can be identified by the signature pack on the primary image is detected and blocked independent of cloud activity.

Sensor tab

The following settings are specific to VDI. See VMware Carbon Black Cloud Endpoint Standard Policy Best Practices for additional settings.

- **Run Background Scan** – False To optimize performance, most VDI vendors recommend disabling any background scan of the file system. Operating under the expectation that the primary image is free of malware, and the clones maintain consistent connectivity to the Cloud, it is not recommended to utilize the background scan feature. Reputation is derived from the Cloud at execution when necessary, per policy configuration. See the following **Delay Execute for Cloud scan** recommendation. For optimal clone performance, run the background scan on the primary image because that prepopulates the sensor cache. A background scan does take some time to complete and not all users will want to wait for the scan when creating a new image. For performance sensitive customers, the extra wait time might be worth it if the image is intended to be deployed at scale. See How to Run a Background Scan in a Non-Persistent VDI Environment for alternative configurations.
- **Scan files on network drives** – Disabled
- **Scan execute on network drives** – Disabled
- **Delay execute for Cloud scan** – Enabled This critical setting serves as the sole point of reference for pre-execution reputation lookups. If it is disabled, endpoints must rely on **Application at Path** and Deny List rules for pre-execution prevention.
- **Hash MD5** – Disabled. The sensor always calculates the SHA-256.
- **Auto-deregister VDI sensors that have been inactive for** – Non-persistent VDI endpoints should be managed in a separate policy from Persistent VDIs. In Non-persistent VDI policies, we recommend that you enable this setting to remove any clones that been inactive for the specified duration. In Persistent VDI policies, we recommend that you disable this setting to prevent unintentional uninstall of the sensor.

Primary Image Considerations

This topic contains considerations and recommendations for the primary image.

Make sure that the primary image never registers as a clone or gets deregistered.

Install the sensor from the command line using the following switches:

```
COMPANY_CODE=" ABCEZGYXWT9PN3REXYZ"

CLI_USERS= sid
```

We recommend that you use this parameter on the primary image to enable RepCLI usage on the clones. The value is the Security Identifier (SID) of the user account/group that will run the `reregister now` command on the clones.

Note With the Windows 3.6+ sensor, `repcli reregister` is not authenticated. Users do not have to be members of the SID to run that command. However, we recommend that you set a CLI_USERS SID in case repcli is needed in support scenarios where the sensor cannot connect to the cloud.

See Windows Sensor Supported Commands.

Clone Considerations

This topic describes how the VDI clone receives a DeviceID and is assigned to a policy.

When `reregister now` is run, a clone performs the following operations:

1 The endpoint requests a new DeviceID.

2 The new DeviceID is identified as a VDI endpoint on the backend.

3 The endpoint inherits the policy from the primary image unless you have previously created sensor groups, and the installed sensor matches a sensor group's criteria. Manual policy assignment post-installation overrides the inheritance.

Non-Persistent VDI Install

This topic describes how to perform a non-persistent VDI install, together with recommendations and best practices.

Important

- Use the following method for installation in environments that are leveraging Horizon version <= 7.12 or another VDI technology.
- For non-persistent deployments leveraging Horizon version 7.13+, and Carbon Black Cloud sensor version 3.6+, see the VMware KB Article for deployment instructions.

Procedure

1 Install the primary OS and required applications.

2 Install the sensor with the CLI_USERS parameter to ensure RepCLI authentication. For example:

```
msiexec.exe /q /i C:\temp\installer_vista_win7_win8-64-3.4.0.xxxx.msi /L* log.txt
COMPANY_CODE=" XYZ" CLI_USERS= sid
```

where: COMPANY_CODE= < company code>

CLI_USERS= < desired user sid>

3 Schedule the command `repcli reregister now` to run on the clones — not on the primary image. Run this as a scheduled task or Group Policy action upon login, preferably from a batch file. Adjust the task to run before the network interface is available. Change PRIMARY to the computer name of the primary machine. For example:

```
@echo off
:: First check if Carbon Black Cloud sensor is installed, this will
also
log the current RegistrationId and DeviceGUID if installed
reg query HKLM\software\cbdefense
if %errorlevel% NEQ 0 ( echo Carbon Black Cloud sensor is not installed
& goto :eof )

:: Please mention the name of primary image if the Parent VM name is
not
PRIMARY_IMAGE
if /i %computername% == PRIMARY_IMAGE (
echo Time[%time%]: Skipping reregistration on primary image
%computername%
) else (
call :reregister 0
)
goto :eof

:reregister

:: Since Carbon Black Cloud sensor service may be in the process of
starting up when cloning customization script runs, we retry the command
up to 5 times before giving up
set /A attempt_number = %1
set /A attempt_number = attempt_number + 1
if %attempt_number% GEQ 5 (
echo Time[%time%]: Reached maximum number of
attempts[%attempt_number%] & exit /B 1
) else (
echo Time[%time%]: Scheduling reregister of Machine[%computername%]
AttemptNumber[%attempt_number%]
)

:: Check if reregister is already scheduled
reg query HKLM\system\currentcontrolset\services\cbdefense /v
ReregisterTime > nul
if %errorlevel% EQU 0 ( echo Time[%time%]: Carbon Black Cloud sensor is
scheduled to reregister & exit /B 0)

:: Try and write to registry key to schedule reregister. That way if CB
isn't running at time of clone the reregister will still happen if/when
CB starts again
reg add HKLM\System\CurrentControlSet\Services\CbDefense /v
ReregisterTime /t REG_QWORD /d 0 /f
set reg_error= %errorlevel%
if %reg_error% EQU 0 ( echo Time[%time%]: Scheduled reregister via
registry & exit /B 0)
```

```
"c:\Program Files\Confer\RepCLI.exe" reregister now
set repcli_error= %errorlevel%
if %repcli_error% EQU 0 (
echo Time[%time%]: Scheduled reregister via repcli & exit /B 0
) else (
echo Time[%time%]: Command Failed: %repcli_error% & call :reregister
%attempt_number%
)
goto :eof
```

4 Shut down the machine.

5 Create the primary image/VM template. Deployed clones register as separate devices and are assigned a new DeviceID when the `reregister` command is run, and the sensor connects to the backend.

6 To create a new template, repeat steps 4-5.

Note

- Make changes as often as necessary to the primary image. When making changes to the primary image, make sure that task/batch files are not disrupted.
- Install the sensor on the primary machine as normal, with RepCLI execution enabled.
- Persistent VDI environments follow the same best practices as physical endpoints.
- If the environment consists of a mixture of persistent and non-persistent VDIs, see Persistent VDI and Non-Persistent VDI Mixture.

Persistent VDI Install

If you are installing into an environment that is comprised of persistent VDIs only, apply the following best practices and use the following installation method.

Procedure

1 Install the primary OS and required applications.

2 Install the sensor with the BASE_IMAGE and CLI_USERS parameter to ensure RepCLI authentication. For example: `msiexec.exe /q /i C:\temp \installer_vista_win7_win8-64-3.5.0.xxxx.msi /L* log.txt COMPANY_CODE=" XYZ" CLI_USERS= sid BASE_IMAGE=1`

3 Make sure that the primary image has updated the most recent sensor permissions.

4 Shut down the primary image.

5 Create the primary image/VM template. Deployed clones register as separate devices and are assigned a new deviceID when installed. BASE_IMAGE looks for changes to `HKLM\SYSTEM \HardwareConfig\LastConfig`and automatically reregisters as a clone after a change is detected.

6 Provision clones from the primary image. Do not enable the sensor setting to deregister inactive VDI machines (**Enforce > Policies > Per Policy > Sensor Tab > Auto-deregister VDI sensors that have been inactive for**).

7 Confirm that newly provisioned clones have registered and are checking in.

Persistent VDI and Non-Persistent VDI Mixture

If your environment uses both persistent and non-persistent VDIs, use the following installation method to deploy to a persistent VDI image.

For all non-persistent VDIs, follow the guidance in Non-Persistent VDI .

Procedure

1 Stage the MSI for installation post provisioning: save the MSI installer on the primary image. (Do not run the installer on the primary image.)

2 After the persistent VM is provisioned, use a scheduled task, runonce, or other supported method to install the sensor using the following switches:

- `COMPANY_CODE= <company code>`
- `CLI_USERS= <desired user sid>`
- (optional) `GROUP_NAME= <desired policy>`

For example: `msiexec.exe /q /i c:\ <path to installer>`
`\installer_vista_win7_win8-64-3.5.0.1590.msi /L*vx log.txt`
`COMPANY_CODE="QFOAZP3AWF8LZ71#FT6" GROUP_NAME="Persistent VDI Systems"`
`CLI_USERS=S-1-5-32-544`

Horizon Linked-Clones and VMware Carbon Black Cloud 3.6 Sensor Deployment Best Practices

VMware Carbon Black supports Horizon Linked-Clones across the entire Carbon Black Cloud Platform (Endpoint Standard, Enterprise EDR, and Audit and Remediation). Linked-Clones are scheduled for extended support and end-of-life in the next few years. VMware Carbon Black recommends migrating to Horizon 7.13 Instant Clones or later for a more seamless deployment experience.

You can manage the performance impact of the Carbon Black Cloud sensor with Linked-Clones by ensuring the following:

1 The correct permission bypass (exclusions) are in place at the policy level.

2 A background scan is completed on the golden image virtual machine prior to using vCenter Server to take a snapshot of the virtual machine.

A background scan takes several hours to complete; there is a significant performance benefit from running it on the golden image. Completing the background scan enables the sensor to gather cloud reputation for hashes found on the golden image. This removes the need for the linked-clones to delay execution to pull reputation when those hashes eventually run.

To Set up the Carbon Black Cloud Sensor on a Golden Image Virtual Machine

1 Create the golden image virtual machine for the linked clone pool deployment. As per Horizon documentation, perform required Windows updates and install the required VMTools and Horizon Agent.

2 Install the sensor on the golden image using the following command: `msiexec.exe /q /i <Sensor Installer Path> /L* msi.log COMPANY_CODE="XYZABC" CLI_USERS=sid GROUP_NAME="<NAME Virtual Policy>"` < Sensor Installer Path> Replace this value with the location of the CBC Sensor MSI file; for example, `c:\tmp\installer_win-64-3.6.0.1941.msi`. CLI_USERS= This parameter on the golden image enables REPCLI usage on the clones. The value is the Security Identifier (SID) of the user account/group that will run the reregister now command on the clone. This means that the user/group SID identified here must run the batch script referenced below. GROUP_NAME: Indicates the policy name that has the necessary exclusions and configurations to be applied to the golden image virtual machine.

3 The following command registers the golden image to the Carbon Black Cloud with a new device ID. Verify that the golden image is registered in the Carbon Black Cloud console in the policy identified in step 2.

4 Complete an expedited background scan on the golden image to optimize clone performance.

   a In the Carbon Black Cloud console, click **Enforce > Policies**, select the policy, and click the **Sensor** tab.

   b Select the **Run background scan** option and select either **Standard** or **Expedited** scanning.

   c Select a background scan to run a one-time scan of online endpoints.

   d Click **Save** for the policy changes to take effect.

   e You can track progress by using the `repcli status` command, which includes scan information in the General Info section. `> "C:\Program Files\Confer\RepCLI.exe" status General Info: Sensor Version[3.3.0.984] Local Scanner Version[4.9.0.264 – ave.8.3.52.154:avpack.8.4.3.26:vdf.8.15.17.116 Details[] Kernel File Filter[Connected] Background Scan[Complete] Total Files Processed[2025] Current Directory[None]`

5 Stage the 79180_CB.bat batch file to run the re-registration command of the sensor on the golden image. Update PRIMARY in the script to the hostname of the corresponding golden image in step 3.

6 Take a snapshot of the golden image.

7 Create a linked clone pool with the golden image and snapshot in the Horizon Console.

8 Provide the path in the Post Synchronization script (that is, the batch file path. For example: `C:\CB.bat`). Note that the batch script is only relevant for Horizon before v7.13 and any pre-3.6 sensor.

9 After the pool becomes available in the Horizon Console, check that the newly created linked clones are registered with a new Device ID in the Carbon Black Cloud console. If there are multiple clones sharing a deviceID in the Carbon Black Cloud console, contact Support.

10 Use the per-policy auto de-registration option to remove inactive VDI machines. Go to **Enforce > Policies > {Instant Clone Policy} > Sensor** where {Instant Clone Policy} is the policy into which you will place your Instant Clone VMs.

   a Select **Delete sensors that have been de-registered for**

   b Set the timeframe to remove inactive VMs.

   Note Previously, the Carbon Black Cloud could automatically de-register golden image machines due to inactivity. Now, the Carbon Black Cloud backend no longer leverages time-based de-registration for any machine that has a child. To make sure that the backend recognizes the machine as a golden image, create at least one clone before powering off the golden image, or place the golden image in a policy that has time-based deregistration disabled.

   Best practices recommend using policies for time-based deregistration, instead of a global setting, and placing persistent/full clones in a policy that does not have auto-deregistration enabled. Enable the per-policy option for policies that have non-persistent clones (linked or instant). Persistent clones and physical machines should not use either of the auto-deregistration options. Any virtual machine that is a registered child and is not a parent is eligible for auto-deregistration.

Expected Limitations and Mitigation

- During the linked clone pool creation, a temporary full clone of the golden image known as the “internal template” (with a device name `itXXXXXX`) powers on and has network access, at which point the Carbon Black Cloud sensor on that internal template device will probably connect to the Carbon Black Cloud backend with the same device ID as the golden image. This connection results in the golden image being overwritten by the `itXXXXXX` device in the Carbon Black Cloud console.

  When the golden image is powered on, the sensor on the golden image re-connects to the backend and overwrites the `itXXXXXX` device. In addition to the duplicate devices overwriting each other’s data on the backend, this can lead to the backend sending a re-register request to the golden image. This causes the golden image to be considered a VDI by the backend,

which could cause the golden image to de-register due to inactivity. The duplicate device scenario can also expose a group membership bug where the golden image is no longer in the expected policy group. The negative implications of the internal template having the duplicate device ID as the golden image are as follows:

- The internal template's events and activities can intermingle with the golden images.
- The golden image's device name in the console might change.
- If using MSM to assign device policy by device name, make sure that golden image and internal template names are accounted for.

- We recommend that you deploy the linked clone with the golden image powered off. This recommendation will not eliminate the internal template duplicate device ID scenario, but it will mitigate the downside of having a duplicate device ID.

Installing Windows Sensors on Endpoints

5

This section describes how to install Windows sensors on the command line or through software distribution tools.

Important Before you begin the processes described here, read Chapter 1 Getting Started. It contains highly relevant information to help you succeed in your sensor installation.

Before you can install sensors, perform the following steps:

To Obtain a Company Registration Code

To Download Sensor Kits

If you are installing Windows sensors v3.5 or later, you can install sensors offline. This is useful for organizations who want to create a primary image and clone it to offline computers. This option is only available if you are installing sensors on the command line, or by using software distribution tools. See also Windows Sensor Supported Commands.

Note With the release of the Windows 3.6 sensor, you can supply either the installation code (obtained via email — see To Invite Users to install Sensors) or the company code (obtained via the console — see To Obtain a Company Registration Code).

Important The 3.6 Windows sensor leverages a content management system to enable dynamic configuration of prevention features. Prior to installing or upgrading to 3.6, if you have restrictive firewall policies active in your environment, you might need to add a new firewall/proxy exclusion for the sensor to be fully functional. See Configure a Firewall.

For additional assistance, see Carbon Black Knowledge Base Articles.

This chapter includes the following topics:

- Install Windows Sensors on Endpoints by using Group Policy
- Install Windows Sensors on Endpoints by using SCCM

Windows Sensor Rollback

With the Carbon Black Cloud Windows 3.6 sensor and later, if a failure occurs during an initial install or uninstall, the endpoint will be returned to the state it was in prior to the attempt.

If a failure occurs during initial installation of the sensor, the sensor will rollback any changes made to the system. This includes files, services, and registry artifacts that were removed, thereby leaving the system in a clean state to reattempt installation. This rollback does not harm other services or files on the endpoint.

If a failure occurs during the uninstall of a sensor, the sensor will roll back any changes including files, services, and registry artifacts. This rollback does not harm other services or files on the endpoint. The endpoint continues to check into the console and is controlled via its designated policy.

Local Scan Settings and the AV Signature Pack

The AV Signature Pack is not packaged with the sensor installation, but should be downloaded and installed automatically after sensor installation based on policy settings. As a best practice, we recommend that you download and install the AV Signature Pack 10 seconds or more after sensor installation.

Note The local scan feature is only available for Windows sensors 2.0 and later.

The AV Signature Pack requires approximately 120MB at rest. During run time, 400MB is required because a second copy is created; the scan continues to function while signatures are being updated. After the update is complete, the old signatures are deleted. At least 200MB of memory is required to run the local scan.

Signature file updates are ON by default via a policy setting. You might encounter high bandwidth utilization upon sensor installation due to the initial signature file download. Subsequent updates following the initial install of the AV Signature Pack are differential. Therefore, setting a regular update schedule ensures that every subsequent update remains small.

For more information about local scan settings, see the *VMware Carbon Black Cloud User Guide*.

To avoid network saturation during sensor installation, we recommend the following best practices:

- Install sensors in small batches.
- Set up a local mirror server for signature updates and configure your policy so that sensors download updates from the local server. See Chapter 10 Signature Mirror Instructions.

- Disable automatic signature updates. Deploy the initial signature pack by using the standalone installer, and then re-enable automatic signature updates.

To Disable Automatic Signature Updates and use the Standalone Installer

Use the following procedure to disable automatic signature updates.

Procedure

1. Sign in to the Carbon Black console.
2. Click **Enforce**, click **Policies**, and select the policy.
3. Click the **Local Scan** tab and disable **Signature Updates**. Click **Save**.
4. Install the sensors. Make sure the sensors are assigned a policy that has signature updates disabled (steps 1 and 2). Wait at least 10 seconds before you run the signature pack installer.
5. Click **Endpoints**, click **Sensor Options**, and click **Download sensor kits**.
6. Download the AV Signature Pack.
7. Run the following command under a user account that has full administrator rights by using system management software:

   `CbDefenseSig–`*`YYYYMMDD`*`.exe /silent`

   Note You can run the installation command through Live Response.

8. On the **Local Scan** tab on the Policies page, enable **Signature Updates**. After you save the changes to the selected policy, sensors in that policy begin to download the AV Signature Pack from Carbon Black servers in the the next 5-60 minutes.

   By default, updates download every 4 hours with a staggered update window of 4 hours. You can change these settings on the **Local Scan** tab of the Policies page.

To Update the AV Signature Pack by using the RepCLI Command

You can update the AV signature pack by using the RepCLI command.

Procedure

1. Log into the machine with a user account that matches the AD User or Group SID that was configured at the time of sensor install.
2. Open a command prompt window with administrative privileges.
3. Change the directory to `C:\Program Files\Confer`
4. Type the following command: `repcli UpdateAvSignature`

Results

If the command is successful, the message **The request of AV signature update has been accepted** displays on the command line.

Note Active Directory-based SID authentication is not required to run the `repcli UpdateAVSignature` command.

Windows Sensor Command Line Parameters

The following command line parameters are used during a Windows command line sensor installation.

Parameter	Required or Optional	Description
`/q`	Required	If you install without using this parameter, the user is prompted for an installation code.
`/i`	Required	This parameter tells the MSI to install.
`/L*`	Optional	Creates an MSI install log file.
`/L*vx`	Optional	Creates a verbose MSI install log file. This is recommended over the `/L*` parameter because it provides more information to troubleshoot installation problems.

Windows Sensor Supported Commands

You can use the following command line parameters during a Windows sensor install.

Using commands or command line parameters other than those listed in the following table can cause the installation to fail.

Carbon Black Cloud supports automatic detection of proxy settings; however, it does not prompt for or pass the machine's credentials for use in proxy authentication, if enabled. If proxy authentication is required for your environment, use the command line options to specify `PROXY_SERVER=value`, `PROXY_USER=value`, and `PROXY_PASSWD=value`. See Configure a Proxy.

Note

- Command options are case-sensitive.
- The `RepCLI` command is included with Windows sensors v3.3.0 and higher. See CB Defense: What is the RepCLI Utility?

Command options	Values	Notes
`AMSI=value`	0	For Windows sensors 3.6+ only. Default is enabled. If you set AMSI=0, then the AMSI feature is disabled and Carbon Black does not detect or block any AMSI activity.
`AUTO_UPDATE=value`	1/0 or true/false	Default is true (enable auto update); turning this off will prevent the update from being pushed from the backend.
`BACKGROUND_SCAN=value`	1/0 or true/false	Default is true. Not applicable to Audit and Remediation Standalone.
`BASE_IMAGE=value`	1/0 or true/false	Default is false; the installed image is a base image that can be cloned to child images. This option is not supported for VDI.
`BYPASS=value`	1/0 or true/false	Default is false; setting it to true will enable bypass mode In bypass mode, the sensor does not send any data to the cloud: the sensor functions in a passive manner and does not interfere with or monitor the applications on the endpoint. Installing the sensor in bypass mode enables thorough testing for interoperability issues.
`CBLR_KILL=value`	1/0	A value of 1 disables Live Response functionality for the sensor. The default value is 0.
`CLI_USERS=sid`	SID value for authenticated users group	Use this field to enable the `RepCLI` tool. Any member in the specified user group can use the authenticated `RepCLI` commands.
`COMPANY_CODE=value`	Company registration code	Required for command line installations.
`CONNECT_LIMIT=value`	Number of connections per hour	Optional; default is no limit.
`DELAY_SIG_DOWNLOAD=value`	1/0	Default is delay signature/definition download. We recommend that you do not change the default value.
`FILE_UPLOAD_LIMIT=value`	4-byte integer representing number of megabytes	Example: value of 3 is a limit of 3*1024*1024 bytes; default value is 5.
`GROUP_NAME=value`	String value	Optional policy name assignment. Enclose this value with double quotes if the policy name includes spaces.
`HIDE_COMMAND_LINES`	1/0	Obfuscates command line inputs. Default is 0.
`LAST_ATTEMPT_PROXY_SERVER`	Value example: 10.101.100.99:8080	Optional. Sensor will attempt cloud access by using this setting when all other methods fail (including dynamic proxy detection).

Command options	Values	Notes
`LEARNING_MODE=value`	Value is the number of hours after sensor install to limit event types. This is a mechanism for reducing the load on the backend by dropping some report types after initial install. Generally, more reports are sent to the backend soon after sensor install because the sensor reports on newly detected hashes. Learning mode reports only on file and process behavior while the sensor is detecting hashes. Reporting of API, registry, and network behavior is dropped during this period.	Optional; default is disabled.
`OFFLINE_INSTALL=value`	1/0 or true/false	Optional. Default is false. This parameter allows you to install sensors when the endpoint is offline. The sensor connects with the Carbon Black Cloud backend and accesses a policy when network connectivity is restored. The sensor is in a bypass state until the sensor can access the policy. This option is only available for Windows sensors v3.5 and later.
`PROXY_PASSWD=value`	Proxy password	Optional.
`PROXY_SERVER=value`	server:port	Optional.
`PROXY_USER=value`	Proxy username	Optional.
`QUEUE_SIZE=value`	Event backlog in MB	Optional; default is 100MB; this value does not include SSL overhead.
`RATE_LIMIT=value`	KB per hour	Optional; default is No Limit.
`VDI=value`	1/0 or true/false	This option is deprecated in sensor versions 3.4+. Default is false.
`USER_EMAIL=value`	Email address Example: user@example.com	Optional.

Obfuscation of Command Line Inputs

Endpoint users might input sensitive data into the command line. The obfuscation of command line inputs protects against unauthorized users accessing the data in plain text in the sensor `.log` files and the sensor databases.

There are three ways to obfuscate command line inputs:

- **Command line -** `HIDE_COMMAND_LINES=1`
- **RepCLI -** `hideCmdLines [0|1]`
- **Set** `HideCommandLines=true` **in** `cfg.ini`

The setting enables the obfuscation of command line input in sensor `.log` files and databases. The data in the VMware Carbon Black Cloud console is not obfuscated.

Windows Command Line Install on Endpoints — Examples

Review the following examples of Windows sensor installations.

The following commands should be on a single line. For documentation formatting reasons, they may appear here on several lines.

Base install using a company registration code

```
msiexec /q /i C:\Users\UserFolderName\Desktop\installer_vista_win7_win8-32-3.3.0.953.msi /L*
log.txt COMPANY_CODE= XYZ
```

In this basic install example, no policy is specified; therefore, the sensors are assigned to either the Standard policy, or to a policy that a sensor group specifies (if sensor groups are defined and the sensors match the sensor group criteria).

For more information about sensor groups, see the *VMware Carbon Black Cloud User Guide*.

Base install into a specific policy

```
msiexec /q /i C:\Users\UserFolderName\Desktop\installer_vista_win7_win8-32-3.3.0.953.msi /L*
log.txt COMPANY_CODE=XYZ GROUP_NAME=Phase1
```

Using the GROUP_NAME (policy assignment option) assigns the sensor to the specified policy. To use sensor groups to determine a policy assignment, omit this option.

Install using RepCLI

```
msiexec /q /i C:\temp\installer_vista_win7_win8-32-2.0.4.9.msi /L* log.txt COMPANY_CODE=XYZ
CLI_USERS=S-1-2-34-567
```

Note To enable RepCLI authentication for installed sensors, see CB Defense: How to Enable RepCLI Authentication on Existing Sensors.

Windows Sensor Log Files and Installed Services

The following log files and installed services reside on endpoints that have a Windows sensor installed.

Windows log files

Use the `/L* log.txt` command line option to obtain an MSI log that shows the Windows installation process. A `confer-temp.log` file is also generated in `C:\Users\Username\AppData\Local\Temp` that shows the sensor registration attempts to the cloud. These two log files are required for troubleshooting installation and upgrade issues.

The Windows 3.6 sensor stores some log files in `Program Files` and some log files in `ProgramData`. Previous versions of the sensor stored logs in in the `\Program Files\ Confer\Logs\` directory. Carbon Black will continue to move all log files to `ProgramData` to align with Microsoft guidelines. You must have administrative privileges to access the log files in `ProgramData`.

- `\Program Files\Confer\Logs\`
- `\ProgramData\CarbonBlack\Logs\`

Windows installed services

- Main sensor service: `RepMgr64.exe, RepMgr32.exe, Scanhost.exe` (if local scanning is enabled)
- Utility: `RepUtils32.exe, RepWmiUtils32.exe`
- UI: `RepUx.exe`

Install Windows Sensors on Endpoints by using Group Policy

To install sensors by using Group Policy, make a batch file to pass the parameters to an edited . `msi` file.

By default, Group Policy installs software on startup; therefore, you must reboot the endpoint to install the sensor.

To Create a Microsoft Installer Transform (.MST) File

To create an .MST file, perform the following procedure.

Procedure

1. Sign in to the Carbon Black Cloud console.
2. On the navigation bar, click **Inventory** and then click **Endpoints**.
3. Click **Sensor Options** and then click **Download sensor kits**. Download the . `msi` file for the Windows sensor .
4. Download and install the Orca installer; see https://msdn.microsoft.com/en-%20us/library/windows/desktop/aa370557(v=vs.85).aspx.
5. Right-click the `.msi` file that you downloaded in Step 3 and click **Edit with Orca**.
6. Click **Transform > New Transform**.
7. Create additional **Property** table entries. Under **Tables > Property**, right-click in a blank space and then click **Add row**.
8. Click **Property** and enter "COMPANY_CODE". Click **Value** and enter the company registration code for your organization. (See To Obtain a Company Registration Code.)

9 If you are installing in a VDI Environment, see Chapter 4 Installing Sensors on Endpoints in a VDI Environment for additional parameters.

10 Click **Transform > Generate Transform** and save the file as an . `mst` file. Use a file name that is easily recognizable.

Note Carbon Black recommends that you create a verbose . `msi` install log file to help troubleshoot Group Policy installation or update issues.

To Automatically Create a Windows Installer .MSI Log

To create an .`MSI` log, perform the following procedure.

1 Open the Group Policy editor and expand **Computer Configuration > Administrative Templates > Windows Components**.

2 Select **Windows Installer** and double-click **Logging** or **Specify the types of events Windows Installer records in its transaction log**, depending on the Windows version.

3 Select **Enabled**.

4 In the **Logging** textbox, type **`voicewarmupx`**.

5 Click **Save Changes**.

This setting creates an . `msi` install log for all users in the GPO in the `c:\Windows\Temp\` folder on the system volume.

To enable Windows Installer .msi log using the registry

1 Open `Regedit`.

2 Go to registry key `HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\Installer`.

3 Set the **Logging** registry value to `voicewarmupx`.

Note If Group Policy is configured to automatically create a Windows Installer . `msi` log, the registry value `voicewarmupx` should match the value that is configured in Group Policy.

To Install Sensors by using Group Policy

You can install sensors via Group Policy by using the `.msi` file that you previously created.

Procedure

1 Click **Start > Administrative Tools > Group Policy Management**.

2 Click **Software Settings > Software Installation > New > Package**.

3 Select the `.msi` file that you downloaded in the To Create a Microsoft Installer Transform (.MST) File procedure.

4 Under **Deployment Method**, click **Advanced**.

5 Add a package name that identifies the sensor (for example, WinSensor34). For 32 bit `.msi` files only: in the **Deployment** tab, click **Advanced** and uncheck **Make this 32-bit x86 application available to Win64 machines**. Click **OK**.

6 Click the **Modifications** tab and click **Add**. Select the `.mst` file and click **Save**.

7 If you use a script to force a reboot to install software, run the script.

8 Check the Carbon Black Cloud console periodically to verify that sensor information is populating and that the sensors are checking in regularly.

Note

- The path to the `.msi` and `.mst` files must be available through a network share that is accessible from everywhere in your network, and to which everyone has at least read permissions.
- For additional optional installation properties, see Windows Sensor Supported Commands.
- Active Directory does not support command line parameters. You must make a batch file to pass the parameters or package to an edited `.msi` file. Upon the next system restart, a drive is mounted and the installation is scheduled. The installation failure rate when using Active Directory is usually higher than with other software management tools.
- By default, Group Policy installs software on startup, You can force an install/reboot by using a script. Consider the restart requirement when you deploy sensors via Group Policy.
- For additional content and related KB articles, see How to Get Started with GPO Deployment on the Carbon Black User Exchange.

Install Windows Sensors on Endpoints by using SCCM

You can install Windows sensors by using System Center Configuration Manager (SCCM).

To Add the Sensor Application to SCCM

As a first step in installing a Windows sensor by using SCCM, perform the following procedure.

Procedure

1 Open SCCM Configuration Manager.

2 In the **Software Library**, click **Overview > Application Management > Applications**.

3 Right-click **Applications** and click **Create Application**.

4 On the **General** page, select **Automatically detect information about this application from installation files**:

- **Type**: Windows Installer (*. `msi` file)

- **Location**: Accessible share that contains the sensor `.msi` file

5 Click **Next**. On the **Import Information** page, a message displays: **Application information successfully imported from the Windows Installer**. Click **Next**.

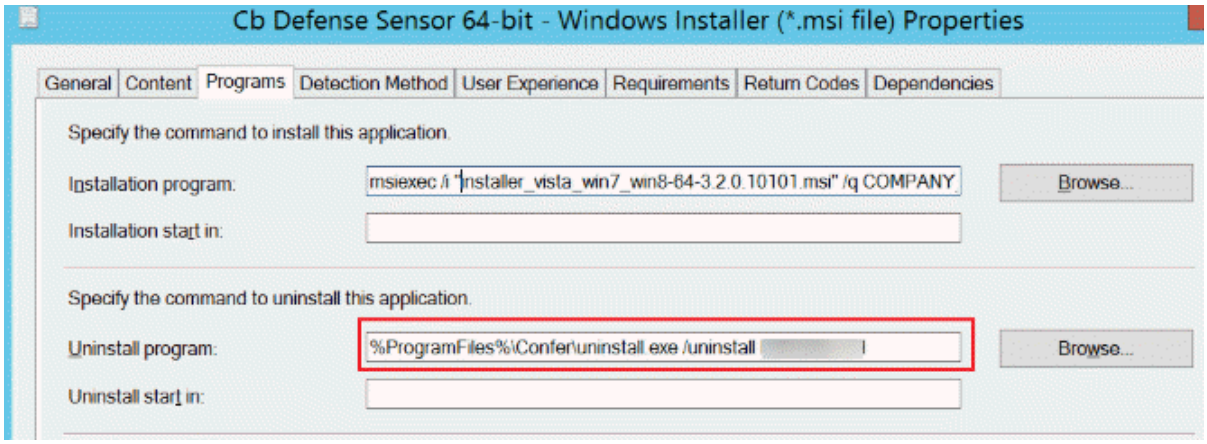
6 On the **General Information** page, add the required COMPANY_CODE install parameter and any other optional install parameters. See Windows Sensor Supported Commands for options. Click **Next**and on the **Summary** page, click **Next**.

7 On the **Completion** page, view the application details and click **Close**.

8 In the **Software Library**, right-click **Cb Defense Sensor Application** and click **Properties**.

9 Click the **Deployment Type** tab. Click the deployment type for CB Defense and click **Edit**. Note that the CB Defense type applies to Endpoint Standard, Enterprise EDR, and Audit and Remediation.

10 Click the **Programs** tab. If the **Require code to uninstall sensor** is enabled for the sensor policy and you want to be able to uninstall the sensor using SCCM, change the uninstall command from `msiexec /x "installer_vista_win7_win8-xx-x.x.x.xxxx.msi"` to `%ProgramFiles%\Confer\uninstall.exe /uninstall <Company Deregistration Code>`.

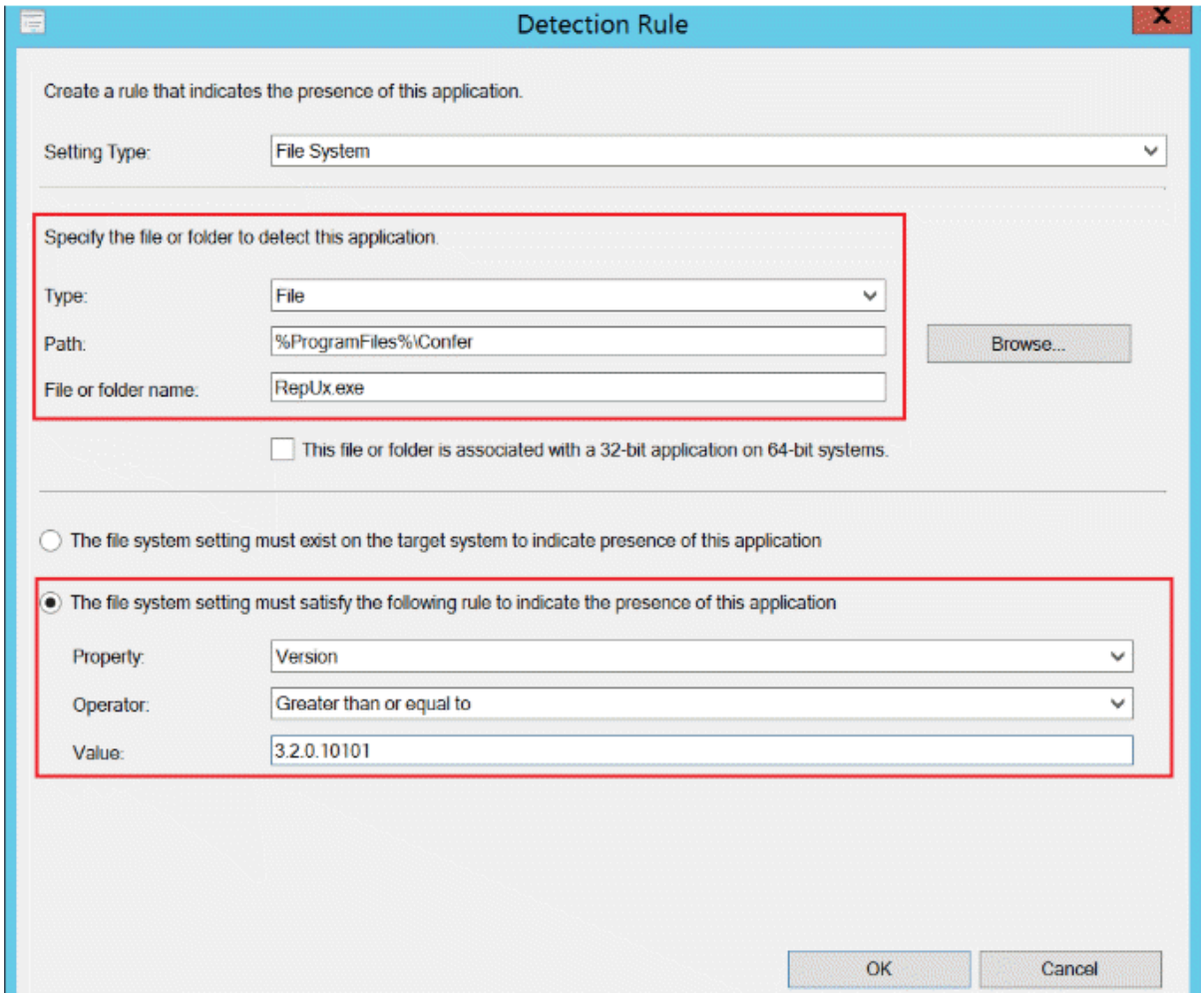


11 Click the **Detection Method** tab. Select the configured detection rule and click **Edit Clause**. Change the **Setting Type** to **File System**

12 Select **The file system setting must satisfy the following rule to indicate the presence of this application**.

13 Set **Path** to `%ProgramFiles%\Confer` and **File or Folder name** to `RepUx.exe`

14 Configure **MSI Property Version**, **Operator Greater than or equal to**. The **Version** is the currently installed sensor version.



15 Click **OK** three times to save **Detection Rule**, **Detection Method**, and **Deployment Type**.

To Deploy the Sensor Application using SCCM

To install a Windows sensor by using SCCM, follow this procedure.

Procedure

1 Open SCCM Configuration Manager.

2 In the **Software Library**, click **Overview > Application Management > Applications**.

3 Select the CB Defense application, and click **Deploy**.

4 On the **General** page, for the `Collection` field, click **Browse**. From the dropdown menu, select **Device Collections** and select a collection of devices. Click **Next.**

5 On the **Content** page, click **Add** to add a distribution point. Click **Next**.

6 On the **Deployment Settings** page, set **Action** to **Install**, set **Purpose** to **Required**, and click **Next**.

7 On the **User Experience** page, set your deployment preferences and click **Next.**

8 On the **Alerts** page, set your alert preferences and click **Next.**

9 On the **Summary** page, review and confirm all settings and click **Next**.

10 On the **Completion** page, click **Close**.

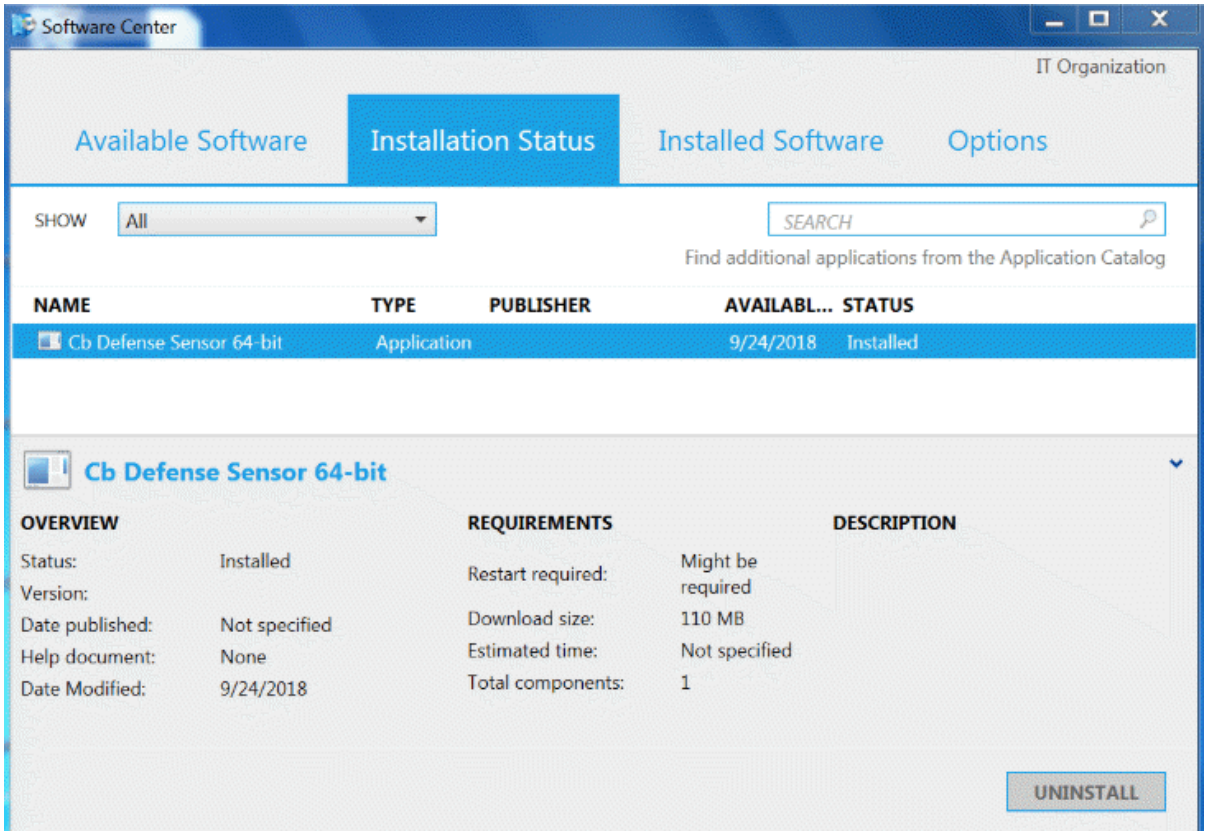
To Verify that the Sensor Application was Deployed via SCCM

After installing the Windows sensor by using SCCM, perform the following procedure to verify the deployment.

Procedure

1 In SCCM Configuration Manager, select the **CB Defense Sensor Application**, click the **Deployments** tab, and check **Compliance** status.

2 On the target device, open the Software Center and view the **Installation Status** or **Installed Software** tab.



Updating Sensors on Endpoints

6

It is important that you keep your sensor versions up-to-date. There are three ways to update sensors.

- You can update sensors on selected endpoints through the console. You can select up to 10,000 sensors to update at one time. After you initiate sensor updates, the selected sensors receive the message to update the next time that they check in with the Carbon Black Cloud backend. The system allows up to 200 concurrent updates. When an individual sensor completes its update process, a new sensor is signaled to start its update.
- You can update a sensor by double-clicking the new installer package, by issuing a command on the command line, or by pushing the command line script through a tool like SCCM. Standard command line options are applicable. Note that the command line options from the first install persist across upgrades.
- You can reinstall the sensors.

This chapter includes the following topics:

Update Sensors on Endpoints through the Carbon Black Cloud Console

You can easily update sensors through the console.

Important If you are upgrading to the Windows 3.6 sensor, see Configure a Firewall.

Procedure

1. Sign into the Carbon Black Cloud console.
2. On the navigation bar, click **Inventory** and then click **Endpoints**.
3. Search for and select the sensors to update.

4 Click **Take Action** and then click **Update Sensors**.

5 Confirm the number of sensors to update.

6 Select the sensor version from the **Version** dropdown menu.

7 Select the checkbox to acknowledge that endpoints might be rebooted and then click **Update**

Note In limited cases, updates can cause endpoints to reboot.

Results

After you have initiated the sensor updates, you can view the progress of the updates on the **Sensor Update Status** tab on the Endpoints page.

When a sensor update status displays **Completed**, a hyperlinked count becomes available in the **Updated** column. Click the hyperlinked count to open a new browser tab to the Endpoints page, where the sensors that successfully updated are shown. If any sensors did not update, a hyperlinked count displays in the **Errors** column. Click this link to open a new browser tab to the Endpoints page, and the sensors which did not update are shown.

If the **Updated** or **Errors** sensor count is greater than 500, the hyperlinks are disabled, and only the **Export** option is available under the **Actions** column. Click **Export** to generate and download a csv file that contains the count details.

If any sensors failed to update, the **Sensor Update Status** tab displays the reason for the failure.

For more information about sensor status and the Endpoints page, see the *VMware Carbon Black Cloud User Guide*.

Update Sensors on Endpoints by using Group Policy

If you deploy sensors by using Group Policy, you must remove the existing sensor from the current Group Policy before you can perform a sensor update using Group Policy, the Carbon Black Cloud console, SCCM, manual updates, etc.

To remove the existing sensor from Group Policy

1 Click **Start > Administrative Tools > Group Policy Management** and select the Group Policy Object (GPO).

2 Click **Computer Configuration > Policies > Software Settings > Software Installation**.

3 Right-click the CB Defense Sensor package and click **All Tasks > Remove...**

4 Select **Allow users to continue to use the software but prevent new installations** and click **OK**.

Note The previous procedure removes the `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows \CurrentVersion\Group Policy\AppMgmt\{Cb Defense GUID}` registry key without uninstalling the current version of the sensor. To confirm that the registry key is removed, open `Regedit` and go to `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Group Policy\AppMgmt`. Search for "CB Defense", "PSC Sensor", or "Carbon Black Cloud". If no results are found, the key is removed.

Note If you are updating from Windows sensor version 3.2.x.x, read GPO upgrade fails on sensor version 3.2.x.x.

To update sensors by using Group Policy

1 Follow the preceding procedure to remove the sensor from its existing Group Policy.

2 Force a Group Policy update on all endpoints.

3 Use the following instructions to update the sensors: To Install Sensors by using Group Policy.

Update Sensors on Endpoints that were Deployed by using SCCM

If you deployed sensors by using System Center Configuration Manager (SCCM), you can configure SCCM to allow alternate methods of updating the sensors.

Procedure

1 Open SCCM and go to the **Software Library**.

2 Click **Overview > Application Management > Applications > Carbon Black**.

3 Click the **Deployment Type** tab and select the **Deployment Type** that is configured for the sensor.

4 Click the **Detection Method** tab, click the configured detection rule, and click **Edit Clause**.

5 Change the **Setting Type** to **File System**.

6 Set **Path** to `%ProgramFiles%\Confer`.

7 Set **File or Folder name** to `RepUx.exe`.

8 Select **The file system setting must satisfy the following rule to indicate the presence of this application** .

9 Configure **MSI Property Version** operator to **Greater than or equal to**. **Version** should be the currently installed sensor version.

10 Click **OK** three times to save the configuration.

Update Linux Sensors on Endpoints through the Command Line

You can update Linux sensors through the command line.

Procedure

1. Sign into the endpoint.
2. Download the updated sensor file; see To Download Sensor Kits.
3. Unpack the agent tar ball into: * `/var/opt/carbonblack/psc/pkgs/upgrade_staging/* Note : If you have not previously upgraded the sensor, this folder does not exist and you must create it.`
4. Run the upgrade script from `/var/opt/carbonblack/psc/pkgs/upgrade_staging` location:

   RPM:

   ```
   $rpm –U cb–psc–sensor–xxx.rpm
   ```

   Note For the RHEL sensors kit, you must specify the rpm package that corresponds to the distro version that you are installing.

   el6 --> centos/rhel/oracle 6.0-6.x

   el7 --> centos/rhel/oracle 7.0-7.x

   el8 --> centos/rhel/oracle 8.0-8.x

   DEB:

   ```
   $dpkg ––force–confold –i cb–psc–sensor–xxx.deb
   ```

5. Verify the following:
   - Agent is upgraded - `/opt/carbonblack/psc/bin/cbagentd –v` to make sure that the agent matches the version you installed.
   - Kernel or BPF module is loaded
     - Kernel module: Run the following command and verify that there is a 1 in the right column of the output. This shows that the kernel module is loaded and enabled. Other versions of the kernel might display as disabled; this is acceptable.

       Command: `lsmod | grep event_collector`

       Sample output: `event_collector_2_x_yyyyyy zzzzz 1`

     - BPF module: Run the following command and verify that the grep returns a single result with the command `event_collector`. Command: `ps –e | grep event_collector`Sample output: `85150 ? 00:00:05 event_collector`
   - Check agent-blade details on the Endpoints page in the server console:
     - Updated agent details are displayed

- Agent checks in with the server at regular intervals

Uninstalling Sensors from Endpoints

7

You can uninstall sensors from the Carbon Black Cloud console or directly at the endpoint.

This chapter includes the following topics:

Uninstall Sensors from the Endpoint by using the Carbon Black Cloud Console

You can uninstall macOS and Windows sensors via the Carbon Black Cloud console.

Note You cannot uninstall Linux sensors via the Carbon Black Cloud console. You must uninstall Linux sensors by using the command line as explained in Uninstall a Linux Sensor from an Endpoint.

Procedure

1. Sign in to the Carbon Black Cloud console.
2. On the navigation bar, click **Inventory** and then click **Endpoints**.
3. Search for and select the sensors to uninstall.
4. Click **Take Action** and then click **Uninstall**.

Results

After you uninstall a sensor, it persists on the **Endpoints** page as a deregistered sensor until you delete it. See Delete Deregistered Sensors from Endpoints.

Require Codes to uninstall Sensors at an Endpoint

If you have deployed v3.1 or later sensors, you can protect the action of uninstalling the sensor at the endpoint by requiring a unique, randomly-generated code. This setting is enabled per policy, and is recommended for security purposes. The uninstall code is case-sensitive.

To require a code to uninstall a sensor at an endpoint

1 Sign in to the Carbon Black Cloud console, click **Enforce**, and then click **Policies**.

2 Select the policy.

3 On the **Sensor** tab, select the **Require code to uninstall sensor** checkbox and then click **Save**

After you have enabled this setting, a user must have an individual device uninstall code or a company deregistration code to uninstall the sensor at the endpoint. No code is required to uninstall sensors from within the Carbon Black Cloud console.

An individual device uninstall code is automatically generated when a sensor is registered with the Carbon Black Cloud.

To view a sensor uninstall code at an endpoint

1 Sign in to the Carbon Black Cloud console.

2 On the navigation bar, click **Inventory** and then click **Endpoints**.

3 Click the **>** next to the sensor to view the uninstall code.

You can also generate a company deregistration code, and use this code to uninstall any sensor in your organization.

Caution The company deregistration code can be used to uninstall all sensors in your organization. If you do not want this capability, do not generate the company deregistration code.

To generate a company deregistration code

1 Sign in to the Carbon Black Cloud console.

2 On the navigation bar, click **Inventory** and then click **Endpoints**.

3 Click **Sensor Options** and then click **Company codes**.

4 Under **Company Deregistration Code**, click **Generate New Code**.

Note Only macOS and Windows sensors can be uninstalled with a company deregistration code. Uninstall a Linux Sensor from an Endpoint by using the command line.

Uninstall a Linux Sensor from an Endpoint

You can use this procedure to uninstall a Linux Sensor from an endpoint.

Note After you run the command, the sensor remains listed in the **Registered Devices** list on the **Endpoints** page in the console until you click **Take Action > Uninstall**.

Procedure

- Run the following command from the location where the installer kit was unpacked: `For CentOS, RHEL, SUSE or Amazon Linux: $ sudo rpm –e cb–psc–sensor`For Ubuntu: `$ sudo dpkg –– purge cb–psc–sensor`

Uninstall a macOS Sensor from an Endpoint

Use this procedure to uninstall pre-3.5.1 sensors from a macOS endpoint.

By default, this mode is interactive and requires a confirmation prompt unless you specify the `–y` parameter. To view all command line parameters, run the command by specifying the `–h` parameter.

For uninstall instructions for macOS Big Sur, see Uninstall of the Sensor on Big Sur.

Procedure

1. Open **Terminal** with elevated privileges.
2. Type **`sudo /Applications/Confer.app/uninstall –y`** and click **Enter**.

   If you require a device uninstallation code or a company deregistration code, enter it as part of the command; for example:

   ```
   sudo /Applications/Confer.app/uninstall –y –c 35BQCCYX
   ```

Uninstall a Windows sensor from an Endpoint

This procedure describes how to uninstall a Windows sensor from an endpoint.

Note You can uninstall multiple sensors by using batch files or system management tools.

Procedure

1. Open a command prompt window with administrative privileges.
2. Go to the `Confer` directory.
3. Run the following command; if you require a device uninstallation code or a company deregistration code, enter it as part of the command; for example: `uninstall.exe / uninstall 35EQCCYG`

Results

The `Confer` directory and log files remain after the sensor is uninstalled. The `Confer` directory is removed after an uninstall and a reboot.

To Uninstall Windows Sensors from an Endpoint by using Group Policy

You can use Group Policy to uninstall Windows sensors by following this procedure.

Procedure

1. Click **Start > Administrative Tools > Group Policy Management** and go to **Software Installation**.
2. In the **Results** pane, right-click the CB Defense Sensor application, click **All Tasks**, and then click **Remove**.
3. In the **Remove Software** dialog box, select **Immediately uninstall the software from users and computers** and click **OK**.

Results

The application is removed the next time a user logs on or restarts the computer.

Caution The sensor does not support uninstall using Group Policy if "Require code to uninstall sensor" is enabled. See https://community.carbonblack.com/t5/Knowledge-Base/PSC-Sensor-uninstalled-without-de-registration-code/ta-p/84736.

You can also uninstall a Windows sensor by using Group Policy **Software Installation > Results > Deployment**; however Carbon Black does not recommend this option.

To Enable SCCM to Uninstall a Windows Sensor from an Endpoint

You can enable SCCM to uninstall a Windows sensor.

On the **Programs** tab in SCCM, if the **Require code to uninstall sensor** is enabled for the sensor policy and you want to uninstall the sensor using SCCM, change the uninstall command from `msiexec /x"installer_vista_win7_win8-xx-x.x.x.xxxx.msi"` to `%ProgramFiles\Confer\uninstall.exe /uninstall <Company Deregistration Code>`.

Delete Deregistered Sensors from Endpoints

To delete deregistered sensors, use one of the following procedures.

To manually delete deregistered sensors on an endpoint

1. Sign in to the Carbon Black Cloud console.
2. On the navigation bar, click **Inventory** and then click **Endpoints**.
3. Filter the list of sensors to show only devices that have deregistered sensors.

4 Select the sensors to delete.

5 Click **Take Action** and then click **Delete deregistered devices**. You are prompted to confirm the deletion.

To automatically delete deregistered sensors on an endpoint

1 Sign in to the Carbon Black Cloud console.

2 On the navigation bar, click **Inventory** and then click **Endpoints**.

3 Click **Sensor Options** and then click **Sensor settings**.

4 Select **Delete sensors that have been deregistered for** and set the time frame. Click **Save**.

Managing Sensors for VM Workloads

8

You can secure VMware workloads in your data center using VMware Carbon Black Cloud. VMware workloads require Windows 3.6+ and Linux 2.9+ sensor versions.

Installation eligibility

The **Eligibility** column indicates the sensors that are eligible for installation.

- **Eligible**: Eligible virtual machine (VM) workloads have the appropriate version of the VMware Tools with the Carbon Black launcher. You can install sensors on the eligible VMs.
- **Not eligible**: The required version of the VMware Tools or Carbon Black launcher is unavailable. To minimize your deployment efforts, a lightweight Carbon Black launcher is available with the VMware Tools. Carbon Black launcher must be available on the Windows and Linux virtual machines (VMs).
  - For Windows VMs, the Carbon Black launcher is packaged with the VMware Tools. To receive the launcher for your workloads, you must install or upgrade VMware Tools to version 11.2 or later.
  - For Linux VMs, you must manually install the launcher available at VMware Tools Operating System Specific Packages (OSPs). To learn more, visit Carbon Black Cloud Workload Guide.

    After the launcher is available, you can proceed to install sensors on your workloads inventory.
  - If VMs are offline, you cannot proceed with the installation. Go to the vCenter Server and power on the VMs.
- **Not supported**: Carbon Black Cloud Workload does not support the Operating System (OS) or the OS version. Upgrade to the supported OS or version as per the system requirements.

This chapter includes the following topics:

- Uninstall Windows Sensors from Workloads
- Delete Deregistered Sensors from Workloads

Install Sensors for VM Workloads

Use this procedure to install sensors for workloads in the Carbon Black Cloud console.

For firewall information, see Configure a Firewall.

1 Set up your Carbon Black Cloud Workload appliance. Workload appliances must be online and connected to the Carbon Black Cloud via an API key to receive a sensor. You can confirm connectivity in two ways:

   a In the Carbon Black Cloud console, check for displayed workloads on the **Inventory > VM Workloads > Not Enabled** tab.

   b In the Carbon Black Cloud console, go to the **Settings > API Access > API Keys** page. Go to the appliance API and click the appliance name to view connection status.

2 A lightweight Carbon Black launcher is required to install a sensor for Workloads.

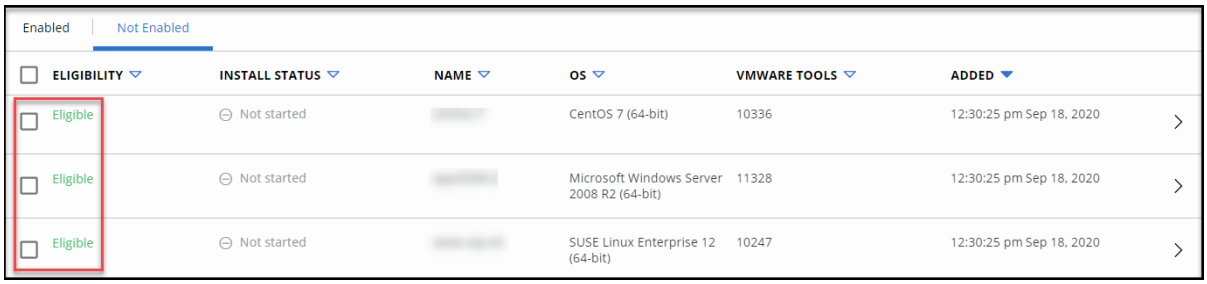
   - For Windows VMs, the Carbon Black launcher is packaged with VMware Tools. You must install or upgrade VMware Tools to version 11.2.0 or later to obtain the launcher.
   - For Linux VMs, you must manually install the launcher from VMware Tools Operating System Specific Packages (OSPs). Download and install Carbon Black launcher for your guest operating system from the package repository at http://packages.vmware.com/. For detailed instructions, see Carbon Black Launcher for Linux VMs.

To install sensors for workloads

1 Sign in to the Carbon Black console.

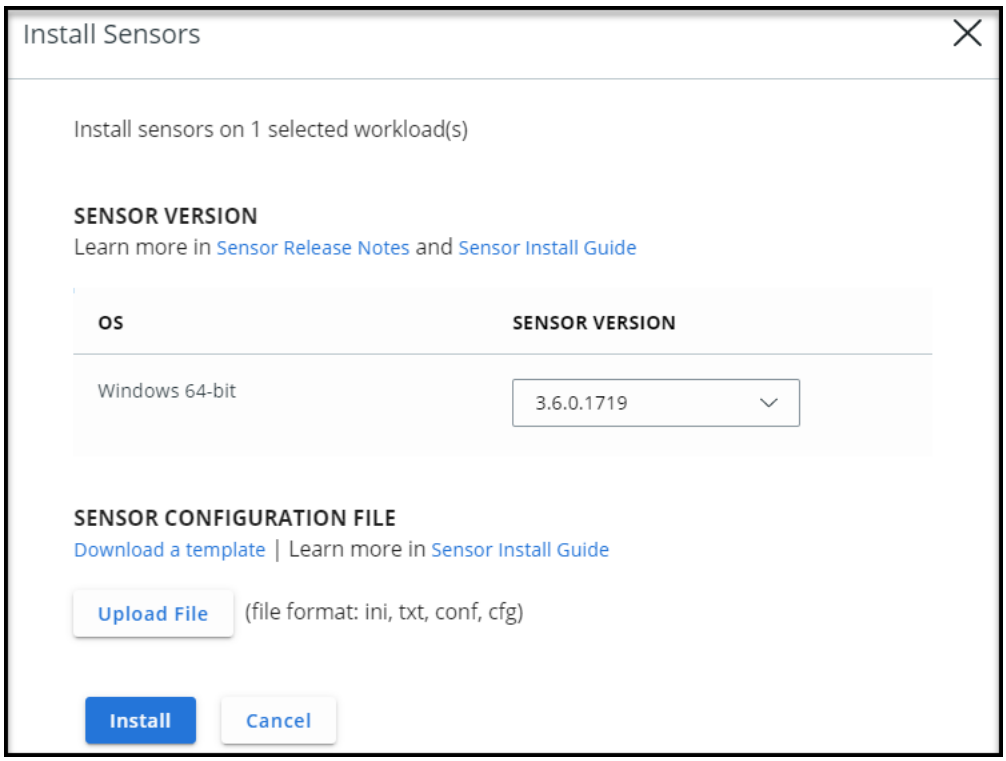
2 On the navigation bar, click **Inventory** and then click **VM Workloads**.

3 Click the **Not Enabled** tab and select eligible workloads. Eligible workloads are running a supported OS and have a correct version of the VMware Tools with the Carbon Black launcher.



4 Click the **Take Action** menu and click **Install sensors**.

5 Select the sensor version to install.



6 You can optionally upload a sensor configuration file that contains command line installation options such as proxy configuration information. See Windows Sensor Supported Commands. Download a template to see an example configuration file. (The company registration code and Carbon Black Cloud URL are pre-populated in the template.)

7 Click **Install**. The status will change to **In Progress**. After the sensor is installed, the workload no longer displays on this tab; instead, it displays on the **Enabled** tab. Click the **Enabled** tab for details about the workload and sensor.

Update Sensors for Workloads from the Console

Use this procedure to update sensors for Workloads from the Carbon Black Cloud Console.

It is important that you keep your sensor versions up-to-date.

There are two ways to update sensors:

- You can update sensors on selected workloads through the console. You can select up to 10,000 sensors to update at one time. After you initiate sensor updates, the selected sensors receive the message to update the next time that they check in with the Carbon Black Cloud backend. The system allows up to 200 concurrent updates. When an individual sensor completes its update process, a new sensor is signaled to start its update.
- You can reinstall the sensors.

Procedure

1 Sign into the Carbon Black Cloud console.

2 On the navigation bar, click **Inventory** and then click **Workloads**.

3 Search for and select the sensors to update.

4 Click **Take Action** and then click **Update Sensors**.

5 Confirm the number of sensors to update.

6 Select the sensor version from the **Version** dropdown menu.

7 Click **Update**

Update Linux Sensors on Workloads through the Command Line

You can update Linux sensors through the command line.

Procedure

1 Sign into the workload.

2 Unpack the agent tar ball into: * `/var/opt/carbonblack/psc/pkgs/upgrade_staging/*` Note : If you have not previously upgraded the sensor, this folder does not exist and you must create it.

3 Run the upgrade script from `/var/opt/carbonblack/psc/pkgs/upgrade_staging` location:

RPM:

```
$rpm –U cb–psc–sensor–xxx.rpm
```

Note For the RHEL sensors kit, you must specify the rpm package that corresponds to the distro version that you are installing.

el6 --> centos/rhel/oracle 6.0-6.x

el7 --> centos/rhel/oracle 7.0-7.x

el8 --> centos/rhel/oracle 8.0-8.x

DEB:

```
$dpkg ––force–confold –i cb–psc–sensor–xxx.deb
```

4 Verify the following:

- Agent is upgraded - `/opt/carbonblack/psc/bin/cbagentd –v` to make sure that the agent matches the version you installed.
- Kernel or BPF module is loaded
  - Kernel module: Run the following command and verify that there is a 1 in the right column of the output. This shows that the kernel module is loaded and enabled. Other versions of the kernel might display as disabled; this is acceptable.

Command: `lsmod | grep event_collector`

Sample output: `event_collector_2_x_yyyyyy zzzzz 1`

- BPF module: Run the following command and verify that the grep returns a single result with the command `event_collector`.

  Command: `ps –e | grep event_collector`

  Sample output: `85150 ? 00:00:05 event_collector`

- Check agent-blade details on the Workloads page in the server console:
  - Updated agent details are displayed
  - Agent checks in with the server at regular intervals

Uninstall Linux Sensors from Workloads

Use this procedure to use the command line to uninstall Linux sensors from Workloads.

Note After you run the following command, the sensor remains listed in the **Registered Devices** list on the **Workloads** page in the console until you click **Take Action > Uninstall**.

Procedure

- Run the following command from the location where the installer kit was unpacked: `For CentOS, RHEL, SUSE or Amazon Linux: $ sudo rpm –e cb–psc–sensor`For Ubuntu: `$ sudo dpkg –– purge cb–psc–sensor`

Results

After you uninstall a sensor, it persists on the **Workloads** page as a deregistered sensor until you delete it. See Delete Deregistered Sensors from Workloads

Uninstall Windows Sensors from Workloads

You can uninstall Windows sensors via the Carbon Black Cloud console.

Note You cannot uninstall Linux sensors via the Carbon Black Cloud console. You must uninstall Linux sensors by using the command line as explained in Uninstall Linux Sensors from Workloads.

Procedure

1. Sign in to the Carbon Black Cloud console.
2. On the navigation bar, click **Inventory** and then click **Workloads**.
3. On the **Enabled** tab, select the sensors to uninstall.
4. Click **Take Action** and then click **Uninstall**. You are prompted to confirm the action.

Results

After you uninstall a sensor, it persists on the **Workloads** page as a deregistered sensor until you delete it. See Delete Deregistered Sensors from Workloads

Delete Deregistered Sensors from Workloads

To delete deregistered sensors from Workloads, use one of the following procedures.

To manually delete deregistered sensors from workloads

1 Sign in to the Carbon Black Cloud console.

2 On the navigation bar, click **Inventory** and then click **Workloads**.

3 On the **Enabled** tab, filter the list of sensors to show only devices that have deregistered sensors.

4 Select the sensors to delete.

5 Click **Take Action** and then click **Delete deregistered assets**. You are prompted to confirm the deletion.

To automatically delete deregistered sensors from workloads

1 Sign in to the Carbon Black Cloud console.

2 On the navigation bar, click **Inventory** and then click **Workloads**..

3 Click **Sensor Options** and then click **Sensor settings**.

4 Select **Delete sensors that have been deregistered for** and set the time frame. Click **Save**.

Managing Kubernetes Clusters

9

You can secure your Kubernetes workloads using the Carbon Black Cloud console.

To get started, you must have a Kubernetes cluster running in your Kubernetes environment.

After completing the cluster setup process, you can view Kubernetes clusters in the Carbon Black Cloud console.

This chapter includes the following topics:

Kubernetes Cluster Setup Prerequisites

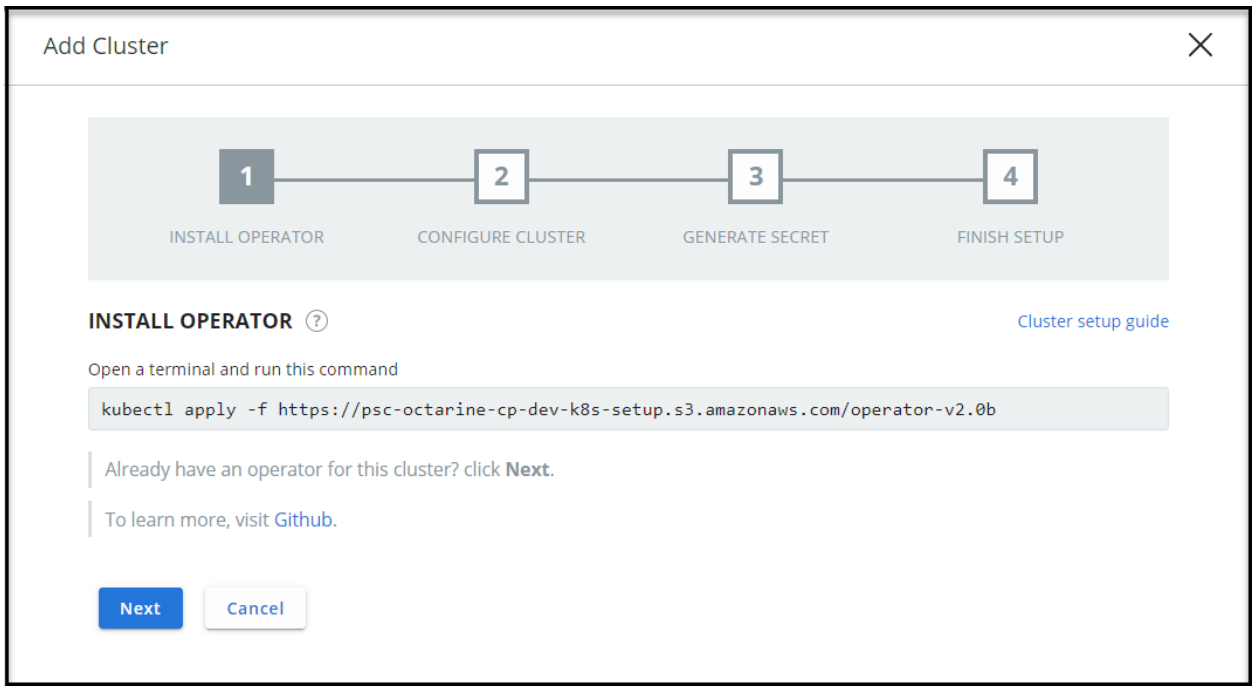
Before starting the cluster setup, make sure you complete the following prerequisites.

- Kubernetes Security DevOps or Super Admin role assigned to you on the Carbon Black Cloud console.
- Administrator privilege on your Kubernetes clusters.
- Kubernetes clusters have an admission control plugin with `ValidatingAdmissionWebhook` enabled.
- Kubernetes clusters can be controlled using the Kubernetes command-line tool `kubectl`. Visit Github to learn more.
- The Kubernetes cluster nodes can access `dashboard.confer.net` for https requests on port 443.
- The Kubernetes cluster nodes can access `events.containers.carbonblack.io` for gRPC traffic on port 50051.
- The Kubernetes cluster nodes can pull container images from the Docker hub registry.

Set up a Kubernetes Cluster

This procedure describes how to set up a Kubernetes cluster in the Carbon Black Cloud console.

In the VMware Carbon Black Cloud console, click **Inventory > Kubernetes > K8s Clusters**. Click **Add cluster** and follow the setup wizard by clicking **Next** after each step.



Procedure

1 **Install Operator**: You must install an Operator on your cluster. Operators run in a Kubernetes cluster; they deploy and manage the component's capabilities and report its health to the Carbon Black Cloud console.

Note If an Operator is already installed in your Kubernetes cluster, you can skip step 1. To determine whether the Operator is installed:

a Copy the `kubectl get pods –A –l name=octarine–operator` command.

b Open the terminal of your Kubernetes environment, and run the copied command.

If the Operator is not installed:

Copy the supplied command. Open the terminal of your Kubernetes environment, and run the copied command.

2 **Configure Cluster**: Define the cluster that you are adding to the Carbon Black Cloud console.

a **Cluster name**: Type the name of the cluster. The cluster name must be unique to your cluster group, and it cannot contain a colon (:) symbol.

b **Cluster group**: Type the name for the cluster group. Organize your clusters in the Carbon Black Cloud console with cluster groups. Cluster groups make it easier to specify resources in the scopes and apply policies.

3 **Generate Secret**: To establish communication between your Kubernetes cluster and the Carbon Black Cloud, you must provide an API key. Generate an API key or use one that already exists. We recommend having a separate API key for each cluster.

   a **Generate a new API key**: Enter an API key name that is unique to your Carbon Black Cloud organization and click **Generate API Key**. Alternatively, select an existing key. Do not reuse keys between clusters. Existing keys can only be used to re-instrument a cluster.

   b Click **Generate Secret**. Copy and paste the secret into the Kubernetes terminal or the Kubernetes secrets management tool.

4 **Finish Setup**: Copy the supplied command. Open the terminal of your Kubernetes environment, and run the copied command. Alternatively, you can copy and paste the YAML details.

5 Click **Done**.

Results

After submitting the action, the **Status** column on the Kubernetes Clusters page shows as **Pending install.** After the cluster setup is successful, the status changes to **Running**.

The **Status** column indicates the status of a cluster.

- **Running**: All components are up and running without any errors.
- **Warning**: One of the non-critical components is down or cannot detect the status.
- **Error**: One of the critical components is down or cannot detect the status.
- **Critical**: No activity is detected from any cluster components for more than 24 hours.
- **Pending install**: Cluster setup is in progress.

Note The cluster setup process can take up to three minutes to complete. The system communicates with the Kubernetes cluster to verify the status every minute. During this time, the status might display as an error. We recommend waiting three to five minutes after submitting the install request to verify the correct status.

Delete a Kubernetes Cluster

You can delete Kubernetes clusters that are no longer in use from the Carbon Black Cloud console.

Procedure

1 Click **Inventory > Kubernetes > K8 Clusters**.

2 Under the **Actions** menu, click the trashcan icon next to the cluster.

Results

Deleting a cluster removes it from Kubernetes cluster list, but does not uninstall its components.

Signature Mirror Instructions

10

This section contains Carbon Black Cloud signature mirror instructions for Linux and Windows.

Note The local scan feature is not available in the Audit and Remediation Standalone product.

See also CB Defense: Getting Started with Local Mirror Servers.

This chapter includes the following topics:

Mirror Server Hardware Requirements

VMware Carbon Black Cloud mirror servers have the following hardware requirements to service 10,000 endpoints.

- 2Ghz CPU
- 4GB RAM

The recommended schedule for pulling down updates is hourly.

Performance of a local mirror server depends on the following:

- Number of endpoints that it serves
- Network bandwidth
- Frequency of updates

You can deploy multiple mirror servers to accommodate large environments.

Signature Mirror Instructions for Linux

This procedure provides instructions on mirroring a local Linux repository of the Carbon Black Cloud local scanning signatures.

Assumptions

These instructions assume:

- A Linux operating system
- Definitions are hosted on an HTTP server at a given URL, which are entered in the **Update Servers** field of the **Local Scan** tab of a policy.

Procedure

1. Make sure that traffic to the signature update server URL is allowed without traffic inspection through any proxy/firewall (TCP/80 or TCP/443): `updates2.cdc.carbonblack.io`.
2. Download the `cbdefense_mirror_unix_x64_v3.0.zip` package from CB Defense: Local Mirror Server for Signature Updates to the server that will provide the updates.
3. Unpack the zipped file and move the contents to a directory. These files automate mirror server updates with a cron job, so they should be stored in a permanent location such as `/root/cbupdate`, `avupdate_msg.avr`, `avupdate.bin`, `HBEDV.KEY`, `update_defs.sh`, or `update_defs_ssl.sh`.
4. Open a command prompt window with administrative privileges and change the directory to the update file location.
5. Download the initial signature pack set and create the signature mirror by using the following command (`/var/www/html` is an example directory that is often used when configuring Apache): `bash ./update_defs.sh /var/www/html Note : The command can also call update_defs_ssl.sh to use https for the download.`
6. Results print to the command line. Confirm that the following directories and files are located in the root of the directory that is targeted with the update command: `ave2`, `avupdate.log`, `idx`, and `x_vdf`.
7. Update the policy:
   a. Click **Enforce**, click **Policies**, and select the policy.
   b. Click the **Local Scan** tab.
   c. Enable **Allow Signature Updates**.
   d. Add the local mirror server URL to the **Update Servers** settings for internal and offsite devices.
   e. Check the box to the right of the URL to set it as primary.
   f. Click **Save**.

Results

For a detailed example of how to configure a signature mirror server on Apache, see CB Defense: How to configure a Local Mirror (Linux).

Signature Mirror Instructions for Windows

This procedure provides instructions on mirroring a local Windows repository of the Carbon Black Cloud local scanning signatures.

Assumptions

These instructions assume:

- A 64-bit Windows operating system
- Definitions are hosted on an HTTP server at a given URL, which are entered in the **Update Servers** field of the **Local Scan** settings of a policy.

Procedure

1. Download the `cbdefense_mirror_win_x64_v3.0.zip` package from CB Defense: Local Mirror Server for Signature Updates.
2. Extract `cbdefense_mirror_win_x64_v3.0.zip files into a temp folder`:
   - `avupdate.dll`
   - `do_update.bat`
   - `do_update_ssl.bat`
   - `HBEDV.KEY`
   - `msvcr120.dl`
   - `upd.exe`
   - `upd_msg.avr`
3. Create a folder for the AV signature update files; for example, `C:\inetpub\wwwroot\CBC_SignatureUpdates`.
4. Copy the extracted files into the folder that you created in Step 3.
5. Open `do_update.bat` and set `outdir` to the folder that you created in Step 3. (To use SSL, open `do_update_ssl.bat` instead.)
6. Configure the signature mirror by running the following commands in an elevated command prompt window: `C:\>cd C:\inetpub\wwwroot\CBC_SignatureUpdates C:\inetpub\wwwroot\CBD_SignatureUpdates>do_update.bat`The following folders are created:
   - `32`
   - `64`
   - `ave2`
   - `idx`
   - `x_vdf`

7 Run Windows Task Scheduler.

a Right-click **Task Scheduler Library** and click **Create Task**.

b Click the **General** tab and define the task by adding a name and description. Select **Run whether user is logged on or not** and **Run with highest privileges**.

c Click the **Triggers** tab. Click **New** and set the trigger to run daily at your preferred start time. Repeat the task every hour indefinitely. Select **Enabled** and click **OK**.

d Click the **Actions** tab. Click **New** and **Start a program**. Set the **Program/script** to `do_update.bat`; for example, `C:\inetpub\wwwroot\CBC_SignatureUpdates>do_update.bat. Click OK .`

e Click the **Conditions** tab. Select the following settings:

- Start the task only if the computer is on AC power
- Stop if the computer switches to battery power
- Wake the computer to run this task

f Click the **Settings** tab. Select the following settings:

- Allow task to be run on demand
- Run task as soon as possible after a scheduled start is missed
- If the task fails, restart every > 1 minute
- Attempt to restart up to > 3 times
- If the running task does not end when requested, force it to stop

g Click **OK**.

8 Create an IIS web site.

a Open IIS Manager. Right-click **Sites** and click **Add Website**. Provide a site name that identifies that this web site is for the AV Signature Updates.

b Keep the **DefaultAppPool** for the **Application Pool** field.

c For the **Physical Path**, browse to the folder that was created in Step 3.

d Keep these values: **Type = http**, **IP address = All Unassigned**, and **Port = 80**.

e For the **Host name** field, type the name of the mirror server.

f Select **Start Website immediately**. Click **OK**.

g On the IIS navigation pane, under **Sites**, select the site name that you created in Step 8a.

h Double-click **Directory Browsing** and click **Enable**

i Double-click **MIME Types**. Add a new MIME type for extension of `.idx` with type of **text/plain**.

j In a command prompt window with administrative privileges, run the command `iisreset`.

k To test the URL, open a browser and type **http://**{ host name from Step 8e}. You should see the folders that were created in Step 6.

9 Update the policy.

a In the Carbon Black Cloud console, click **Enforce**, click **Policies**, and select the policy.

b Click the **Local Scan** tab and enable **Allow Signature Updates**.

c Add the local mirror server URL to the **Update Servers** settings for internal and offsitedevices. Check the box to the right of the URL to set it as the primary. Click **Save**.

Results

Note

- `do_update.bat` generates (and appends to) a log file in `%TEMP%\scanner\upd.log`. You can use this log file to troubleshoot issues.
- The **Update Servers for Onsite Devices** checkbox on the **Local Scan** tab in a policy can impact connections to the mirror server. If you have sensors that can't receive updated signatures from the mirror server, toggle the switch to resolve the issue.

Configuring Carbon Black Cloud Communications

11

Configure your network infrastructure and endpoints to ensure proper communication between sensors and the backend.

Network proxies and firewalls can interfere with communication between the Carbon Black Cloud sensor and the Carbon Black Cloud backend if they are improperly configured.

A sensor can connect to a Carbon Black Cloud backend server over TCP/443. The backend server also listens for sensors on port TCP/54443.

There is no static IP, range of IP addresses, or subnet to allow or exclude in firewall or proxy settings.

This chapter includes the following topics:

Configure a Firewall

A sensor can connect to the backend in a firewall-protected network in several ways.

- Configure a bypass on the network firewall to allow communication between the sensor and the backend over TCP/443. This is often the simplest approach.
- Configure a bypass in your network firewall to allow outgoing connections to the Carbon Black Cloud alternate port TCP/54443.
- If specific network firewall changes are not made to access the Carbon Black Cloud backend applications, the sensors try to connect through existing proxies.

Contact your authorized support representative to learn the URLs you should use.

Workloads must be able to access `prod.cwp.carbonblack.io` on port 443.

Configure the firewall to allow outgoing and incoming connections to the following service URL/ hostnames, protocols, and ports:

Backend URL	API URL	Sensor URL
https://dashboard.confer.net	https://api.confer.net	https://devices.confer.net
https://defense.conferdeploy.net	https://api5.conferdeploy.net	https://dev5.conferdeploy.net

Backend URL	API URL	Sensor URL
https://defense-prod05.conferdeploy.net	https://api-prod05.conferdeploy.net	https://dev-prod05.conferdeploy.net
https://defense-eu.conferdeploy.net	https://api-prod06.conferdeploy.net	https://dev-prod06.conferdeploy.net
https://defense-prodnrt.conferdeploy.net	https://api-prodnrt.conferdeploy.net	https://dev-prodnrt.conferdeploy.net

Note Windows sensors 3.6 and above also require access to `content.carbonblack.io`.

Signature URLs:

- http://updates2.cdc.carbonblack.io/update2 (TCP/80, default definition update server)
- https://updates2.cdc.carbonblack.io/update2 (TCP/443, default definition update server for sensor versions 3.3+)

Third-party certificate validation URLs (sensor version 3.3+: optional but recommended and on by default):

- http://ocsp.godaddy.com (TCP/80, Online Certificate Status Protocol [OCSP])
- http://crl.godaddy.com (TCP/80, Certificate Revocation List [CRL])

Configure a Proxy

The Carbon Black Cloud sensor uses a variety of mechanisms to determine whether a network proxy is present.

If a proxy is detected (or if one is specified at install time), the sensor attempts to use that proxy. If no proxy is detected, the sensor will attempt a direct connection through port 443 or 54443.

The sensor attempts to contact the Carbon Black Cloud backend by using the following methods:

- A static configured proxy that is configured during sensor installation.
- A direct connection over TCP/443.
- Auto-detection of a proxy and proxy credentials (when applicable) from the local computer's operating system settings.

If you cannot establish connectivity over the standard SSL port, the sensor can fail over to the alternate port, which is TCP/54443.

Note Carbon Black Cloud sensors automatically try to detect proxy settings during initial installation. This should be tested. If the automatic proxy detection doesn't succeed, you must define the parameters to include the Proxy IP and Port in the MSI command line during a command line installation.

If user authentication is required, the user might be prompted for credentials. This typically does not occur in environments that require proxy credentials because the sensor uses an existing configuration that avoids requiring end users to enter credentials.

To avoid going through a network proxy (and/or to avoid being blocked by a firewall), you might need to configure a bypass on your proxy server/firewall to allow outgoing connections from the sensor to the backend. Options for bypass configuration include the following:

- Configure a bypass on your firewall or proxy to allow outgoing connections to your Carbon Black Cloud domain over TCP/443.
- Configure a bypass in your firewall or proxy to allow outgoing connections to the Carbon Black Cloud alternate port TCP/54443.

Important The host domain name for the Carbon Black Cloud backend server is included in the server's certificate. Some network proxies and gateways might try to validate the certificate and deny the Carbon Black Cloud backend application connection because of a name mismatch between the certificate and real host name of the system that is running in AWS. If this occurs, you must configure the proxy or gateway so that it does not validate the backend server certificate. Note that you cannot access the certificate or hostname in the server's certificate.

Connection Mechanism Precedence

If a sensor fails to connect to the backend, it tries the last known working settings, starting with the most recent ones. These include the following:

- Proxy
- No proxy
- Credentials
- No credentials
- Proxy used at install time
- Direct connection
- Alternate 54443 port

The sensor attempts the connection in the following sequence:

1. By using a statically configured proxy server that was provided at the time that the sensor was installed.
2. A direct connection to the backend with no proxy.
3. A direct connection to the backend using the alternate port 54443 with no proxy.
4. A dynamic proxy (Internet/network settings), if present, without credentials. If other attempts fail and the proxy is identified and credentials are required, the sensor attempts this connection as a last resort.

For every proxy server connection that is attempted, the sensor tries to connect with:

1. The proxy port that is configured.
2. The alternate port 54443 if this was configured during the sensor installation.

Configure a Proxy for Linux

Use this procedure to configure a proxy through the `cfg.ini` file for all distributions.

Procedure

1. Extract the contents of the installer package into a temporary directory.
2. Use the `install.sh` script to install the agent, but do not provide a company code:

   ```
   sudo cb-psc-install/install.sh
   ```

3. Update the `cfg.ini` file with the v3.x+ company code:

   ```
   sudo /opt/carbonblack/psc/bin/cbagentd -d '<COMPANY_CODE>'
   ```

4. Append the following entry in the `/var/opt/carbonblack/psc/cfg.ini`file. You can use the IP address instead of the hostname.

   ```
   ProxyServer=<hostname>:<port number>
   ```

5. Start the agent:
   - Centos/Rhel 6:
     - `$ service cbagentd start`
   - All other distributions:
     - `$ systemctl start cbagentd`