Carbon Black.

Cb Response

6.1.2 Release Notes

October 2017

Carbon Black, Inc.

1100 Winter Street, Waltham, MA 02451 USA

Tel: 617.393.7400 Fax: 617.393.7499

E-mail: support@carbonblack.com

Web: http://www.carbonblack.com

Introduction

The *Carbon Black (Cb) Response v6.1.2 Release Notes* document provides information for users upgrading from previous versions as well as users new to the product. It consists of the following major sections:

- Preparing for Server Installation or Upgrade, on page 4 – Describes preparations you should make before beginning the installation process for Cb Response server.
- Upgrading the Cb Response Server, on page 6 – Provides information and instructions specific to server upgrades.
- New and Modified Features, on page 10 – Provides a quick reference to the new and modified features introduced with this version.
- Corrective Content, on page 36 – Describes issues resolved by this release as well as more general improvements in performance or behavior.
- Known Issues and Limitations, on page 63 – Describes known issues or anomalies in this version that you should be aware of.
- Contacting Carbon Black Support, on page 67 – Describes ways to contact Carbon Black Technical Support, and it details what information to have ready so that the technical support team can troubleshoot your problem.

This document is a supplement to the main Cb Response product documentation.

Purpose of this Release

Cb Response v6.1.2 release contains a new *Linux sensor version, bug fixes*, and *stability and performance improvements*. It packages the following component versions:

- Server: 6.1.2.171005.1008
- Windows Sensor: 6.0.3.171001.1616
- OS X Sensor: 6.0.5.170830.1306
- Linux Sensor: 5.2.12.170825.1225

Documentation

The standard user documentation for Cb Response product includes:

- *Cb Response User Guide* – Describes Cb Response feature functionality in detail, plus administrative functions.
- *Cb Response Server/Cluster Management Guide* – Describes how to install, manage, backup/restore, etc. a Cb Response server/cluster. This guide is for on-premises Cb Response installations only.
- *Cb Response Server Sizing Guide* – Provides details on infrastructure sizing for Cb Response server.
- *Cb Response API* – Documentation for the Cb Response API is located at https://developer.carbonblack.com.

Additional documentation for special tasks and situations is available on the Carbon Black User eXchange.

Preparing for Server Installation or Upgrade

This section describes requirements to meet and key information needed before beginning the installation process for the Cb Response server. All users, whether upgrading or installing a new server should review this section before proceeding. Once you have reviewed this document, see the following for specific installation instructions:

- **To install a new Cb Response server**, see "Installing the Cb Response Server" section in the *Cb Response Server/Cluster Management Guide* for version 6.1.*x*
- **To upgrade a Cb Response server**, see Upgrading the Cb Response Server later in this document.

Yum URL

Cb Response Server software packages are maintained at the Carbon Black yum repository (yum.distro.carbonblack.io). Use caution when pointing to the yum repository; different versions of the product are available on different branches as follows:

- The current 6.1.2 version is available from the Carbon Black yum repository specified in the following base URL: baseurl=https://yum.distro.carbonblack.io/enterprise/stable/x86_64/
- The current 5.3.1 version is available from the Carbon Black yum repository specified in the following base URL: baseurl=https://yum.distro.carbonblack.io/enterprise/release/x86_64/

Note: Communication with this repository is over HTTPS and requires the presence of appropriate SSL keys and certificates. During the Cb Response server install or upgrade process, other core CentOS packages may be installed to meet various dependencies. The standard mode of operation for the yum package manager in CentOS is to first retrieve a list of available mirror servers from http://mirror.centos.org:80 and then select one of those mirrors to download the actual dependency packages. If your Cb Response server is installed behind a firewall that blocks access to the outside, it is up to the local network and system administrators to ensure that the host machine is able to communicate with standard CentOS yum repositories.

System Requirements

Operating system support for the server and sensors is listed here for your convenience. The document *Cb Response – Server Sizing Guide* describes the full hardware and software platform requirements for the Cb Response server and provides the current requirements for systems running the sensor. Both are available on the Carbon Black User eXchange.

Both upgrade and new customers should be sure to meet all of the requirements specified here and in the Server Sizing Guide before proceeding.

Server / Console Operating Systems

Note: For best performance, Carbon Black recommends running the latest supported software versions.

- CentOS 6.7-6.9 (64-bit)
- Red Hat Enterprise Linux (RHEL) 6.7-6.9 (64-bit)

Installation and testing is done on default installs using the 'minimal' distribution and the distribution's official package repositories. Customized Linux installations must be individually evaluated.

Sensor Operating Systems (for Endpoints and Servers)

- **Windows**: XP SP3 – 10 / Server 2003 – 2012R2, 2016 x86 and x64

  Windows embedded OSes are individually evaluated
- **Mac**: OS X 10.7 through 10.12.4, x64 on Intel
- **Linux:** RHEL & CentOS 6.4-6.9, 7.0-7.3 x64 – standard kernel versions (2.6.32-358.el6, 2.6.32-431.el6, 2.6.32-504.el6, 2.6.32-573.el6, 2.6.32-642.el6, and 3.10.0-123.el7, 3.10.0-229.el7, 3.10.0-327.el7,3.10.0-493.el7, 3.10.0-514.el7) and the standard minor/ maintenance releases. Non RHEL/CentOS distributions or Modified RHEL/CentOS environments (those built on the RHEL platform) are not supported.

Technical Support

Cb Response server and sensor update releases are covered under the Customer Maintenance Agreement. Carbon Black recommends reviewing content on the User eXchange prior to performing the upgrade for the latest information that supplements the information contained in this document. Technical Support is available to assist with any issues that may develop during the upgrade process. Our Professional Services organization is available to assist with the upgrade process to ensure a smooth and efficient upgrade installation.

Upgrading the Cb Response Server

Supported Upgrade Paths

Server upgrades to v6.1.2 are supported from the following previous versions:

- All 5.1.*x* versions, including earlier patch releases
- All 5.2.*x* versions, including earlier patch releases
- All 6.*x* versions, including Early Access Program (EAP) and Controlled Distribution releases

For more detailed instructions for installing or upgrading the server, please refer to the *Cb Response Server/Cluster Management Guide*, which is available on the Carbon Black User eXchange. For upgrading from earlier versions, please call or email Carbon Black Technical Support.

Important: Ports and protocol requirements in version 6.*x* have changed since version 5.*x*. Refer to the "Ports and Protocols" chapter of the *Cb Response Server/Cluster Management Guide* for details.

Configure Sensor Updates Before Upgrading Server

Cb Response v6.1.2 comes with updated sensor versions. If you are upgrading your server, you should determine if you would like to upgrade to the new *sensor* versions *before* you run the server upgrade program. Servers and sensors can be upgraded independently, and sensors can be upgraded by sensor groups, rather than all at once.

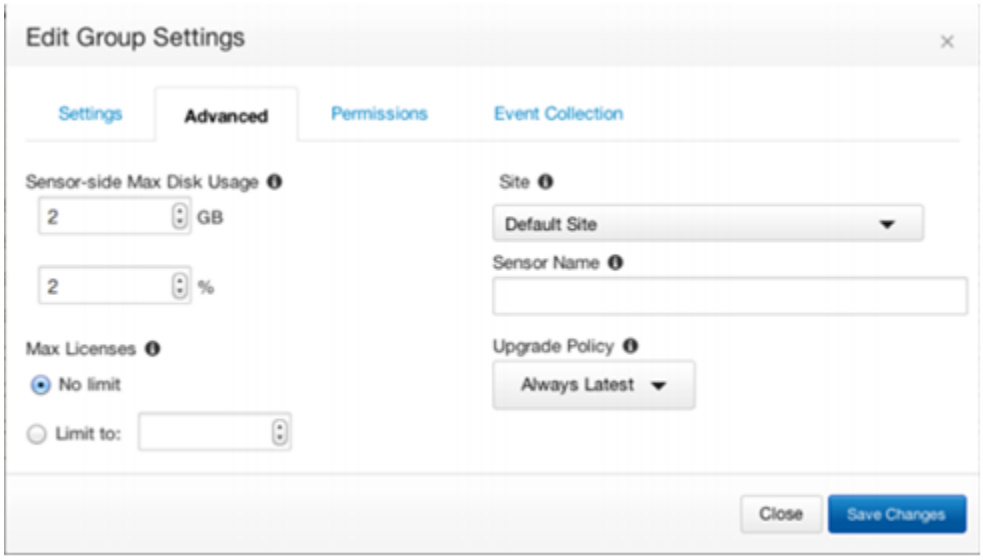
Decide if you would like the new sensor to be deployed immediately to existing sensor installations, or if you want to install only the server updates first. Carbon Black recommends a gradual upgrade of sensors to avoid any unacceptable impact on network and server performance.

Note: *There is no expected degradation to sensor performance with Cb Response v6.1.2.*

To configure deployment of new sensors via the Cb Response web UI, follow the instructions below that correspond to the version you are upgrading from.

Versions 5.1.1 and Below:

1. Log into the console, navigate to the Sensors page, and edit the group settings for each active Sensor Group:

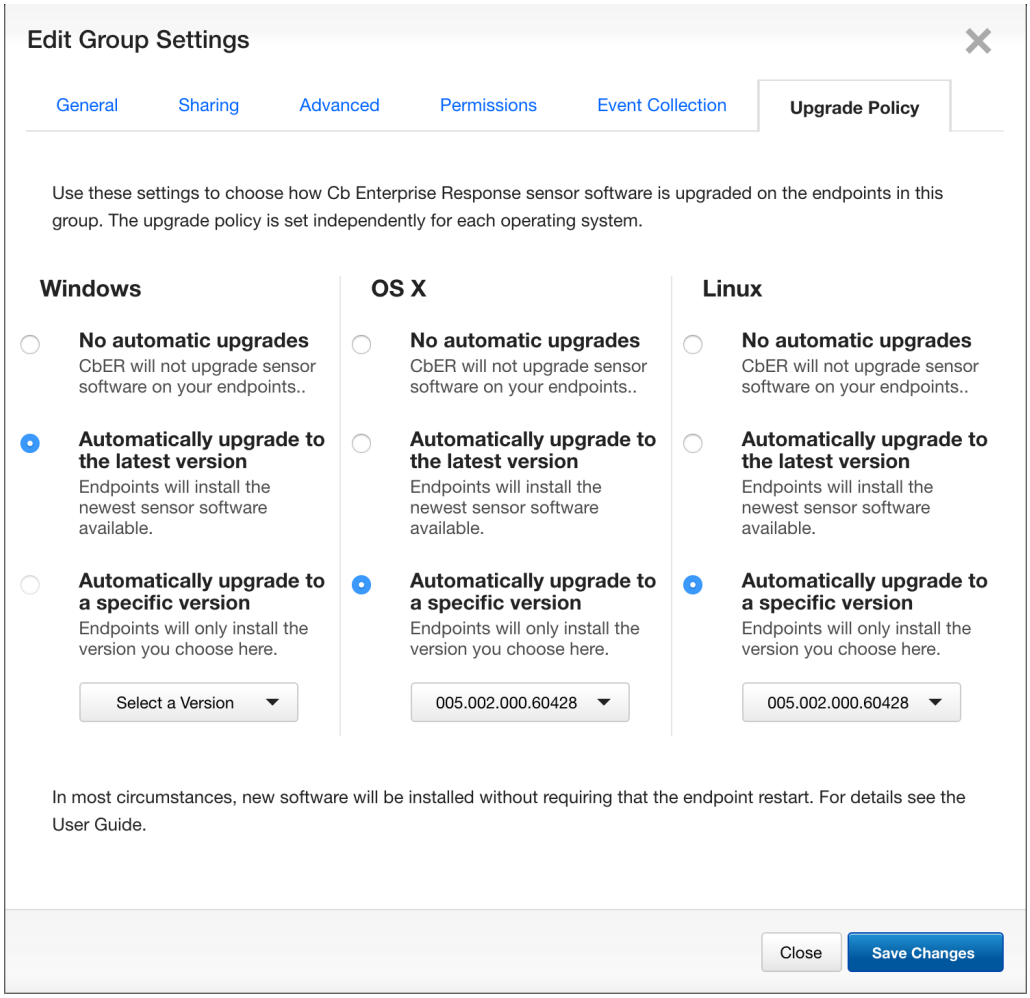


2. Under the Advanced tab, find the Upgrade Policy setting. If this is set to **Always Latest**, the server will automatically upgrade sensors in this group to the latest sensor version.
    a. To keep the sensors at a specific version, select that version number from the dropdown prior to upgrade.
    b. To continue using whatever sensor versions are already installed, regardless of version, select **Manual**.

Note: Automatic upgrade settings for Sensor Groups apply to Windows sensors only. To change OS X and Linux sensor upgrade settings please see the “Installing Sensors” chapter of the *Cb Response User Guide*.

Versions 5.2.0 and Above:

1. Log into the console, navigate to the Sensors page, and edit the group settings for each active Sensor Group:



2. Under the Upgrade Policy tab, find the platform type you would like to configure. If this is set to Automatically upgrade to the latest version, the server will automatically upgrade sensors in this group to the latest sensor version.
    a. To keep the sensors at a specific version, select that version number from the dropdown prior to upgrade.
    b. To continue using whatever sensor versions are already installed, regardless of version, select **No automatic upgrades**.

Updating Cb Response Server

If you are upgrading the server, please follow the steps in this section. These steps require SSH or console access to the server and minions with root privileges.

To upgrade a standalone server:

1. On the server, stop the Cb Response services: `service cb-enterprise stop`.
2. Update the Cb Response services: `yum update cb-enterprise`.
3. Restart the Cb Response services: `service cb-enterprise start`.

To upgrade a clustered server:

1. On the Master server, navigate to the cb install directory (defaults to /usr/share/cb) and stop the Cb Response services: `./cbcluster stop`.
2. Update the Cb Response services on each Master and Minion server node: `yum update cb-enterprise`.
3. On the Master server, restart Cb Response services: `./cbcluster start`.

Note: Improvements of Cb Response server will occasionally require using a utility called 'cbupgrade' (after `yum install/update cb-enterprise`) to migrate the database schema or alliance feed data. Upgrading from a previous stable version of Cb Response server to the current release does not require this step. However, running the utility is required when there are local changes to configuration files that have to be manually consolidated with the newer versions distributed by this release. The operator will be notified of this requirement when attempting to start the cb-enterprise services. In a clustered server configuration, this utility will need to be run on all nodes before restarting the cluster. *When running this utility in a clustered environment, be sure to answer 'NO' when asked to start server services; the administrator will need to use 'cbcluster' to start the clustered server.*

New and Modified Features

This section lists new and modified features in this version of Cb Response.

Cb Response 6.1.2 Feature Changes

This section describes product functionality that has changed in this release.

New File Extension Indexing

With this release any filemod, modload, regmod, and process path field that contain a file extension will have the extension indexed separately so that extensions can be searched efficiently. In the past, the only way to search for file extensions was by using leading wildcard queries, which are known to be very resource-intensive and can cause performance problems.

File extensions are now separately indexed as individual tokens and can be searched without using leading wildcards.

For example, a filemod of:

```
c:\windows\system32\notepad.exe
```

You can now search:

```
filemod:.exe
```

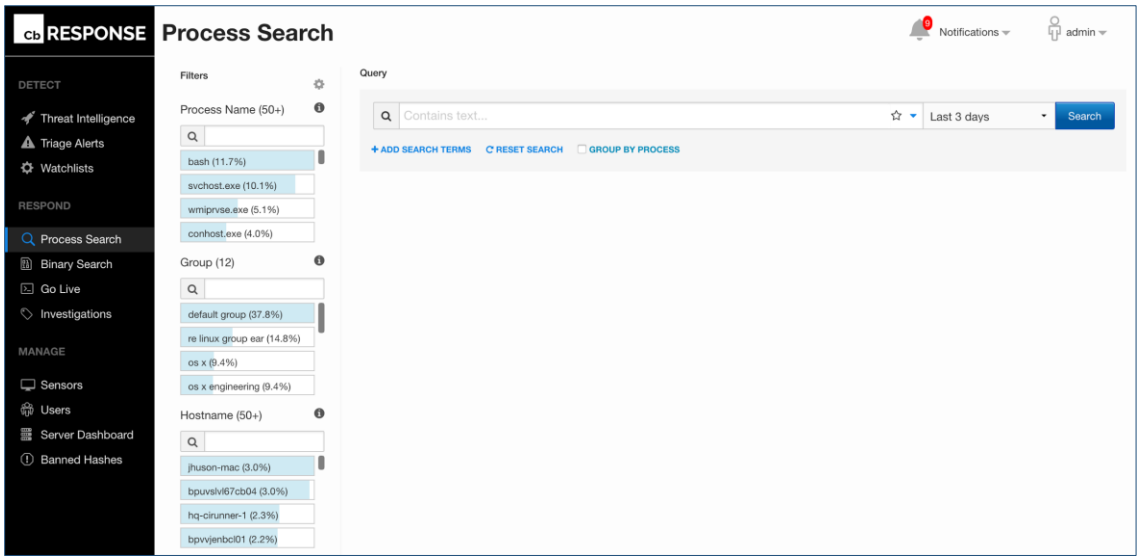
Cb Response 6.1 and 6.1.1 Feature Changes

The following product functionality have changed in this release:

- New navigation bar
- New Heads-Up Display (HUD) page
- Improved command line search
- Changes to Ingress filtering
- Process event exclusions (OS X only)
- Filter known binary module loads (extended supported on OS X)
- Improved Watchlists page

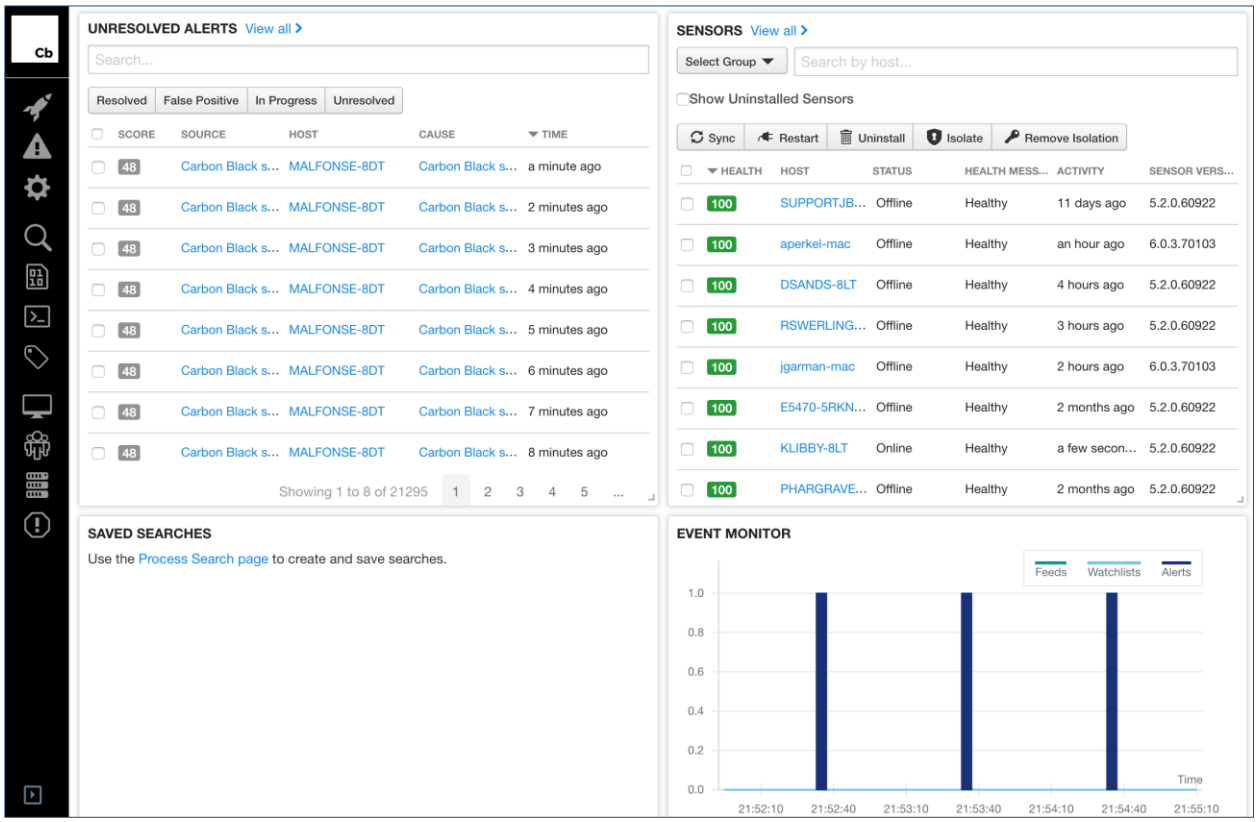
New Navigation Bar

In v6.1.0, the navigation bar has been moved to the left-hand side and redesigned for easier access to different workflows. The new navigation bar leaves more vertical screen space to display content and allows users to collapse the navigation area to reclaim horizontal screen space.



New Heads-Up Display (HUD) Page

Detect Dashboard has been redesigned in v6.1.0 to allow one-stop access to all critical information regarding the status and health of your endpoints. It is now the landing page after signing into the product. The new dashboard encompasses alerts, endpoints, saved searches, and enterprise statistics (event monitoring, endpoint hygiene, and alert resolution). Each of these components can be resized and positioned and also allow diving down into more detail.



Improved Command-Line Searches

The new tokenization and query capability introduced in v6.1.0 solves shortcomings of command-line tokenization in earlier versions of Cb Response as follows:

- Added the following characters to the set of characters removed before the command-line is tokenized:

  `\ " ' ( ) [ ] { } , = < > & | ;`

- Can now search for command-line switches starting with a "`/`" character—it is no longer treated as a path character and converted to a space.
- Added tokens of filename extensions to allow searching for extensions in addition to entire command or filenames.
- Added wildcard support for non-leading "`?`" and "`*`" characters in queries to enable search for a single character and multiple characters within a token, respectively.

Some characters are not included in this list. The "%" and "$" characters are often used for variables, so remain untouched. The "-", "." and "_" characters are often parts of file names so those also remain untouched. Other characters that remain a part of tokens include "^" and "@" and "#" as well as "!" and "?".

Handling of the following characters has changed:

- Slash ("/") – At the start of a token, it is treated like a command line switch, unless it is the start of the entire command line. In that case, it is treated as part of the path. The result is that on Linux or Mac, absolute paths passed on the command line are tokenized as if the beginning of the path were a command line switch. For example, a command line `/bin/ls /tmp/somefile` produces the tokens "`bin`", "`ls`", "`/tmp`", and "`somefile`". This compromise is due to the inefficiency of the parser distinguishing between a command-line switch and a UNIX-style absolute path.
- Colon (":") – At the end of a token, it is treated as something the user intends to search for, such as a drive letter, so it is included. If there are multiple colons at the end, or if the colons are not at the end of a token, they are converted to white space for tokenization purposes.

Based on the aforementioned rules, consider the following command line:

```
"C:\Windows\system32\rundll32.exe" /d srrstr.dll,ExecuteScheduledSPPCreation
```

This is now divided into the following tokens:

c:	windows	system32	rundll32.exe	.exe	/d	srrstr.dll	.dll	executescheduledsppcreation

As a result of this tokenization, you can now do the following:

- Search for "`.exe`" or "`.dll`" as part of the command-line query.
- Search for "`/d`" as a command-line argument without returning false positives by searching only for "`d`".
- Search for strings such as "`execute*`" to find a specific term passed to the command line.

Also, single- or double-quotations no longer result in odd queries that include them as part of a path or a command name.

Note: *This tokenization is disabled by default. This feature can be enabled in the `/etc/cb/cb.conf` file by adding the following line:*

```
CurrentEventsSchema=cbevents_v2
```

Previously configured watchlists that include command-line tokens may require rewriting to take advantage of the new tokenization. Carbon Black recommends that you review your watchlist entries to make sure they return the intended results.

Changes to Ingress Filtering

Previous versions of Cb Response allowed filtering of sensor traffic using a static JSON filter definition file that was loaded on server or cluster startup. This file (typically called "`ingressFilter.json`") needed to be placed on server nodes and was referenced from `cb.conf` file through the

IngressFilterConfigFile property. It was assumed that all cluster nodes had an identical filter definition file.

Example of such reference in 5.2 `cb.conf`:

```
IngressFilterConfigFile=/etc/cb/ingressFilter.json
```

In Cb Response 6.1, the ingress filter configuration file is no longer used. Instead, the server now uses a faster and more flexible data model for ingress filtering and has a set of APIs for managing ingress filters.

Legacy JSON filter definitions will be migrated to the new data model during the server upgrade, at which point reference to the file in `cb.conf` will be removed and file will be renamed by appending the suffix "`.61_upgrade`".

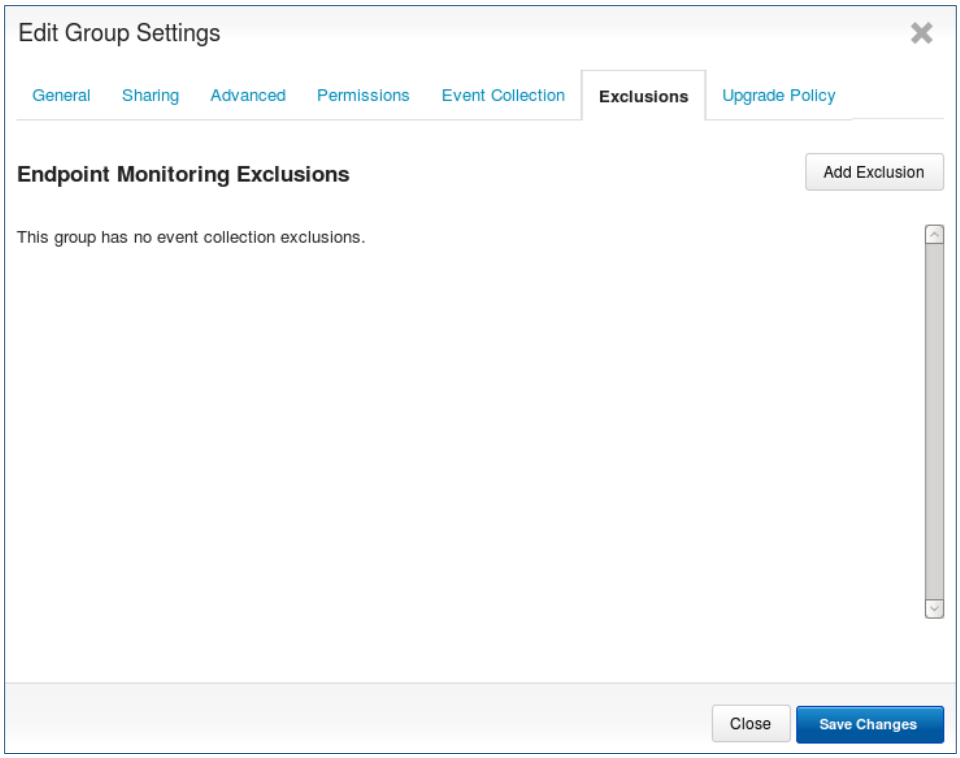
If you have a Cb Response cluster, note that only the filter file on the master node will be migrated to the new data model and this filter will apply across the entire cluster (again, assuming that all nodes of the cluster had identical filter definition file).

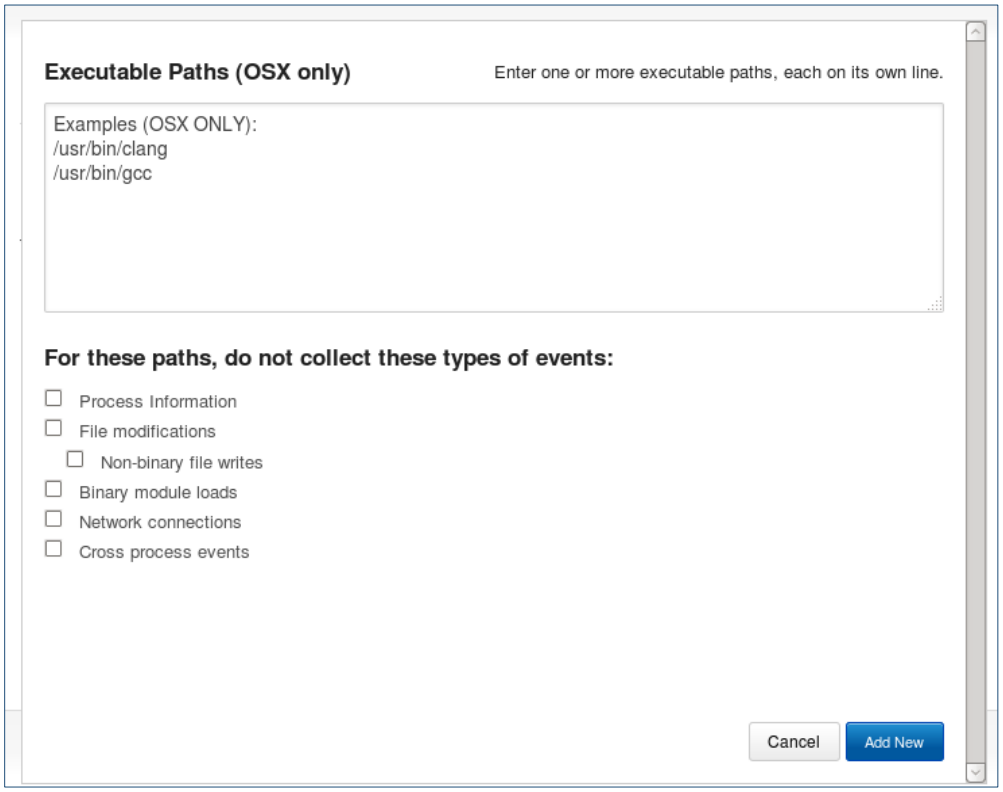
Ingress filters found prior to Cb Response upgrade to 6.1 will continue to be enforced within the new data model. However, future changes to the definition file will no longer be incorporated by the server on the startup. If you want to modify your ingress filters after the upgrade, contact Cb Support for more information about ingress filter APIs.

Process Event Exclusions (OS X Only)

In OS X sensor versions 6.0.4 and 5.2.7, a new feature is introduced to exclude collection of certain process events from OS X hosts based on the path of the parent process. This feature allows users to fine tune the performance of their OS X endpoints vis-à-vis the desired event visibility by selecting the events they would like to stop collecting from a given executable. The settings are applied on a per Sensor Group basis.

*This feature is currently supported **only** on OS X platform for Cb Response OS X sensor versions 6.0.4 and 5.2.7 and above.*



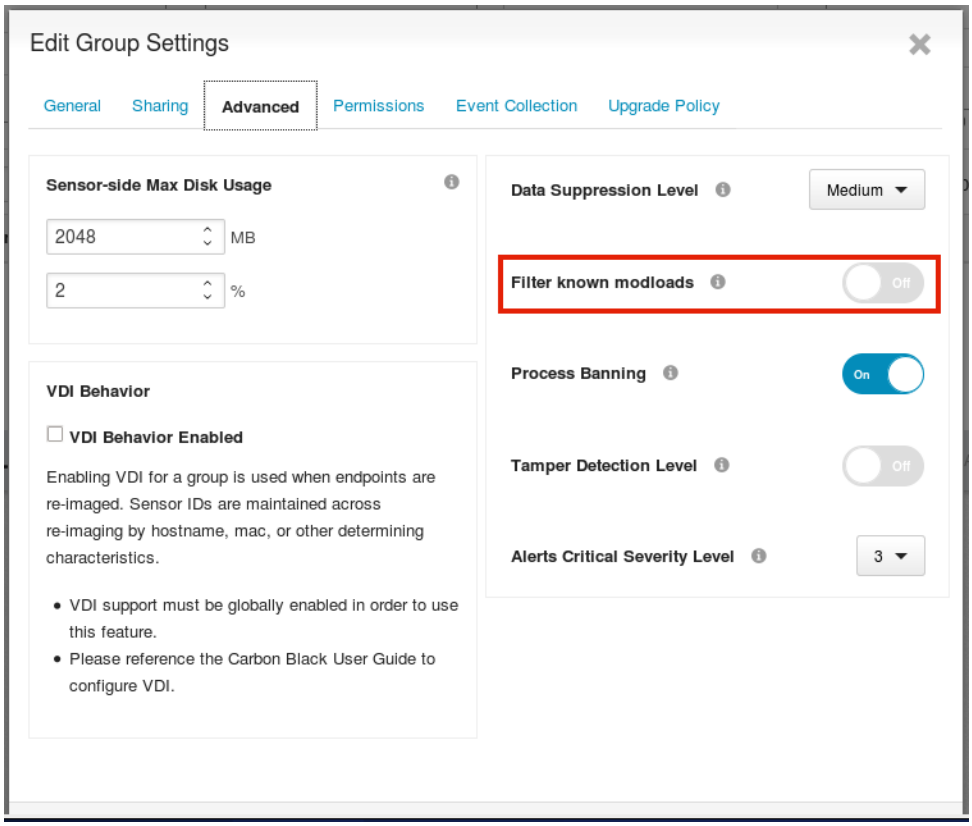


The feature can be enabled by adding `EventExclusionsEnabled=True`, in `/etc/cb/cb.conf` file and restarting services.

For best performance, Carbon Black recommends configuring exclusions for events generated by certain build and compile executables.

Filter Known Binary Module Loads (extended supported on OS X)

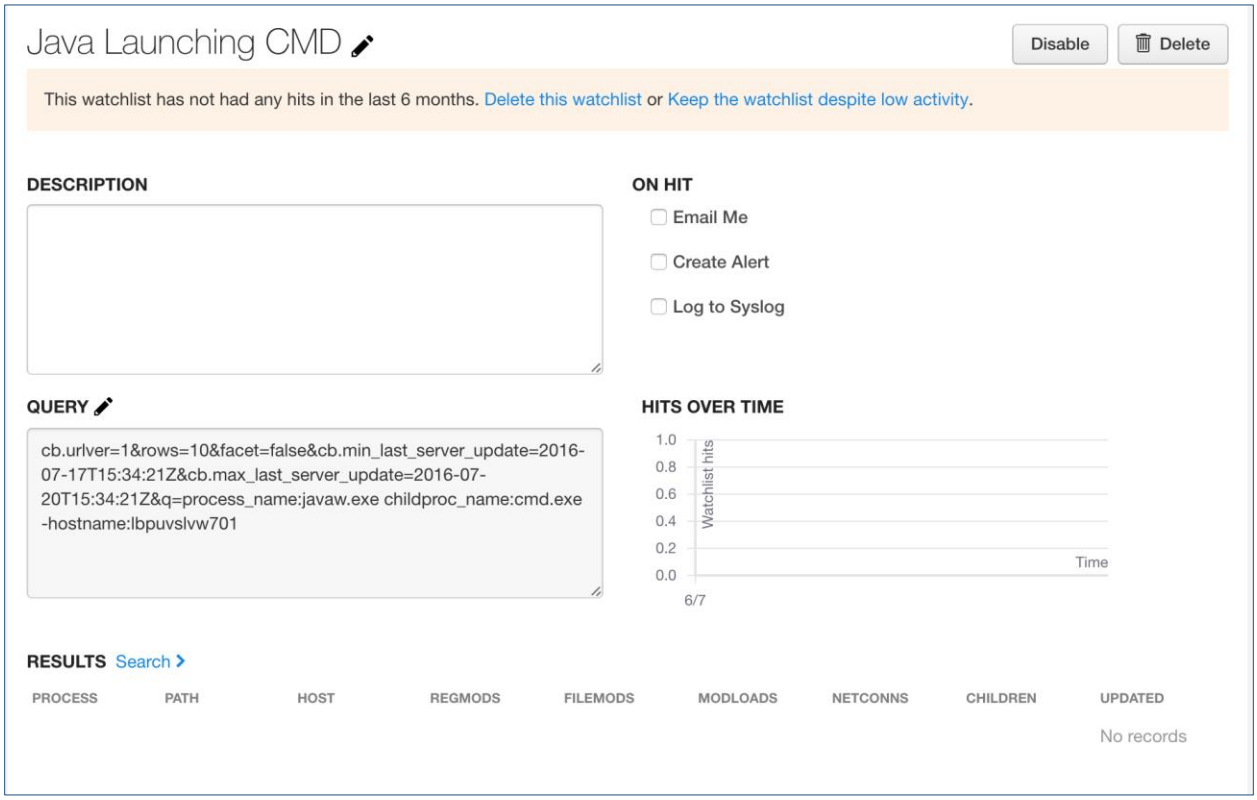
Filtering of known Binary Module Loads (previously implemented in 5.2.0 as Windows platform only – Filter Known Windows DLLs) is now extended to OS X support and renamed as "Filter known modloads" under Sensor Group – Advanced settings tab to reflect the cross-platform support. This feature is disabled by default. The known modules on OS X are determined based on dyld_cache entries under `/var/db/dyld`.



Carbon Black recommends enabling this feature on OS X sensor groups, especially if endpoints are used for XCode software builds.

Improved Watchlist page

Version 6.1.0 introduces a completely new Watchlists page. It's now easier than before to manage your watchlists: create or delete them directly from the watchlists page, or create them from your process and binary searches as before. You can quickly search and filter your watchlists to see which are enabled, or not yielding any hits. We also call attention to watchlists that have not produced hits in a time period that you choose.



In this version the watchlists are also editable allowing users to edit the watchlist query in place from the Watchlists page.

Cb Response 6.0 Feature Changes

The following product functionality have changed in this release:

1. Process Search Page
2. Process Analysis Page

Process Search Page

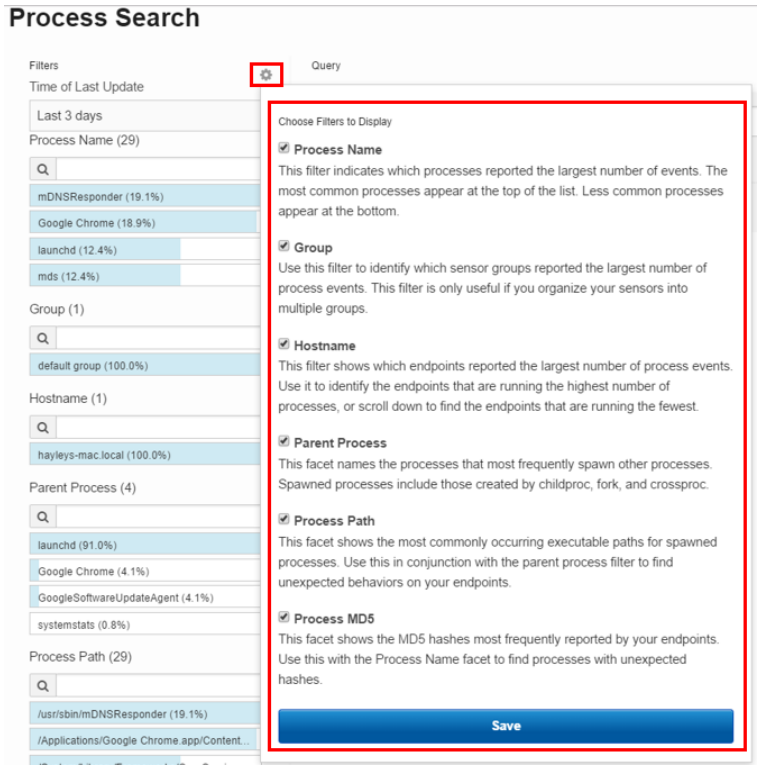
This section introduces changes in the Process Search page.

Enable/Disable Filters

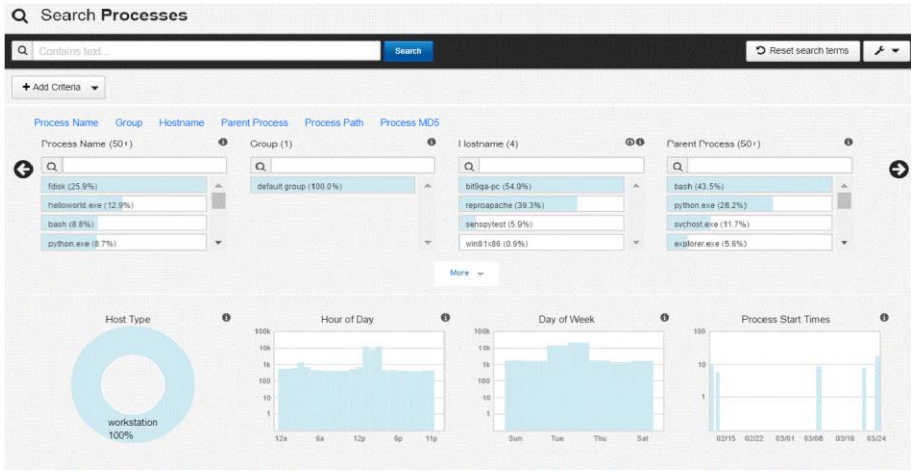
You can enable/disable filters by selecting the **Gear** icon to the right of **Filters**. The **Choose Filters to Display** page appears. Use the checkboxes to the left of the filters to enable/disable those that you

want to display. Disabling a filter removes it from the view, and if it is part of the search query, those pieces of the query are removed. Enabling a filter places it back in view. Click the **Save** button at the bottom of the **Choose Filters to Display** window to save your selections:

New UI



Old UI



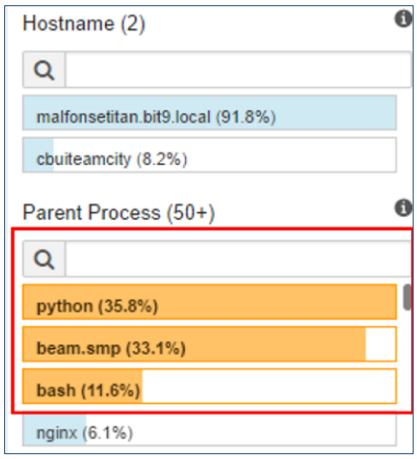
Select Multiple Filters

You can select specific filter rows within a filter table using your cursor. The search results are updated based on these selections.

Selecting multiple rows within a single filter updates the query with a logical OR between those filters. For example, choosing “bash” and “nginx” in the **Process Name** filter shows events related to either bash or nginx.

Selecting multiple rows across multiple filters updates the query with a logical AND between those filters. For example, choosing “bash” in the **Process Name** filter and “python” in the **Parent Process** filter shows instances of bash that were spawned by Python.

Selected filter rows are highlighted as yellow. To deselect a filter row, click it a second time.



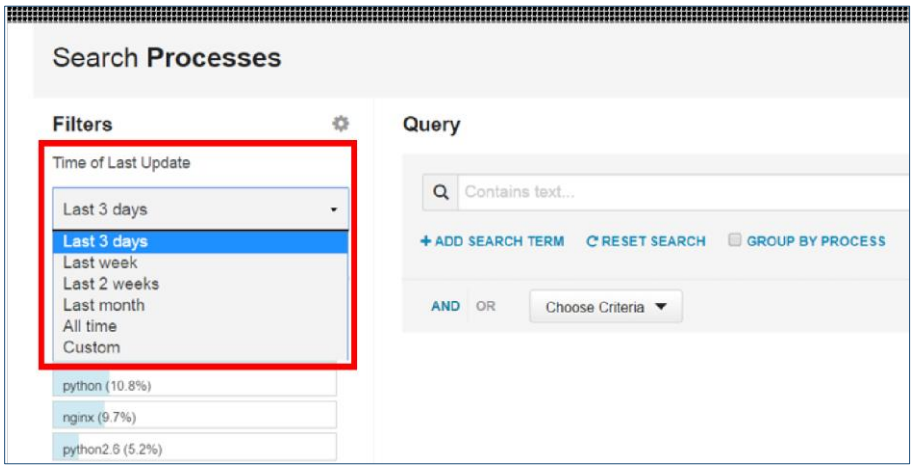
Time Filters

In the **Process Search** page, you can filter processes that were updated with events within a specified time period. For example, if you select the **Last 3 days** option, the search results will show processes that were updated with events within the last three days:

Time filters include:

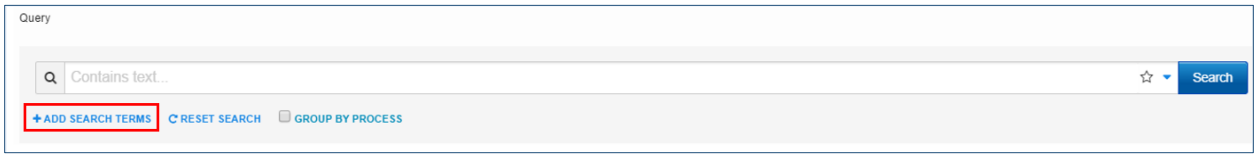
- **Last 3 days** – Displays search results based on process data that was updated over the last three days.
- **Last week** – Displays search results based on process data that was updated over the last week.
- **Last 2 weeks** – Displays search results based on process data that was updated over the last two weeks.
- **Last month** – Displays search results based on process data that was updated over the last month.
- **All time** – Displays search results based on process data that was updated over the life of the server.

- **Custom** – Allows you to create a custom time filter. Selecting this option presents the **Start Time** and End **Time** calendars. You can use these to select a time range (down to the very hour/minute) within which you can query processes. This queries for processes that have been updated within the selected "sensor time range", as opposed to when the server stored the event.



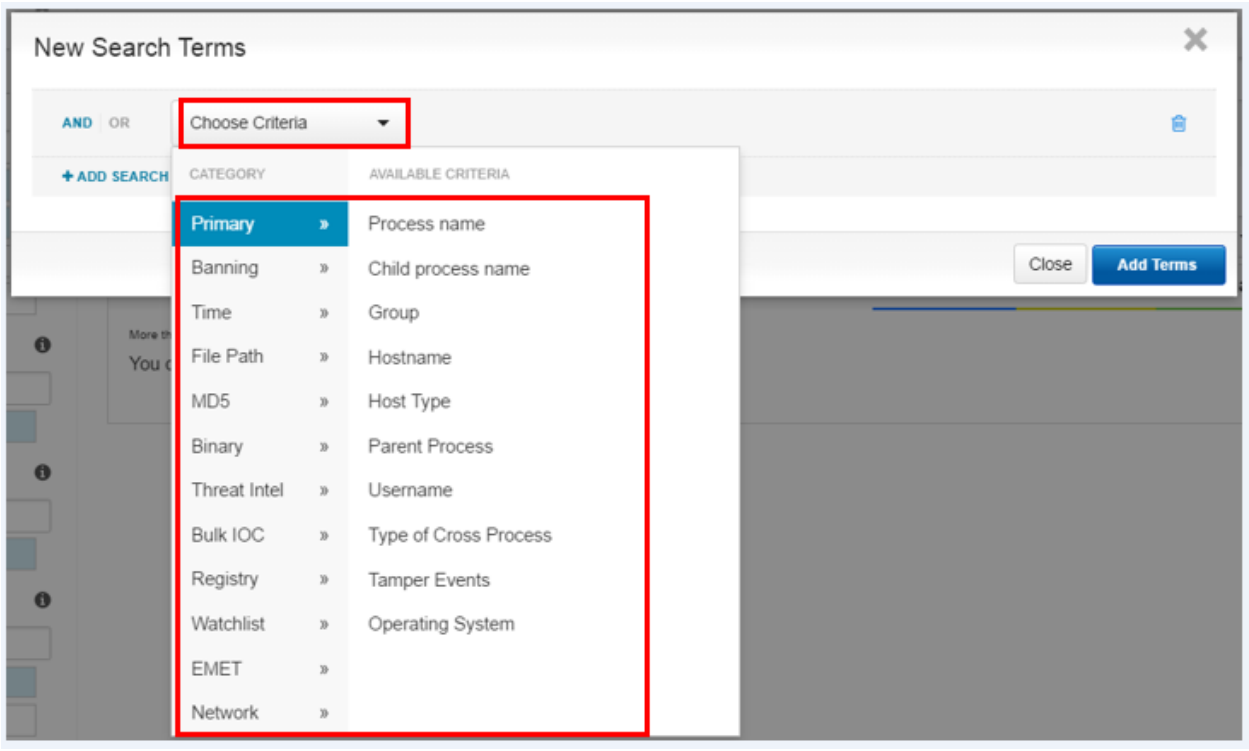
AND/OR Operator Searches

Process searches explicitly support AND/OR operators. You can select from an array of filters to form your search using these AND/OR operators. You can add as many search terms as needed (in the form of AND/OR operators) by clicking **Add Search Terms** on the Process Search page below the **Search** field:



This displays the **New Search Term**s window:

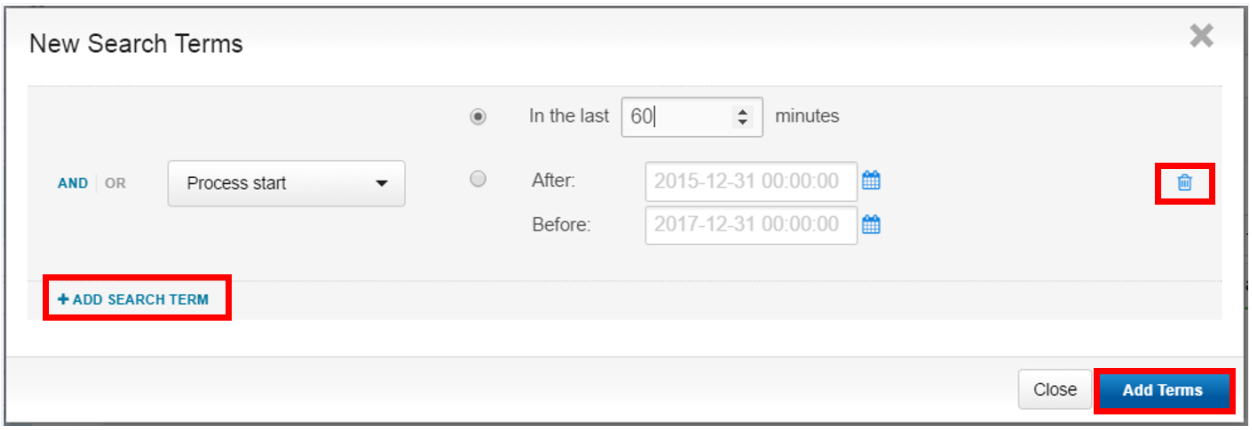
In the New Search Terms window, click **Choose Criteria** to open a drill-down list of search terms from which to select. Use this list to select and then define a search term:



When you have defined a search term, you can add additional search terms by clicking **Add Search Term**. In the example below, we added a **Process start** search term that will display processes that have started in the last 60 minutes.

To remove a search term, click the **Delete** (trash can) icon to the right of the search term.

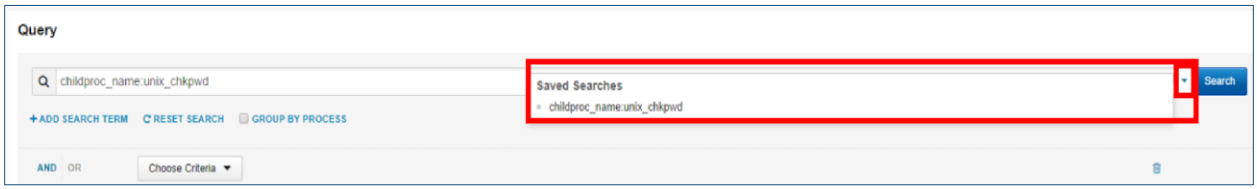
When you have added all search terms, click **Add Terms** in the bottom-right corner of the New Search Terms window to add the search terms to your search:



Saved Searches

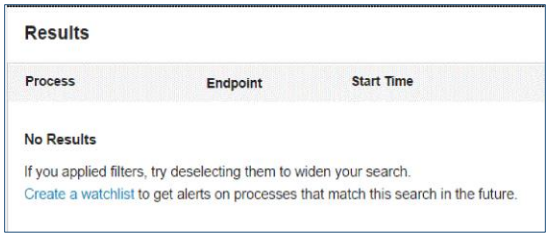
You can save frequently executed searches by selecting the **Favorite** (star) icon to the right of the **Search** field. A confirmation appears in the top-right corner of the console indicating that the search has been saved.

To execute a saved search, click the down arrow to the right of the **Favorite** (star) icon and select the saved search from the drop-down list. The selected saved search is loaded and executed:

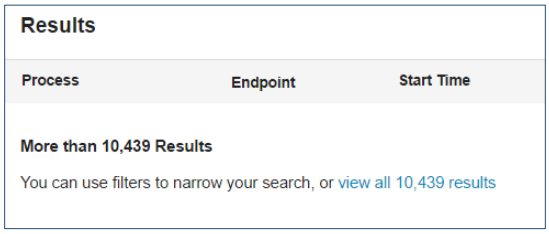


Search Results Warnings

If no search results match your search criteria, the **Process Search** dialog displays the following warning:



If search results are too large, the **Process Search** dialog displays the following warning. You can select one of the options below to pare down the search results:



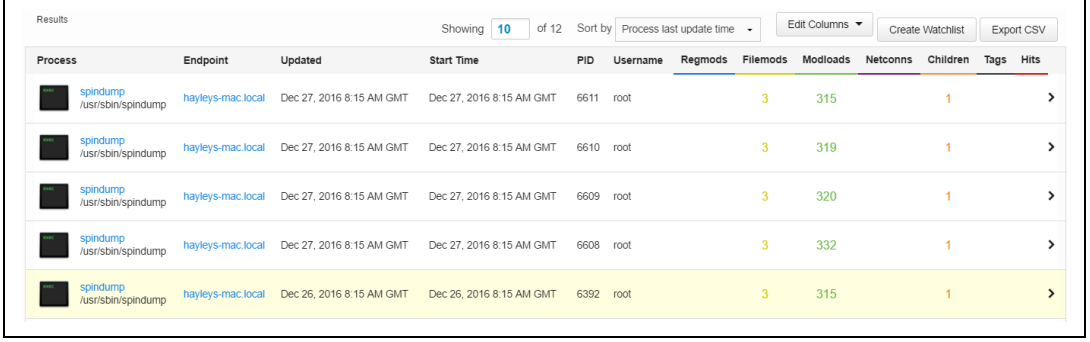
Search Results Use All Space in Browser

Past iterations of Cb Response only used 980 pixels of real estate; on a 1080p screen, we only used about half of the available space. This makes it hard to display all of our data in so little space! As a result, we have changed the page to use all of the space that customers want to use. The **Results** table now scales to the full width of the screen.

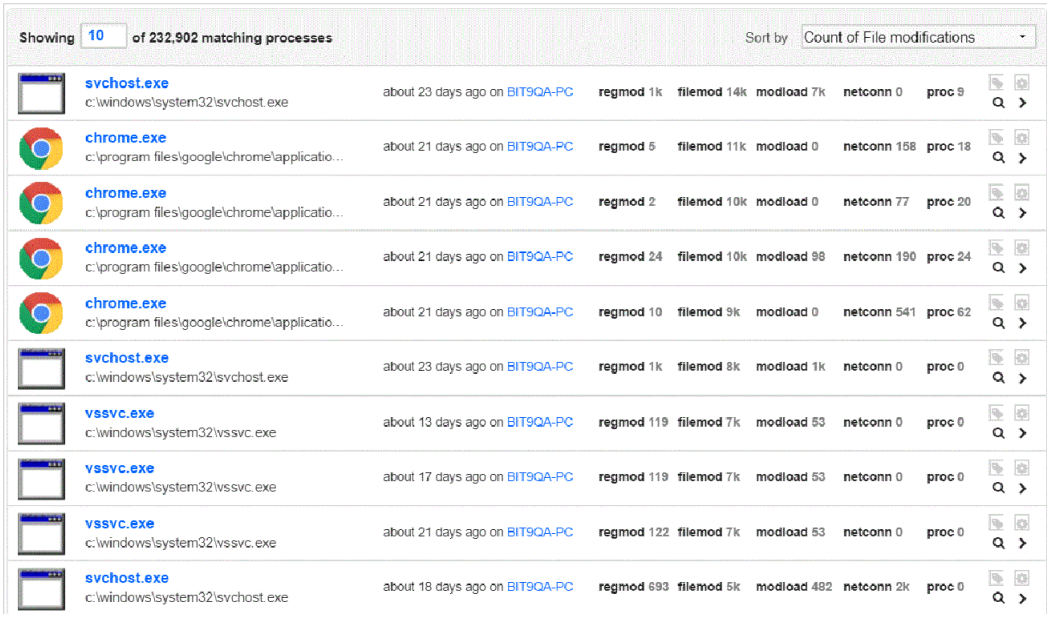
Search Results Table Events

The **Search Results** table events contain color-coded columns to indicate the type and number of processes in the search results that triggered events:.

New UI



Old UI



Legacy Data Limitations

This section applies only in case you have upgraded your server from older 5.*x* version. In this case, 5.*x* processes will be available for searching for 30 days with two limitations:

Filters on the left hand side of the process search will not include historical 5.*x* data, but only data received since the 6.0 upgrade

Get Comprehensive Results button appears on the **Process Search** page if a search query spans both 6.0 and 5.*x* data and the query has complex search terms requiring special processing on the server.

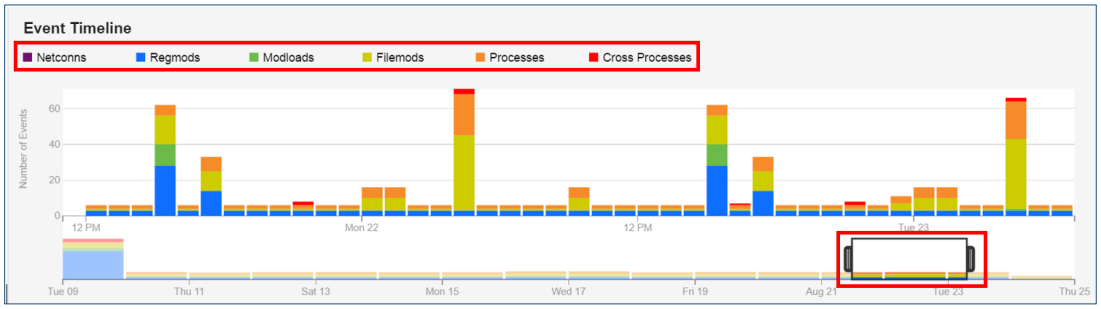
- If you don't require comprehensive results, server will return correct results for 5.*x* data, but results for the new 6.0 data might be incomplete
- If you decide to get comprehensive results for the query, server will return full results for 6.0 data, but completely exclude 5.*x* data in the search results.

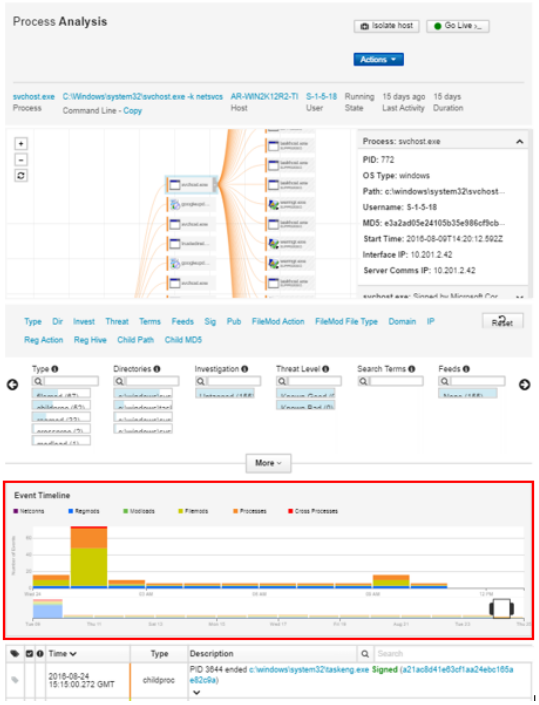
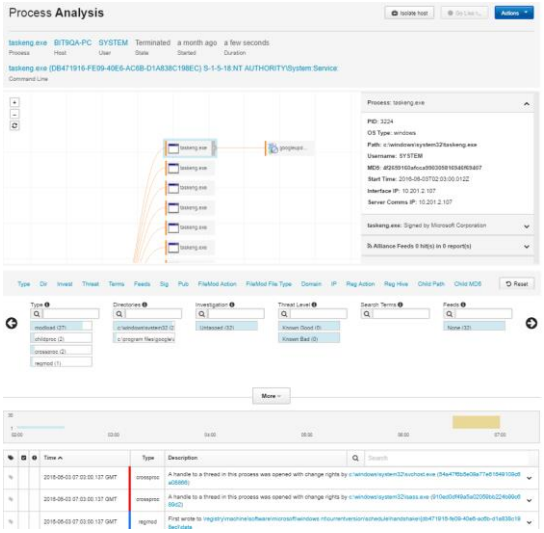
For more information, see [What has changed in terms of how Cb Response stores process data?](#) on page 26.

Process Analysis page

The **Process Analysis** contains an interactive **Event Timeline** that introduces these features:

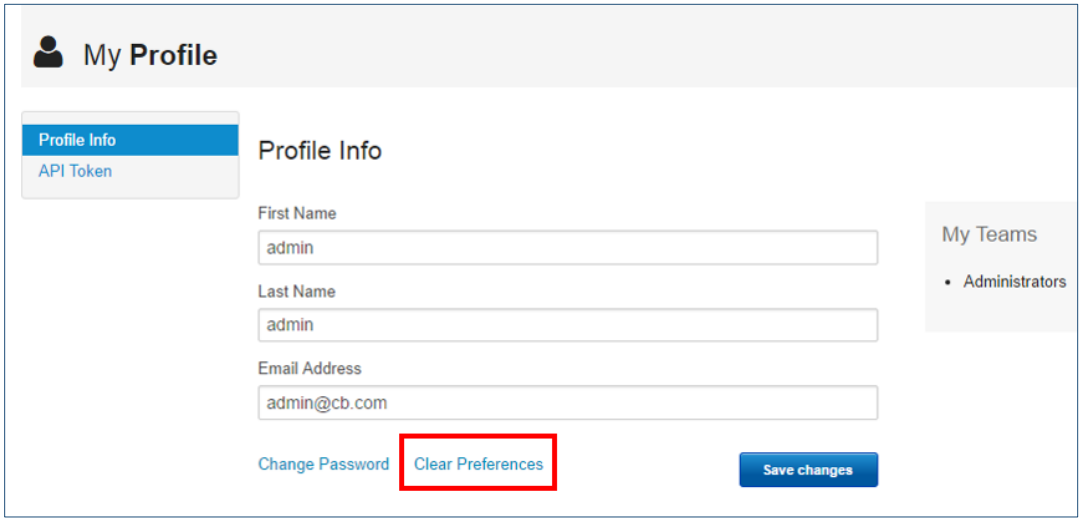
- A legend of color-coded event types appears at the top of the timeline. These colors are carried over to the bottom two timeline graphs to represent particular event types.
- The top graph in the timeline displays event counts, which are broken down by event type.
- The bottom graph is broken down into time segments. These time segments can be days, hours, minutes, seconds, or milliseconds. This depends on how long the process has been active and creating events. Processes that start and stop in a very short time period produce a higher resolution (up to milliseconds). Processes that run for a longer period produce a lower resolution (down to days).
- The bottom graph also contains an interactive range selector widget that users can expand/collapse (by placing the cursor on the left or right side and pressing the left mouse button) and slide back and forth across the timeline. The top graph expands/collapses and slides back and forth in conjunction with the range selector. Users can essentially zoom in on event segments in the top graph to view event counts for particular time segments.



New UI	Old UI
	

My Profile Page

A new **Clear Preferences** button exists on the **My Profile** page (accessible by clicking your username in the top-right corner of the Cb Response console and selecting **My Profile**). This button allows you to clear user preferences, such as searches that you have saved in the **Process Search** page.



What has changed in terms of how Cb Response stores process data?

Cb Response v6.0 server stores per-process information across smaller documents inside its data store. This data includes all events reported by a process, such as file/registry modifications, network connections, and so on. Each document represents a segment of the running process activity, which is limited by the size and time of the process they cover.

When using the 6.0.*x* sensor, each process document segment will be limited to 5 minutes of process activity or 10 MB of event content (whichever comes first). When using older (5.*x*) sensors with the 6.0 server process document segments in the data store, this timeframe can be as short as 30 seconds.

While this approach will greatly boost the performance and scalability of Cb Response servers, it has some impact on search results for specific types of events or combinations of events.

Impact of process segmentation on complex queries

Document segmentation makes searching for data more complex in cases when search involves terms for events, including filemod, regmod, netconn, domain, modload, and crossproc.

For example, if a cmd.exe process modifies two files (A and B) and you search for “filemod:A AND filemod:B”, the results from searching the specific cmd.exe process in the search results are dependent on the two modifications falling within the same process segment. A second example involves searching for “NOT filemod:A”. The search excludes process segments where filemod:A was not detected, but other segments of the same process would still be returned.

Server addresses this by using Solr ability to join all process segments when doing the search. This “comprehensive” mode of search will be used by the server automatically as needed, and it might cause queries with above-mentioned conditions to execute slower.

One limitation of comprehensive search is that it cannot be applied to historical data collected while the server was on version 5.*x*. If requested query happens to search through the 5.*x* data and uses logic that would require comprehensive search, server will show a warning message, as described in the previous sections of this document.

In that case, a query can be re-executed with the more expensive (comprehensive) search option, but exclude results from older 5.*x* data.

Duplicate query results

Because there will be multiple process document segments, some searches will return more than one result per process. The search UI has a new **Group by Process** option (discussed in Process Search Dialog) to de-duplicate results and show a single entry per process. De-duplication of results can be costly in terms of both memory and CPU and take a long time when the number of results is high. For that reason, server will silently ignore request to group result by process if number of results exceeds 1 million.

Impact of process segmentation on “count” queries

Cb Response 6.0 servers store event counts (for example, filemod_count, netconn_count, modload_count, regmod_count) in process segments differently than 5.*x* servers. Each segment has a cumulative count of events up to that point. For that reason, the server must handle event count conditions in a special way:

- “Less than equal to” (for example, netcon_count:[* TO 10])
- “Equal to” (for example, netcon_count:10)
- “Between” (for example, netcon_count:[5 TO 10])

Given these conditions, the server considers only the last segment of each process (the one that includes the process termination flag). This is because the final count is known at this point. This helps avoid false-positive results where a query may return progress segments that satisfy the condition, such as netcon_count::[* TO 10]). This occurs even if the process creates more conditions and should be eliminated from the search results. Following this logic, use of the conditions listed above avoids returning processes that are still alive and have not yet sent the termination flag.

Note: The following remaining condition:

“Greater than equal to” (for example, netcon_count:[10 TO *])

…still returns live and terminated processes, since no danger of false positives exists in this case. Processes exceeding the network count in a segment would exceed this condition for all subsequent segments as well.

Support of Multiple Volumes for Event Data

This section explains how customers can add more storage to their existing Cb Response deployment after upgrading to the latest release. This involves adding multiple Solr data directories for cbevents cores. These directories can be added as mount points into new storage arrays, so that you can easily add more disk space. If you need more disk space, attach a new volume, mount it into the Solr data directory, and the server starts using it automatically.

Naming Conventions

Solr uses new cbevents directories (mount points) if their name is prefixed with:

cbevents*

or

_cbevents*

Warning: The cbevents directory (without the suffix) is the default directory but does not need to remain on the original data partition. You can remove it if needed.

The following is an example of a valid multi-volume configuration:

```
[root@ip-172-31-14-184 solr5]# df
Filesystem     1K-blocks     Used Available Use% Mounted on
/dev/xvda1      32895856 10341996  20860168  34% /
tmpfs           31389104        0  31389104   0% /dev/shm
/dev/xvdb      206293688 28033360 167758184  15% /data
/dev/xvdf      206293688 35809668 159981876  19% /data/solr5/cbevents2
/dev/xvdg      226936188 63976332 151409136  30% /data/solr5/cbevents3
```

In this example, the default data drive is mounted to /dev/xvdb, and /data is configured as the data root inside of `cb.conf`. In addition, two more volumes are added and mounted to /data/solr5/cbevents2 and /data/solr5/cbevents3.

Warning: The system assigns the correct user:group upon cb-enterprise restart. If you created the mount points on a live server, ensure that the user assigned to the Cb Response server has write permissions on the mounted directory. Failure to do so causes the system to ignore the new mount points.

Another option to expand cbevents storage is to use symlink as follows:

1. Create a mount point in another location in the file system, such as /data2.
2. Create a symlink to the cbevents* directory inside the solr5 directory that points to the mounted directory. For example: ln -s /data2 /var/cb/data/solr5/cbevents2
3. Ensure that the Cb Response user has write permissions in the mounted directory (/data2).

Using New Data Directories

This section discusses partitioning and purging relating to new data directories.

Partitioning

New data directories are used when the next partition occurs (every three days by default) or sooner if the current data disk becomes full. The server uses simple heuristics in calculating when to partition and where to place the new event partition:

1. A new partition is created in the cbevents* directory with the most free space at the time of partitioning.
2. If the current data volume is more than 95% and additional partitions exist that have more than 5% free space available, the server immediately partitions.

You can control this threshold using the following configuration parameter:

SolrTimePartitioningFreeSpaceThresholdPerc

- Rule 1 ensures that new volumes are used in a balanced fashion. As old data is aging out (being purged), some partitions free up. This ensures that free space is optimally used.
- Rule 2 ensures that the system uses fragmented disk space efficiently in case many cbevents* directories exist. For example, assume you have five volumes, and each has 20% free space. This could result in none of the volumes fitting into the three-day partition. The system will continue trying to use one of the partitions (up to its maximum available space) before moving to the next one. As a result, the server might end up with smaller partitions. However, this scenario should be rare.

Active Directories

Any cbevents directories prefixed with cbevents* will be used to create new cbevent partitions.

Retiring Directories

Any cbevents directories prefixed with _cbevents* will be used to load existing partitions, but new partitions will not be created on it. This approach can be used when retiring old volumes. Old partitions will eventually be purged based on time.

Partition Purging

The system purges partitions based on disk space, time, or the maximum number of allowed partitions.

When purging based on disk space, a purging algorithm considers the overall amount of free disk space. For example, assume three 100 GB volumes exist, each with 30 GBs of free space. This gives you 90 GBs of free space and a total disk space of 300 GBs. The total event data size is the sum of index sizes on all three volumes. (This could be less than 210 GBs since the main data volume may also contain store files and other data.)

The following shows how the current purging thresholds (in `cb.conf`) are interpreted when multiple volumes exist:

- **MaxEventStoreSizeInPercent** – Purge the oldest partition when the total sum of all event core sizes exceeds the given percentage of a total disk space (on all volumes).
- **MaxEventStoreSizeInMB** – Purge the oldest partition when the total event store size (on all volumes) exceeds the given threshold.
- **MinAvailableSizeInMB** – Purge the oldest partition when the total free disk space (on all volumes) falls below the given threshold.

Extending Disk Space on the Fly

You can add disk space on the fly without having to restart their Cb Response server. New directories are automatically used when a new partition occurs, avoiding any server downtime.

What has changed in terms of servers/clusters?

Cb Response 6.1.0 allows you to remove unneeded minions from a cluster to reduce the overall size of the cluster and simplify the cluster deployment. This improves the cluster performance by reducing network overhead.

Best Practices

Be sure to adhere to these best practices before attempting to remove minions from an existing cluster. Consult the Cb Response Server Sizing (OER) Guide to ensure that the reduced cluster has the capacity to support the total number of endpoints.

- The remaining minions must have enough disk space to contain the full data.
    - Calculate the daily usage of the disk per sensor and re-calculate the required disk space to new minions for full retention. Here is an example:

        - Assume that you want to collapse the six-minion cluster that has 40K endpoints down to three minions and have retention for 30 days. This is supported by Cb Response v6.1.0, because each minion will still have less than the maximum of 18,750 endpoints.
        - You have calculated the current daily disk usage per sensor to be 20MB (dividing the total cluster data volume storage with the current daily retention and number of endpoints).
        - The new total storage requires a minimum of 20MB * 40K (endpoints) * 30 (days) = 24 TB (or 8 TB per node).
        - Add 20-30% to the available disk space on top of the calculated amount to allow for increased sensor activity in the future.
    - Additional extra disk space must be provided for desired cold storage in days.
- The master cluster node must have enough disk space to contain binary files for all removed minions. Binary files typically take less space than events but should still be taken into account. You can calculate the required extra space on the master by logging into Cb Response and navigating to **Administration > Server Dashboard** and then looking at the **Storage Statistics** for the minions. Add up the total binary size on the minions that will be removed to obtain the maximum additional space required on the master node.
- The remaining minions must have enough CPU and memory space to support the sensor load (per OER).

Read-Only Minions

Before removing minions from an existing cluster, you must convert them to a read-only state. Read-only minions are searchable throughout the API and user interface but are not used for sensor checkins or event/datastore pushes. While minions are in the read-only state, the system copies the binary files to the master.

The event data is not added to the minions and existing data is purged periodically as in normal operations. As a result, the read-only minions become completely inactive after the retention period. After the desired data retention period expires, you can safely remove the read-only minions.

Note: *Read-only minions can be reverted to active minions at any time if needed.*

Removing Minions

Perform the following steps to mark/unmark minions as read-only and to remove a cluster minion:

To mark minions as read-only:

1. Stop the cluster.
2. For each minion that you want to remove, run the following command:

   ```
   cbcluster change-node -N {node} -R True
   ```
3. (Optional) If you have an eventless master node and want it to contain events, run the following command:

   ```
   cbcluster change-node -E True
   ```
4. Start the cluster.

To unmark minions as read-only:

1. Stop the cluster.
2. For each minion that you want to remove, run the following command:

   cbcluster change-node -N {node} -R False
3. (Optional) If you want to convert master node back to eventless, run the following command:

   cbcluster change-node -E False
4. Start the cluster.

To remove a cluster minion:

1. Stop the cluster.
2. For each minion that you want to remove, run the following command:

   cbcluster remove-node -N {node}
3. Start the cluster.

Cb Response 5.2 Feature Changes

The following sections provide a quick reference to the new and modified features introduced in version 5.2.0.

Eventless (Uninteresting) Process Suppression

Starting with version 5.2, a process is classified as eventless (uninteresting) and therefore suppressed depending on the following definitions:

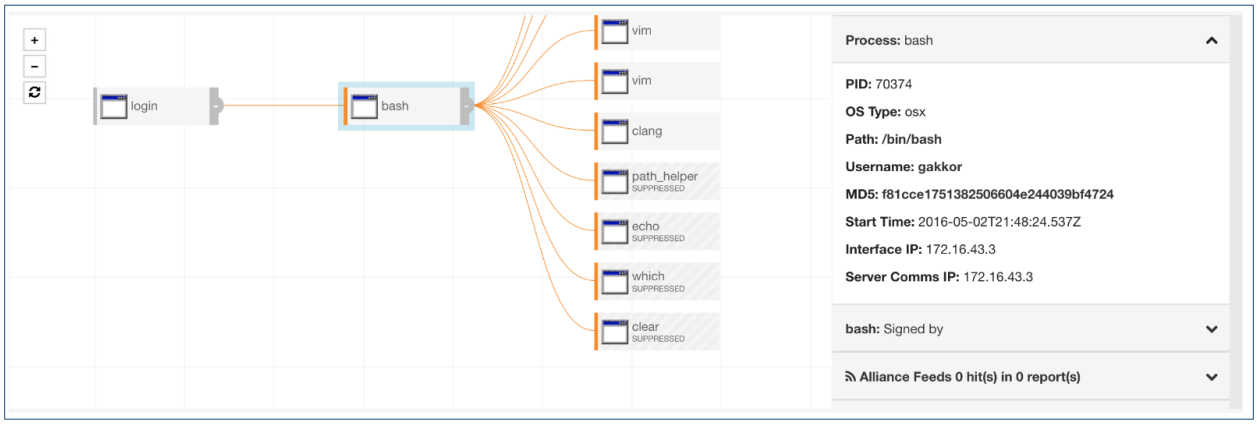
- **No suppression** – all processes are interesting, a process document is created for each process execution instance regardless of its activities, and product works as before.
- **Medium suppression** – A process that has no network connections, file modifications, registry modifications, cross-process events, or child process events is classified as uninteresting. The only event in an uninteresting process would be module loads.

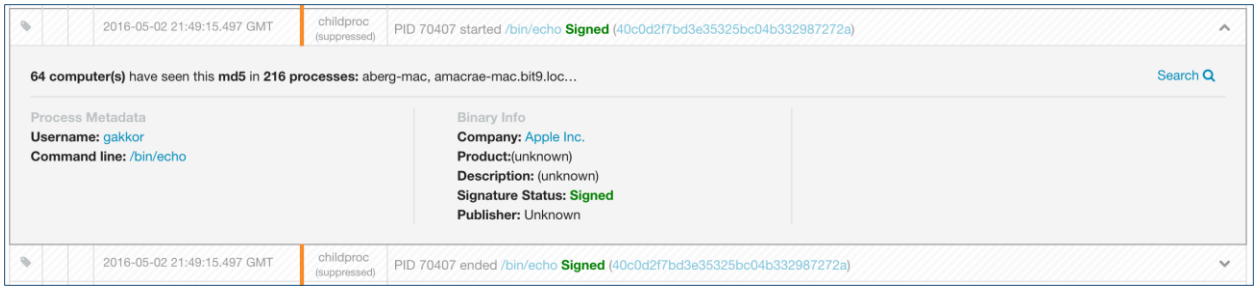
- **High suppression** – A process that has no network connections, file modifications, registry modifications or child process events is classified as uninteresting. The only events in an uninteresting process would be module loads and cross-process events.

What happens to the suppressed processes?

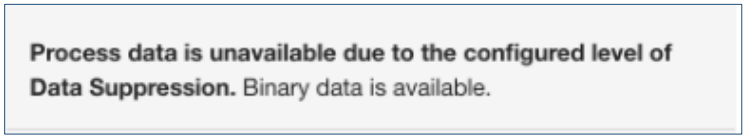
Suppressed processes are not stored/indexed by the server as stand-alone process documents. From the UI workflow perspective, this means that such processes will not have their own Process Analyze page; they can not be queried by process_name field. However, there is tracking of the execution of suppressed processes under the parent process. Version 5.2 expands the metadata details for the childproc event type under the parent to include, in addition to existing process and binary information metadata, command line and username information for suppressed processes. Such processes can still be searched by *childproc_name, childproc_md5, cmdline* and *username* field from the search pages.

The following figures show how Process Analyze page would look for a parent process that has suppressed child processes, in this example, **bash** has **echo**, **which**, **path_helper** and **clear** as suppressed child processes.





Note that there is no *Analyze* link within the event dropdown (since there is no process document), and if the process node were to be selected on the process tree, the metadata panel would warn you of the fact that this process is suppressed:



Eventless process suppression will have a pronounced impact on the number of process documents created by OS X and Linux sensors, for example, many executions of *clear*, *cat, which*, and *ls* type commands on an OS X or Linux host will have reduced data processing impact on the deployment.

Frequently Asked Questions

- Suppression levels are configurable per sensor group basis from the UI.
- Suppression is supported on all endpoint platforms (Windows, Linux and OS X).
- Taking advantage of suppression features require upgrading endpoints and the server. to 5.2. Legacy sensors will report all events as before even if they connect to a 5.2 server with suppression enabled.
- On a new install or an upgrade, ALL existing sensor groups will have their suppression level set to MEDIUM by default. The server upgrade process will notify customers of this fact during upgrade. This can be later changed from the UI.

Improved POSIX process tracking

In previous versions of the Cb Response sensor, process tracking attempted to map each process fork and each process execution into unique process instances. This resulted in creation of a high number of process documents as forks that occur in POSIX environments don't always correlate with a new logical process. Additionally, the tracking of fork() system calls was not always accurate, which under some circumstances resulted in missed or incorrect process information.

In version 5.2, the OS X and Linux process tracking becomes more nuanced. POSIX process execution is now handled differently. First, any time a process performs a fork() system call, all activity for that process will continue to be associated with the parent. A new "**fork**" event type will be displayed on the Process Analyze page of the parent, indicating that the parent process performed a fork. The PID of the forked process and the timestamp of when the fork has occurred will be recorded. The first time a process (with a given PID) performs an exec() system call, a new process document will be created and the product will track the execution as a new logical process (current child process behavior). The create time for that new execution will be reported and will correlate to the timestamp when the process was created, that is when the fork occurred.

If at any point a process performs a second (or any subsequent) exec() system call, a new process document will **not** be created. This activity will be reported as a new "**exec**" event type within the process and the process meta-data will be updated to reflect the new image and command line associated with the exec() system call.

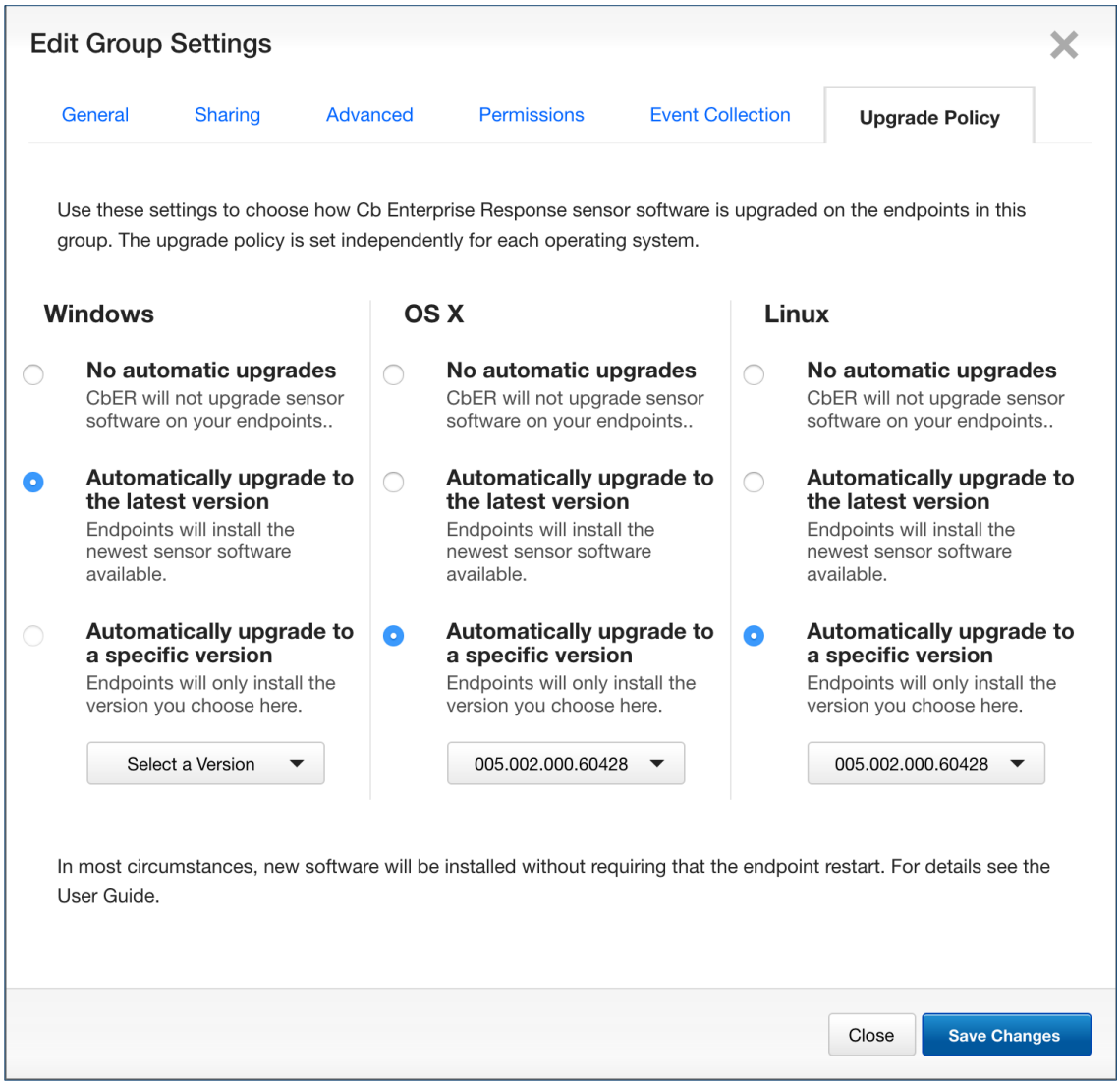
This new process tracking will reduce process document counts generated from OS X and Linux sensors considerably and give better visibility to different execution/instantiation paths. Fork and Exec type events apply only to OS X and Linux sensors. Windows sensors still report child process execution as before.

Support for the OS X and Linux sensor upgrades from UI

In this version, the OS X and Linux sensor upgrades become fully configurable and controllable via the UI. In previous versions of Cb Response, only the Windows sensor upgrade policy was configurable via

the UI on a per-sensor-group basis. The OS X and Linux sensor upgrade policy applied globally to all sensor groups at once and had to be done by editing the `cb.conf` file.

With this version, the upgrade policy for all platforms can be configured from the UI and differ on a per sensor group basis. The new upgrade policy tab on the Edit Group Settings dialog is shown below:



Configuration options that existed for Windows sensor are now extended to OS X and Linux sensors. When upgrading from a previous version of Cb Response server, the following rules will apply:

- Configuration options previously set in `cb.conf` for upgrading the OS X and Linux sensors will be ignored.
- For all sensor groups, the OS X and Linux upgrade policies will be set to manual.
- Windows sensor upgrade policy will remain the same as what was previously set for each sensor group.

Other Features and Improvements

Suppression of known Windows DLLs

In this version, sensor group settings has a new option to enable suppression of known Windows DLLs. This is a Windows platform only feature. A known DLL is a Microsoft Windows term for basic DLLs that are loaded into RAM instead of being read from disk with every single process load. When this feature is enabled, trusted DLLs are simply not sent from sensor to the server on a per sensor group setting. More information on the definition of known DLLs can be found here:

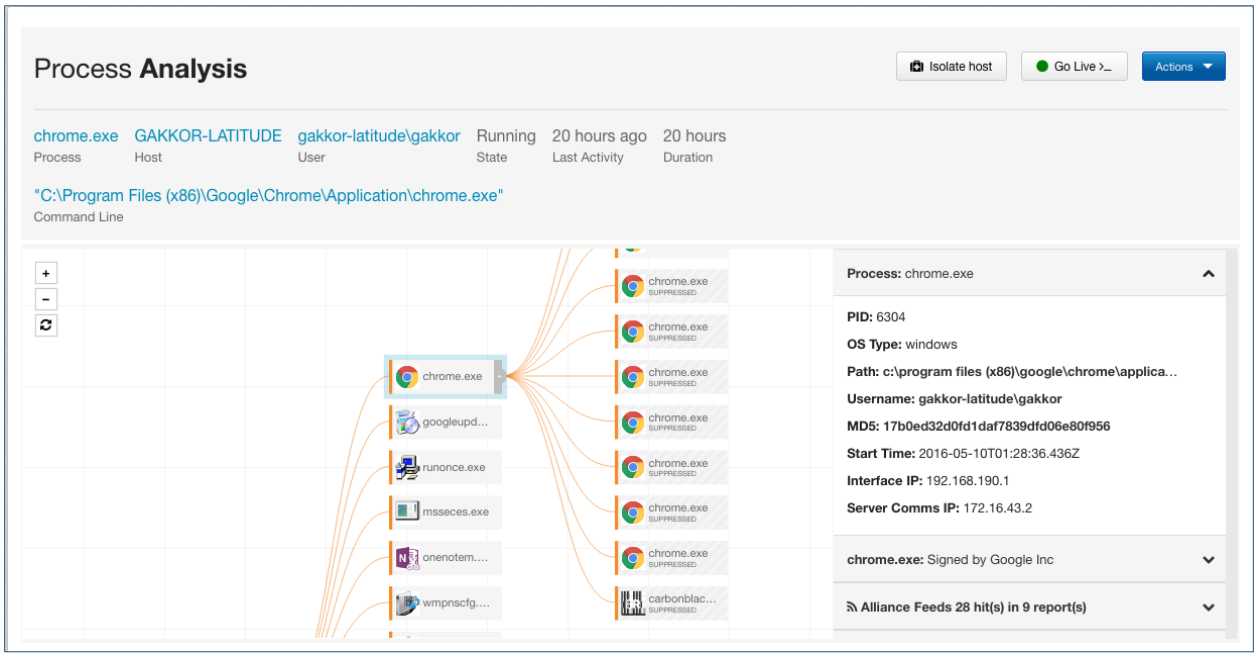
https://technet.microsoft.com/en-us/magazine/2007.09.windowsconfidential.aspx

Improved Triage Alerts page performance and workflow

In this version, Triage Alerts page is re-designed to load faster (with support for viewing more rows at a time) and provides a cleaner workflow for triaging alerts.

Redesigned Process Analyze page

In this version, Process Analyze page Process Information header and Process Tree view have been reworked to provide a cleaner look and richer content.



Corrective Content

The following section provides the corrective content changes made for each release.

Cb Response 6.1.2

Console and Server

1. [NEW] Server now indexes filepath, modload, and process path file extensions allowing efficient searching, for example, path:.ext (CB-14167)
2. Fixed an issue where phrase queries do not work as expected when using new (non-default) command line tokenization. (CB-14171)
3. Fixed an issue where left hand-side navigation bar does not expand correctly under Chrome on OS X platforms. (CB-11840)
4. Fixed an issue where bulk operations on Triage Alerts page caused page to be unresponsive for long time. (CB-13583)
5. Fixed an issue where some UI assets/icons required external Internet access, failing to render under closed deployments. (CB-14026)
6. Fixed an issue where banning multiple hashes (more than ~10) caused banning request to fail. (CB-14149)
7. Fixed an issue with file permissions of /etc/cb directory that caused services startup to fail after an upgrade. (CB-13623)
8. Fixed an issue where /var/log/cb/audit logs did not rotate properly. (CB-8610)
9. Fixed an issue where multiple requests are sent to the server from the browser client code unnecessarily causing performance degradation. (CB-13349, CB-14164)
10. Fixed an issue where process search terms starting with “or” or “and” are incorrectly parsed as logical OR or AND causing the beginning of terms to be dropped from the query. (CB-14417)
11. Fixed an issue where editing a binary query using the new Editable Watchlist feature causes query to be incorrectly converted to a process search query. (CB-14368)
12. Selected investigation now persists over new tabs so that users can add events from Process Analyze pages opened in a new tab to chosen investigation. (CB-14408)
13. Improved e-mail UI regex rules to be more accommodating to accept e-mail addresses that have multiple periods. (CB-14457)
14. Improved core roll-over logic to resolve problems in the field that caused core roll-over to fail, resulting in disk space usage problems. (CB-13790)

Windows Sensor (6.0.3.171001.1616)

1. Added explicit binary check on files sent to Cb Response Server (CB-15357)

OS X Sensor (6.0.5.170830.1306)

1. Added explicit binary check on files sent to Cb Response Server (CB-15254)

Linux Sensor (5.2.12.170825.1225)

1. [NEW] Added support for RHEL 7.4 (CB-14245)
2. Added explicit binary check on files sent to Cb Response Server (CB-15281)
3. Added vfork syscall hook to enable more reliable collection of fork events (CB-14786)
4. Fixed issue that could lead to dereferencing a NULL pointer (CB-14878)
5. Users can now turn off cb user event exclusions locally. This allows for collection of events generated by the cb user on a Cb Response Server (CB-14686)
6. Fixed site throttling issue (CB-13963)
7. Fixed daemon crash resulting from an infinite loop condition on process exit (CB-11419)

Linux Sensor (5.2.9.170606.0910)

1. [NEW] Add paravirtualization support for Xen hypervisors. (CB-14131)
2. Improve how memory usage is reported to Cb Response server. (CB-14303, CB-14126)
3. Fixed an issue where kernel module fails to load when memory is fragmented. (CB-13986)

Cb Response 6.1.1

Console and Server

1. Fixed an issue where new license application did not work properly in a clustered environment causing minions to not to have the new license applied. (CB-13195)
2. Fixed an issue where an execution of a watchlist on a binary fails if no endpoint executed the binary or the previous process executions are purged from the database. (CB-13983)
3. Fixed an issue where some registry queries from CB Live Response session failed. (CB-13692)
4. Fixed an issue where Investigations page failed to create new investigations correctly. (CB-13597)
5. Fixed an issue where event tagging and untagging for Investigations did not function correctly. (CB-13469)

6. Fixed an issue where long filenames overlapped with other data in Triage Alerts page rows. (CB-13270)
7. Fixed an issue where an erroneous watchlist with no search query produced false positives. (CB-13581)
8. Fixed an issue where due to incorrect ordering of events of a process by endpoint local time, Process Analyze page events timeline showed no events for a process. (CB-13672)
9. Fixed an issue where server task responsible from communicating to Cb cloud services used high CPU. (CB-13604)
10. [NEW] Added Sort By functionality to new Watchlist page. (CB-13457)
11. [NEW] Sensor list and detail pages now display endpoint domain name in addition to hostname when present. (CB-13427)
12. Improved performance of Process Analyze page event timeline rendering and slider bar operation. (CB-13662, CB-13707)
13. Improved user workflow for adding new Watchlist from the Watchlist page. Added help for writing correct query syntax and validation before save. (CB-13625, CB-13556, CB-13792)
14. Corrected an error that prevented user from adding new Throttle Sites. (CB-13802)
15. Improved resiliency of in-memory config storage component to cluster nodes going offline to prevent service startup errors. (CB-13554)
16. Restore missing statistics in cbstats command line tool after upgrading to 6.*x* from previous versions. (CB-13751)
17. Corrected an issue that resulted in IP address searches producing incorrect results. (CB-13987)
18. Fixed an issue where Cb Live Response session fails if sensor group has a revoked certificate. (CB-13940)
19. Fixed an issue where a race condition causes Cb Live Response GET command to fail before completion. (CB-14010)
20. Fixed an issue where 2-letter domain names failed validation for a valid user account e-mail address. (CB-13755)
21. Fixed an issue where clicking search link on a netconn event produced HTTP 400 error when domain name is unknown. (CB-13824)
22. Fixed an issue where server ingest stops if SOLR document contains invalid data. (CB-14040)
23. Fixed an issue where restarting Cb Live Response service incorrectly updates the close time on closed sessions adversely affecting purging of old sessions. (CB-13977)

24. Corrected an issue that caused incorrect e-mail settings to be stored when multiple users try to enable e-mail action for watchlists. (CB-24289)

25. Corrected an issue that cause incorrect user id to be stored in e-mail settings for a watchlist. (CB-14256, CB-14225)

26. Corrected an issue where the UI did not correctly differentiate between cross-process events that are by the actor process versus the target process. (CB-14221)

27. Corrected an issue where searching for reports of a particular Threat Intelligence feed returned reports for all enabled feeds. (CB-14220)

28. Improved message displayed when banning previously unseen hashes from Ban Hashes page. (CB-14147)

Linux Sensor (5.2.8.170427.1025)

1. [NEW] Add CentOS/RHEL 6.9 support in 5.2-series sensor (CB-13430)

2. [NEW] Add CentOS/RHEL 7.3 support in 5.2-series sensor (CB-11511)

3. Major re-write of the Linux sensor to address high CPU, performance, stability, and accuracy (CB-12825, CB-6647, CB-12845, CB-13306, CB-11817, CB-11611, CB-11233, CB-11497, CB-12082, CB-9818)

4. Corrected an issue where MD5 hash was not calculated for all processes. (CB-12688)

5. Corrected an issue where receive UDP events were not correctly reported to the server. (CB-13815, CB-12258)

6. Fixed a possible crash when receiving UDP packets. (CB-13781)

7. Corrected an issue where sensor reported incorrect or incomplete process path information. (CB-8891)

8. Corrected several issues on install/upgrade of sensor daemon or kernel modules. (CB-12829, CB-13007, CB-132754, CB-13429, CB-11141, CB-9469, CB-10633, CB-11741, CB-11615, CB-8676)

9. Corrected an issue where proxied web network connections are not reported correctly. (CB-6669, CB-6714)

10. Corrected an issue where running process is not terminated when banned from server. (CB-12223, CB-12408)

11. Corrected an issue where banning a binary that has multiple running instances only terminates one of them. (CB-9731)

12. Corrected an issue where sensor health score reported as healthy even when the driver is not loaded. (CB-12119, CB-11964)

13. Corrected an issue where sensor stops sending event logs. (CB-11540)
14. Resolved an issue where some command line commands were reported as “-bash” instead of the full command details. (CB-6721)
15. Fixed multiple issues with CB Live Response. (CB-7512, CB-9024)
16. Added features to help Support with debugging. (CB-7576, CB-8963, CB-9277, CB-10994, CB-11487)
17. Multiple fixes to improve the performance and accuracy of network connection reporting. (CB-8830)
18. Corrected a potential interoperability issue with the CB Protection agent. (CB-11639)
19. Fixed an issue where CNAME’s were not always properly resolved. (CB-9638)
20. Added ability for sensor to report MD5 for non-binary files. (CB-9206)
21. Fixed an issue where the sensor could not communicate with the server in isolation mode if port was not specified in the URL. (CB-8238)

Cb Response 6.1.0

Console and Server

1. Resolved an issue that caused an infinite spinner while rendering Process Analyze pages. (CB-12945)
2. Fixed an issue that resulted in RabbitMQ service not stopping on cb-enterprise service stop. (CB-12922)
3. Corrected an issue where Sensor Details page did not render IP Address/MAC Address fields correctly. (CB-12903)
4. Fixed an issue where network protocol was not displayed correctly in Process Analyze page event rows. (CB-12571)
5. Fixed an issue where modifying e-mail notifications on a given feed threw an error if multiple users are subscribed. (CB-12546)
6. Fixed an issue where some timestamps were parsed incorrectly in Process Analyze page event rows. (CB-12428)
7. Fixed an issue where the invalid Analyze link URL in childproc event detail expansion redirected to splash page. (CB-12251)
8. [NEW] Improved command-line tokenization and query syntax. (CB-11740)
9. Fixed an issue where all alert notifications failed on the Cb Response server if the server is configured to connect to Cb Protection server but the connection fails. (CB-12634)

10. Fixed an issue where IP address validation failed before user completes typing. (CB-12892)
11. Fixed an issue where logging out from an error page redirected back to error page on next login. (CB-12935)
12. Corrected an issue where Sharing Settings were not saved correctly from Sensor Groups page create group dialog. (CB-12569)
13. Resolved an issue where clicking on a process icon on the process analysis tree does not correctly update Process Analysis page content to the select process. (CB-13150)
14. Corrected an issue where the description field is not returned by the API for custom feed reports, resulting missing information on the Triage Alerts page. (CB-8315)
15. Fixed an issue where the process tree display is rendered in wrong location under Chrome 57. (CB-13311)
16. Fixed an issue where the initial selection of “Add Search Term” in Process Search page results in a red toaster indicating the wrong query text. (CB-13123)
17. Fixed an issue where Alerting via E-mail page does not allow anonymous SMTP configurations. (CB-13037)
18. Fixed an issue where Alerting via E-mail page does not allow any Top-Level-Domain for a valid e-mail address. (CB-13038)
19. Corrected styling of Unresolved alert counts in Triage Alerts page. (CB-13172)
20. Corrected an issue where Threat Level and Feed facets on the Process Analyze page did not work filter results correctly.
21. Corrected an issue where some child process events with reused PIDs are omitted from the event API point. (CB-13284)
22. [NEW] Deprecated use of TLS v1 in nginx configuration for v1.1. (CB-10834)
23. Corrected an issue where cd (change directory) command in CBLR does not work for some Windows drives. (CB-11026)
24. Corrected an issue where Threat Intelligence Feeds page Create Watchlist action did not work. (CB-12009)
25. Corrected an issue where Process Preview pop-up did not render process content. (CB-13347)
26. Corrected an issue where a JavaScript exception was thrown on navigating to Binary Details page for some binaries. (CB-13455)
27. Corrected an issue where following an upgrade from 5.*x* to 6.1.0, Process Analyze page for legacy events did not render correctly. (CB-13551)
28. Corrected an issue where process transition from one to another using the Process Analyze page Process Tree resulted in an infinite spinner. (CB-13555)
29. Fixed an issue where Investigations page failed to create new investigations correctly. (CB-13597)

30. Fixed an issue where event tagging and untagging for Investigations did not function correctly. (CB-13469)

31. Fixed an issue where long filenames overlapped with other data in Triage Alerts page rows. (CB-13270)

32. [NEW] Added Sort By functionality to new Watchlist page. (CB-13457)

33. [NEW] Sensor list and detail pages now display endpoint domain name in addition to hostname when present. (CB-13427)

Windows Sensor (6.0.2.170329.1804)

1. Corrected an issue where GPO upgrade from 5.2.*x* to 6.*x* does not run. (CB-13034)

2. Corrected an issue with MSI based upgrades over 5.1.*x* and 5.2.*x*. (CB-13115)

3. Corrected an issue where uninst.exe did not remove Carbon Black product name from Add/Remove Program dialog. (CB-13163)

4. Corrected an issue where EXE installer upgrade from 5.1.*x* to 6.*x* leaves double entries in Add/Remove Program dialog. (CB-13369)

5. [NEW] Sensor installer now would check if Windows “Base Filtering Engine” is enabled and turned it on if not enabled. (CB-13333)

OS X Sensor (6.0.4.170328.1642)

1. [NEW] Configurable path exclusions per sensor group – limit event collection from a process with given path by event type to fine tune endpoint performance. (CB-13305)

2. [NEW] Filter known dylibs per sensor group – limit modload event collection of known dylibs to fine tune endpoint and server performance. (CB-13304)

3. Corrected an issue that caused daemon crash when reporting a file modification (filemod) event on a binary file larger than 2,147,483,647 bytes. (CB-13478)

4. Resolved an issue where upload of an eventlog or binary upload can get stuck consuming 100% CPU. (CB-13160)

5. Fixed an issue that caused kernel panic while waiting for a hash computation (in banning). (CB-13237)

6. Corrected an issue where sensor sent eventlogs that do not have metadata before an upgrade. (CB-11211)

Linux Sensor (5.1.4.170131.1504)

1. Update third party OpenSSL package to 1.0.1b and Curl package to 7.50.3 (CB-12165)
2. Sensor now correctly reads disk quota values from sensorsettings.ini file (CB-12562, CB-12692)

Cb Response 6.0.1

Console and Server

1. Fixed an issue where watchlist and feed hits in v6.0.0 have missing comms_ip and interface_ip fields. (CB-11312)
2. Resolved an issue where large memory allocations resulted in increased tenured-space usage in SOLR Java heap and caused Out-of-memory errors. (CB-12879)

Cb Response 5.2.5

Console and Server

1. Provide error context when users don't use full email addresses when entering SMTP configuration. (CB-11502)
2. Expose VirusTotal scores in email templates for process feed hits. (CB-11730)
3. Use correct non-default UI port in email templates. (CB-11525)
4. Fixed an issue that caused Triage Alerts page to fail rendering when an observed binary did not have digital signature information. (CB-11032)
5. Corrected an issue that caused attempts to ignore a threat report to fail when initiated from Threat Report Details page. (CB-10360)
6. Corrected an issue that caused entire query to be negated when the last group of terms in parenthesis is negated. (CB-11001)
7. Corrected an issue where selecting more than one item in Signature Status drop down menu in Add Criteria search fails with 500 error. (CB-11359)
8. Fixed an issue that caused logrotate.d postrotate script to fail for cb-rabbitmq component. (CB-11823)
9. Fixed an issue that caused logrotate.d postrotate script to fail for cb-rabbitmq in SELinux context. (CB-12042)
10. Corrected an issue where "sensors -a" command on a Cb Live Response session failed causing UI to hang indefinitely. (CB-11727)

11. Made RabbitMQ handshake timeout configurable to avoid partial services startup on environments that are slower. The new option, RabbitMQHandshakeTimeout is configurable via `cb.conf`. Default is 10000ms (results in effective timeout of 5000ms)

12. Corrected an issue where CSV export of events resulted in misaligned columns. (CB-12002)

13. Resolved an issue that resulted in process metadata with a command line that does not match the process name or path in Process Analyze page. (CB-11991)

Windows Sensor (5.2.1.161026.0747)

1. Fixed an issue where sensor provided MD5 hash of an empty string as an executable file hash to the server. (CB-11293)

2. Fixed an issue where disabling “network connections” event collection lowered the health score of the sensor. (CB-8851)

3. Fixed an issue where the sensor did not respect CarbonBlack\store\catalog file after a reboot causing increases IO load on each reboot. (CB-11509)

4. Fixed an issue where malformed values passed to sensor device driver (via malicious executables) can cause kernel panic due to device driver accessing invalid memory. (CB-8677)

OS X Sensor (5.2.5.170103.1147)

Fixes an issue where sensor installer required Xcode command lines tool. (CB-12347)

OS X Sensor (5.2.4.161216.1642)

1. [NEW] Added support for OS X 10.12.1 version (CB-11663)

2. Fix an issue where installer set incorrect permissions to /Application/Carbonblack folder. (CB-11229)

3. Disabling “Non-binary Filewrite” event collection now works correctly. (CB-6491)

4. [NEW] Propagated event collection filters to kernel driver to improve sensor performance when collection for certain events are disabled. (CB-11510)

5. Added daemon-level awareness of incomplete/failed upgrades to avoid kernel panics. (CB-12187)

6. Fixed an issue with sensor using increased memory unders stress and causing endpoint to become unresponsive. (CB-9268)

Linux Sensor (5.1.2.161109.0849)

[NEW] Added support for RHEL/CentOS 7.3 version. (CB-11511)

Cb Response 5.2.0 Patch 3

Console and Server

1. Corrected an issue where Process Analyze page failed to render event rows for some processes that had events with milliseconds apart. (CB-10376)
2. Corrected an issue that prevented users from selected multiple values for sensor facets in the Sensors page. (CB-10387)
3. Corrected an issue where a spinner on the UI won't go away until page reload following a binary download. (CB-10486)
4. Restored user's ability to download all hosts to a CSV file. (CB-10768)
5. Resolved an issue with sorting of watchlist by name. (CB-10487)
6. Removed a duplicate "Unresolved" status facet from Triage Alerts page. (CB-10635)
7. Corrected the URL for "Community Watchlists" to point to correct User Exchange link. (CB-10718)
8. Restored rendering of Feed hits metadata in event rows detail. (CB-10640)
9. Corrected an issue with deleted users still getting alert notifications. (CB-10537)
10. Corrected an issue with alerts being generated for MD5 based IOCs that were marked as false positive. (CB-10810)
11. Fixed an issue with syslog messages using the CEF format template due to incorrect escaping of some characters. (CB-10274)
12. Added sensor interface and communication IP addresses to syslog notifications for Query-based Feed hits. (CB-10536)
13. Corrected an issue where database table for pending updates to Carbon Black Alliance server was purged too aggressively causing Threat Intelligence tags on some binaries to be missed. (CB-11212)

Windows Sensor (5.2.0.160922.1638)

Added support for Windows 10 Anniversary Edition. (CB-10444)

OS X Sensor (5.2.0.161003.1756)

1. Corrected an issue with binaries downloaded from UI was malformed. (CB-10494)
2. Corrected an issue with sensoruninst.sh script failing to completely uninstall the sensor. (CB-11029)

3. Corrected an issue with sensor occasionally creating unnamed processes. (CB-8907)
4. Corrected an issue where incorrect username is reported for some processes. (CB-10457)
5. Corrected an issue with sensor service crash on shutdown due to an extra reference count decrement. (CB-10326)
6. Resolved an issue which cause kernel panic on upgrading from previous versions of 5.1.1 and 5.2.0 under some load circumstances (CB-11224)

Cb Response 5.2.0 Patch 2

Console and Server

1. Corrected an issue where services failed to start if ModstorePath setting is changed from its default value in `cb.conf` (CB-8449)
2. [New Feature] Added ability to rate-limiting of feed hit events published on the enterprise bus by feed id and IOC value (default is OFF) (CB-8535)
3. Corrected an issue where a user with no team assignment could view all processes in the Search Processes page. (CB-8799)
4. Process Analyze page “Search Term” facet now works correctly. (CB-9169)
5. Process Analyze page “Filemod” facet now works correctly. (CB-9610)
6. Process Analyze page now correctly selects IP addresses from the IP facet dropdown. (CB-8988)
7. Selected facets now move up to the top of the list in Process Analyze page. (CB-9037)
8. “Ignore future events” option now correctly applied when marking alerts as “False Positive” (CB-9241)
9. Banned hashes list is now correctly sent to sensors when banning is enabled for a sensor group. (CB-8906)
10. Event purge cron job now does not fail if module store path is mounted on a different disk volume. (CB-8737)
11. Event purge now gracefully handles the case, where a binary set to be deleted is not available on disk. (CB-9401)
12. Feed hit notification e-mails no longer have truncated file/path names. (CB-10187)
13. When logging in as a non-global admin, the login screen no longer continuously displays a spinning icon. (CB-10041)
14. Long command lines can now be copy/pasted easily from Process Analyze page header. (CB-9631)
15. FQDN with underscore is now allowed when entering server URL in sensor group settings dialog. (CB-4622)

16. The Sensor Details page now correctly shows upgrade policy settings. (CB-9232)
17. Corrected an issue where a single negated search term concatenated with Add Criteria terms fails to return results. (CB-9880)
18. [New Feature] Feed report ids are now included in the feed hit notification e-mails. (CB-9632)
19. Details of last banning attempt of a hash are now displayed correctly on the Banned Hashes page. (CB-9542)
20. An infinite spinner icon is no longer displayed on Process Analyze page if the process has no command line. (CB-10126)
21. Events can now can be added to Default investigations correctly. (CB-9487)
22. Corrected an issue where add new feed modal resulted in error on the Threat Intelligence page. (CB-9827, CB-9772)
23. Corrected an issue where event purge mechanism unnecessarily removed binary files stored under module store directory for purge metrics not tied to disk pressure. (CB-2789)
24. Users that are associated with existing investigations or banned hashes now can be deleted from the system without error (historical context/association continued to be maintained.) (CB-9317)
25. Improved Sensor Details page workflow – added pagination and ability to configure number of row displayed for a given set of search terms or facet selections (CB-9202, CB-9203, CB-9250, CB-9255, CB-9257)
26. Server now uniquely identifies alerts generated on ingress feed hits (for example, from feed hit events feed.hit.ingress.process and feed.hit.ingress.binary) from alerts generated by the feed_searcher cron job nightly runs. While the former alerts only on new process executions or binary reports that match a given feed report, the latter also creates an alert when there is a change in the feed report content or score since the last time a process or binary was tagged. Alerts generated from feed_searcher cron job now have specific alert type that refers to “feedsearch” in name and are displayed in Triage Alerts page with yellow color instead of red for easy visual differentiation. (CB-9393)
27. New Triage Alerts page workflow now displays IOC value for IPv6, MD5, and domain under the source column in addition to feed report name. (CB-9627)
28. Corrected an issue where UTF-8 characters in process events caused exceptions in the SOLR datastore negatively impacting data ingest (CB-10424)
29. Corrected an issue where clicking on the hyperlink for IP address on a netconn event produced an error (CB-10403)
30. Corrected an issue where hyperlinks to Process Analyze page from process alerts were invalid (CB-10592)
31. Corrected an issue where Process Analyze page failed to render events because some event timestamps were missing in the API response (CB-10097)

Windows Sensor (5.2.0.160824.0930)

1. Improved sensor operation efficiency to reduce overhead during boot time (CB-8232, CB-9748)
2. Sensor no longer trigger AV alert during EICAR test signature in code (CB-9396)
3. [NEW] Added support for Windows 10 Anniversary Edition.

OS X Sensor (5.2.0.160721.1909)

1. Corrected an issue where sensor service was not stopping correctly causing problems during uninstall/shutdown. (CB-10127)
2. Sensor now correctly reports module load of an executable image. (CB-8491)
3. Sensor now correctly computes MD5 with lastWrite filemod events. (CB-9207)
4. Sensor now correctly synchronizes banning status with server on startup. (CB-9190)
5. Sensor now correctly reports lastWrite filemod events. (CB-9140)
6. Corrected an issue where invalid/truncated MD5 hashes were reported for binaries (CB-10095)

Cb Response 5.2.0 Patch 1

Console and Server

1. Corrected an issue where setting `cb.conf` option SensorLookupInactiveDays=X for limiting Sensor Details page view to sensors that have been active in the past X days fails with exceptions in coreservices debug logs. (CB-9593)
2. Corrected an issue where parsing of malformed filemod events causes exceptions in datastore, leading to poor data ingest performance. (CB-9514)
3. Corrected an issue where resolving multiple alerts as “false positive” failed with exceptions. (CB-9491)
4. Corrected an issue where Feed Reports in Threat Intelligence page failed to render if they contained IOCs that referred to binary documents. (CB-9422)
5. Corrected an issue where expanding child process terminate event rows in Process Analyze page UI showed incorrect information. (CB-9403)
6. Corrected an issue where process or binary path metadata for alerts created from query based feeds were truncated. (CB-9185)
7. Corrected a text box overrun in Ban Hashes” page. (CB-9173)
8. Improved service bus topology around watchlist/feed hit events to reduce traffic when those events are not of interest to anyone. (CB-8536)

Windows Sensor (5.2.0.160603.1453)

1. Corrected an issue where storefile disk quota (for storing binary files on sensor) may be exceeded. (CB-8447)
2. Corrected a rare bugcheck that occurred in the cbk7.sys sensor driver. (CB-9328)
3. Corrected a potential issue where sensor service caused divide-by-zero exception. (CB-6839)

OS X Sensor (5.2.0.160603.1436)

1. Fixed an issue where sensor install pkg file triggered an AV alert with EICAR test signature. (CB-9392)
2. Corrected an issue where sensor allowed banning of its own service. (CB-9397)
3. Corrected an issue where already running process failed to terminate if it ignored SIGTERM signal. (CB-9403)
4. Corrected an issue where suppressed child process reported within the parent process context did not have an unique process identifier. (CB-9316)
5. Corrected an issue where some child process terminate events were missed. (CB-9233)
6. Sensor now correctly reports CNAMEs in network connection events. (CB-9549)
7. Child process terminated events now correctly report the timestamp of end event, rather than the start event. (CB-9357)
8. Corrected an issue where remote commands executed via Cb Live Response left behind zombie processes. (CB-9193)
9. Corrected an issue where force umount on a directory currently being used caused a kernel panic. (CB-9244)

Linux Sensor (5.2.0.160603.1441)

1. Added support for RHEL/CentOS 6.8 version on the endpoint. (CB-9253)
2. Corrected an issue where remote commands executed via Cb Live Response left behind zombie processes. (CB-9236)

Cb Response 5.2.0

Console and Server

1. Corrected an issue where CSV export of hosts that observed a binary in Binary Detail page failed to work if Search Processes page facets were disabled from `cb.conf`. (CB-4073)

2. Corrected an issue where count of hosts displayed on Dashboard page did not correlate with the value displayed on the Sensor Details page. (CB-4042)
3. Corrected an issue where bulk resolve of more than 1000 alerts did not resolve all alerts on the Triage Alerts page. (CB-4031)
4. Fixed an issue where search links in Watchlist page failed if the search term for the watchlist contained forward slashes. (CB-7275)
5. Corrected an issue where Cb Live Response registry query command failed to return results for registry hives with spaces in them. (CB-3730)
6. Corrected an issue where nightly cron job for tagging documents that match newly added feed reports failed with a KeyError. (CB-7472)
7. Corrected an issue where the searches for time based process document fields showed incorrect syntax under “Showing Results for…” link on the UI. (CB.7724)
8. Fixed an issue where startup script for setting SELinux security context on a NFS share causing startup failures. (CB-3765)
9. Corrected an issue where feed tags associated with a process event erroneously deleted when process document was split into multiple files in the SOLR database. (CB-8346)
10. Fixed an issue where failure to download a file from Carbon Black Alliance Server using cbget when requested file did not exist erroneously reported connectivity to Alliance Server status on the UI as disconnected. (CB-8423)
11. Threat Report create time based searches from the UI now correctly works. (CB-8613)
12. CSV export of events from all search pages are now generated on the server side for robustness. (CB-2826)
13. Triage Alert page is redesigned for cleaner workflow and faster load times. (CB-7548)
14. Sensors page is redesigned for faster load times and ability to page list of sensors within a sensor group for cleaner workflow. (CB-8788)
15. Searches for command lines now correctly works for search terms that contain single quotes. (CB-2807)
16. Process Analyze page preview now correctly renders if process is missing process name or path. (CB-4956)
17. Notes are now retained correctly if a hash is unbanned. (CB-5113)
18. Resolved inconsistency in the Action button functionality on sensor detail and sensor list pages. (CB-5130)
19. Improved Watchlist Name edit functionality. (CB-4913)

20. Added a visual cue for facets selected when no search results are return to improve workflow. (CB-5189)
21. Directory (path) facet on Process Analyze page now correctly displays terms for Linux sensors. (CB-4642)
22. Now sharing settings can be configured while creating a new sensor group. (CB-5471)
23. Corrected an issue where default values for various settings on the sensor group dialog were not reflected correctly. (CB-7529)
24. Watchlist page sidebar now correctly persists sort order after item selection. (CB-7277)
25. “E-mail Me on Hit” option from Watchlist page now works correctly. (CB-7388)
26. Threat Reports page no longer erroneously display deleted reports for manually added feeds. (CB-7416)
27. Watchlist page tooltips now correctly disappear when cursor is moved away from the selection. (CB-7532)
28. Corrected an issue where the Sensors page did not load correctly when a user with access rights to a customer group did not have permissions to default sensor group. (CB-8320)
29. Sensor Group Settings dialog now correctly handles team names with longer than 23 characters. (CB-7629)
30. Tooltips that contain quotes are now handled correctly in Triage Alerts page tooltips. (CB-7679)
31. Confirmation dialog for network isolation now more accurately inform users on the actions/limitations of this feature. (CB-8228)
32. Binary Search page UI now correctly allows wildcard searches in filename field. (CB-7735)
33. SMTP server names that contain hyphen now can be correctly entered in e-mail settings. (CB-8330)
34. Server UI client application now is prevented from running inside another frame. (CB-8432)
35. Process Analyze page now correctly removes spinner when page is rendered. (CB-8886)
36. Watchlist page correctly displays the “last hit” time when there are positive hits to the query. (CB-8957)
37. Corrected an issue where some feed tags were erroneously removed from the process instances when such event data from such processes were split over multiple SOLR documents. (CB-8346)

Windows Sensor (5.2.0.160518.1524)

1. Fixed an issue that caused system crash if the sensor was running on a VM that was going through live migration. (CB-7158)
2. GPO installer now have correct product version. (CB-6953)
3. Sensor core driver can cause system crash if installation fails for any reason. (CB-6929)
4. Sensor can associate wrong parent information to processes which it did not see start (sensor was installed on a running system.) (CB-6911)
5. Sensor can associate wrong start up context to processes which it did not see start (sensor was installed on a running system.) (CB-6873)
6. Sensor service can leak memory on system that are under heavy load (seeing high volume of process execution and termination events.) (CB-7065)
7. Sensor cbstream driver can cause softlock on boot or shutdown on Google Cloud Platform. (CB-6977)
8. Corrected an issue where MSI installer failed re-installation. (CB-7372)
9. Sensor stealth mode installation fails if sensor process name provided does not have .exe extension. (CB-7609)
10. Sensor service may cause network shares to disconnect or otherwise fail when accessing files (CB-7764)
11. Sensor uninstall from web UI fails if sensor name is changed under stealth mode. (CB-8291)
12. Corrected an issue where sensor cbtdiflt driver cause system crash when accessing buffers in chained receive handlers. (CB-8245)
13. Fixed an issue where DNS cache in sensor service was not being populated correctly. (CB-8407)
14. Fixed an issue where cbtdiflt driver was causing system crash due to access to pointers without checking their validity (CB-7718)
15. [New Feature] Sensor now implements suppression of eventless (uninteresting) processes. (CB-7266)
16. [New Feature] Sensor now suppresses known DLLs in Windows process executions when enabled per sensor group. (CB-7294)

Linux Sensor (5.2.0.160518.1322)

1. Fixed an issue where network connection events were associated with incorrect parent process under load. (CB-8214, CB-8485)

2. Fixed an issue where network connection events did not have process path in the raw protobuf events similar to Windows platform, impacting monitoring of raw events from the enterprise event bus. (CB-9045)
3. Cb Live Response on Linux sensor now correctly accesses directories with apostrophes in their name. (CB-9024)
4. Corrected an issue that caused system crash when sensor is put in isolation mode. (CB-8236)
5. Corrected an issue where some process events were missing process PID information. (CB-8748)
6. Corrected an issue where cbdaemon initialization script referred to a directory that no longer exists. (CB-8467)
7. Sensor no longer reports username after user context event collection option is disabled. (CB-8419)
8. Sensor now correctly updates sensorsettings.ini file values received from the server. (CB-8424)
9. Corrected an issue where sensor driver sporadically crashed sending health alert level 75 (driver failure) to the server. (CB-6700)'
10. [New Feature] Sensor now implements suppression of eventless (uninteresting) processes. (CB-7266)
11. [New Feature] Sensor now differentiate between process forks and other executions. (CB-6756)
12. Fixed an issue where putting sensor in network isolation caused it to go offline. (CB-9295)
13. Corrected an issue where installer placed an unexpected file under /opt/cbsensor following install. (CB-9079)

OS X Sensor (5.2.0.160518.1339)

1. OS X sensor now correctly updates sensorsettings.ini file values received from the server. (CB-6463)
2. OS X sensor sensordiag.sh diagnostic script now does not collect log and diagnostic directories that are not pertaining to its operation when packaging diagnostic information. (CB-7785)
3. Corrected an issue in PSC_fork call in OS X sensor causing a kernel panic in process tracking. (CB-6476)
4. Corrected an issue in parsing of DNS packets that caused high CPU usage. (CB-8394)
5. Corrected an issue that caused up to 6 seconds delay in starting applications on a sensor that did not yet check-in with the server. (CB-8885)
6. Sensor no longer reports its own events from CbOsxSensorService. C(B-8840)

7. Corrected an issue where exceptions in protobuf library causing sensor daemon to crash randomly. (CB-6486)
8. [New Feature] Sensor now implements suppression of eventless (uninteresting) processes. (CB-7266)
9. [New Feature] Sensor now differentiate between process forks and other executions. (CB-6564)
10. Added process md5 to child process execution event message protobuf headers. This is useful when parsing raw events on the enterprise message bus for third party analysis. (CB-9446)

Carbon Black Enterprise Server 5.1.1 Patch 4

Console and Server

1. Added sensor interface and communication IP addresses to syslog notifications for Query-based Feed hits. (CB-10536)
2. Fixed an issue with syslog messages using the CEF format template due to incorrect escaping of some characters. (CB-10274)

Windows Sensor (5.1.1.160913.1023)

1. Added support for Windows 10 Anniversary edition. (CB-10591)
2. Fixed an issue where network driver for legacy Windows XP/Windows 2K support caused system crash due to incorrect handling of buffers. (CB-10964)
3. Fixed an issue with third party application conflict with “Imaging for Windows 4.0 by Global360”. (CB-10963)
4. Addressed an issue with excessive boot time when sensor is installed on endpoints with several other security and management tools that all run on startup. (CB-10990)

OS X Sensor (5.1.1.160915.1527)

1. Added support for OS X 10.12 Sierra version. (CB-10540)
2. Removed warnings from logging when getting sensor version from CLI. (CB-10984)
3. Addressed an issue where a memory leak in CreateVnodePath caused memory to be exhausted causing kernel panic. (CB-10959)
4. Optimized sensor efficiency when computing eventlog queue quota sizes. (CB-10962)
5. Fixed an issue where sensor became unresponsive and lost network connectivity while under heavy load. (CB-10957)

Linux Sensor (5.1.1.160913.1004)

1. Fixed an issue with due to malformed ZIP causing binary files downloaded from UI to be corrupt. (CB-10548)
2. Fixed an issue with accessing files on an NFS share that caused system crash. (CB-10993)
3. Fixed an issue where binary file store location under /var/lib/cb/store grew past configured limits. (CB-10992)

Cb Response 5.1.1 Patch 3

Console and Server

1. Changed requests from datastore to use POST method rather than GEt when querying for feed reports so that long report ids can be accommodated without hitting URL limits. (CB-9635)
2. Improve ingress matching for domain name based IOCs to matching on subdomains in addition to the FQDNs, for example, an IOC domain *example.com* would now match both a network connection to *a.example.com* and *b.example.com.* (CB-7478)
3. UI now correctly honors use_proxy and validate_server_cert options correctly when adding a custom feed. (CB-9649)
4. Server now uniquely identifies alerts generated on ingress feed hits (for example, from feed hit events feed.hit.ingress.process and feed.hit.ingress.binary) from alerts generated by the feed_searcher cron job nightly runs. While the former alerts only on new process executions or binary reports that match a given feed report, the latter also creates an alert when there is a change in the feed report content or score since the last time a process or binary was tagged. Alerts generated from feed_searcher cron job now have specific alert type that refers to “feedsearch” in name and are displayed in Triage Alerts page with yellow exclamation marks instead of red for easy visual differentiation (CB-9393)
5. UI now asynchronously requests facet data on all search requests instead of only when visiting a search page the first time, reducing the time it takes to load them. (CB-9797)

Windows Sensor (5.1.1.160603.1529)

1. Fixed an issue where sensor service’s attempt to access files on network shares as SYSTEM was causing problems with various DFS shares, ranging from corrupted file writes to disconnected share drives. (CB-7764)
2. Corrected a potential issue where sensor service caused divide-by-zero exception. (CB-6839)
3. Corrected an issue where legacy TDI filter driver was accessing pointers without checking if they are valid. (CB-7718)
4. Corrected a slow memory leak that occurred when sensor service is under heavy load. (CB-7065)

5. Corrected a rare bugcheck that occurred in the cbk7.sys sensor driver. (CB-9328)

OS X Sensor (5.1.1.160603.1506)

1. Child process terminated events now correctly report the timestamp of end event, rather than the start event. (CB-9357)
2. Improved DNS parsing code to avoid high CPU usage. (CB-8394)
3. Sensor service no longer reports itself and its child processes to the server. (CB-7534)
4. Fixed an issue that caused sensor service crash in DNS parsing library. (CB-8798)
5. Corrected an issue where remote commands executed via Cb Live Response left behind zombie processes. (CB-9193)
6. Corrected an issue where force umount on a directory currently being used casued a kernel panic. (CB-9244)
7. Sensor now correctly reports CNAMEs in network connection events. (CB-9549)

Linux Sensor (5.1.1.160603.1515)

1. Corrected an issue where sensor service failed to start on reboot following an upgrade. (CB-8864)
2. Netconn events now contain process path as part of the protobuf message headers like in Windows platform. This is useful when parsing raw events on enterprise message bus for 3rd party analysis. (CB-9045)
3. Corrected a slow memory leak that caused elevated memory usage over long periods of time. (CB-9134)
4. Corrected an issue where remote commands executed via Cb Live Response left behind zombie processes. (CB-9236)
5. Added support for RHEL/CentOS 6.8 version on the endpoint. (CB-9253)

Cb Response 5.1.1 Patch 2

Console and Server

1. Corrected an issue where binary file store synchronization cron job was inserting incorrect MD5 hash values into PostgreSQL and therefore was never synchronizing correctly with files stored on disk. (CB-8750)
2. Added ability to broadcast raw sensor eventlogs to api.rawsensordata RabbitMQ exchange. (CB-7330)
3. Incoming network connection events that are tagged as feed hits now correctly shows up as feed hits in the Process Analyze page. (CB-7513)

4. CB Tamper feed hits are now correctly shown in Process Analyze page. (CB-8826)

5. Corrected an issue where Alliance feed hit tags were not correctly copied over when SOLR documents for long-lived processes split into multiple segments causing hit information to be lost. (CB-8346)

Windows Sensor (5.1.1.160415.1734)

1. Corrected an issue where eventlogs were sent to the wrong minion in clustered environment when a CB Live Response session was initiated. (CB-8486)

2. Corrected an issue where last eventlogs were not written to disk upon power off of endpoint causing some events occurring right before shutdown event to be lost. (CB-8420)

3. Fixed an issue that caused core driver to bugcheck in error path during initialization. (CB-8903)

OS X Sensor (5.1.1.160415.1724)

1. Fixed a memory corruption in network connection tracking that caused a crash. (CB-8785)

2. Fixed a memory leak in sensor user space service code. (CB-8740)

3. Fixed a memory leak in sensor kernel extension code. (CB-8802)

4. Fixed a sensor crash due to a failure to map a file to memory. (CB-8410)

5. Added process path and process MD5 to the header of network connection eventlogs uploaded by sensor. This is useful if raw sensor events are broadcast on RabbitMQ bus for archiving or further analysis. (CB-8924)

6. Fixed a spelling mistake in sensor uninstaller output. (CB-8720)

Linux Sensor (5.1.1.160415.1732)

1. Addressed memory leak in cbdaemon on RHEL 7.1/CentOS 6.7 (CB-8444)

2. Added process path and process MD5 to the header of network connection eventlogs uploaded by sensor. This is useful if raw sensor events are broadcast on RabbitMQ bus for archiving or further analysis. (CB-8924)

3. Added event timestamp to the header of process start eventlogs uploaded by sensor. This is useful if raw sensor events are broadcast on RabbitMQ bus for archiving or further analysis. (CB-8551)

4. Fixed an issue that resulted sensor driver to fail after install. (CB-8313)

5. Fixed a kernel panic that was result of a NULL pointer being dereferenced in kernel space. (CB-8754)

Cb Response 5.1.1 Patch 1

Console and Server

1. Corrected an issue where query of feed reports into memory for ingress matching could take a long time and cause data ingest to stop due to small default database paging size of 100. Paging size is now configurable via `cb.conf` (CB-7487, CB-8287)
2. Corrected an issue with cbinit script failing to create “cb” service user when it is ran as a non-root user. (CB-7545)
3. Corrected an issue with cbinit script failing to locate iptables if it is not in the running user’s PATH variable. (CB-7622)
4. Corrected an issue where cb-enterprise daemon does not successfully re-connect to RabbitMQ message bus if RabbitMQ socket temporarily goes down. (CB-8216)
5. cb-solr service throws UnknownHostException on feed hits if server hostname can’t be resolved causing feed hits to not to be reported. (CB-8218)
6. Corrected an issue where failure to download a file using the command line utility cbget causes Carbon Black Alliance Server communication status to show failure, even though server communication is intact. (CB-8423)
7. If a non-root user has been added to cluster.conf during *cbcluster add-node*, changes to this user in cluster.conf are not reflected in subsequent ssh communication with minions causing other *cbcluster* commands to fail. (CB--7571)
8. Corrected sensorsettings.ini file values for eventlog disk quota percentage and absolute size which were inadvertently reversed. (CB-8387)
9. Corrected an issue with CB API usage where passing an empty string as a sort parameter into a query API caused search to fail. (CB-7351)
10. Corrected an issue with *binary metadata* index purge script command line parsing that caused -g option to not to be honored when in dry-run mode. (CB-4578)
11. Corrected an issue with CBLR execfg command incorrectly parsing its arguments. (CB-7779)
12. Corrected an issue with UI dialog for ignoring future alerts from a feed not appearing when alerts are resolved as false positive. (CB-7640)
13. Corrected tooltips that were not correctly escaped for binary hashes banned from the UI. (CB-8380)
14. Corrected incorrect sizing of process icons in Search Processes page. (CB-7768)
15. Corrected incorrect reference to documentation in VDI sensor group settings. (CB-7547)

16. Modified the feature to filter out sensors that are dormant or inactive. Instead of pruning them from the database, they are now filter at the API level to preserve the historical context of process activity stored by the server. The configuration option in `cb.conf` has also been modified to reflect the change in implementation (see section under server upgrade topic.) (CB-4096)

Windows Sensor (5.1.1.160314.0129)

1. Fixed an issue with sensor service frequency computation that caused intermittent “divide-by-zero” errors that resulted in system crash. (CB-8533)
2. Corrected a memory leak in core driver that only occurred if all event collections were disabled. (CB-6969)
3. Corrected an issue in sensor TDI driver (for Windows XP and Windows server 2003) that caused a bug check by accessing pointers without checking if they were valid. (CB-8520)
4. Corrected an issue in sensor TDI driver that caused a bug check due to incorrect handling of chained receive buffers. (CB-8521)
5. Corrected an issue where sensor missed process events generated close to endpoint shutdown due to a missing flush to disk in shutdown path. (CB-8524)
6. Corrected an issue where sensor uninstall from the UI failed when service name has been changed for obfuscation. (CB-8519)

OS X Sensor (5.1.1.160314.0122)

1. Fixed an issue with excessive memory usage on CbOsxSensorService due to incorrect tracking of some processes where sensor did not see the process start (for example, because service was restarted after). (CB-8230)
2. Fixed an issue with excessive debug messages printed to /var/log/system.log by the sensor. (OS-8227)
3. Fixed a kernel panic under 10.11.2 due to changes to underlying OS kernel structures. (CB-7408)
4. Fixed an issue where some of the child process terminate messages were not reported to the server. (OS-8487)

Linux Sensor (5.1.1.160314.0136)

1. Fixed an issue with excessive memory usage on cbdaemon due to incorrect tracking of some processes where sensor did not see the process start (for example, because service was restarted after). (CB-8314)
2. Corrected an issue where cbdaemon stopped working after some time and a status check on it returned “cbdaemon is dead but subsys locked”. (CB-8371)

3. Fixed an incorrect reference to a directory path during cbdaemon initialization script. (CB-8386)
4. Fixed an issue that caused sensor to post CBLR commands incorrectly. (CB-7510)
5. Corrected an issue that caused sensor to hang under heavy system load. (CB-6650)

Carbon Black Response 5.1.1

Console and Server

1. Improved logging in feed synchronizer background task. (CB-3932)
2. Corrected an issue with sensor uninstall from the UI when user does not have global administrator privileges. (CB-4006)
3. Resolved a race condition between SQL purge maintenance task and Alliance Server binary uploads. (CB-4008)
4. Updated nginx cb-multihome.conf.example to match the nginx `cb.conf` that is shipping in 5.1. (CB-4012)
5. Fixed incorrect time stamps on sensor communication failures. (CB-4014)
6. Corrected misleading `cb.conf` content. (CB-4016)
7. Resolved emails not being sent for host-based Tamper Detection events issue. (CB-4020)
8. Fixed failures in moduleinfo_insert statements because of an integer overflow in primary 'id' sequence on the SQL table. (CB-4028)
9. Fixed an issue with bulk resolve of alerts due to a logic error in API calls. (CB-4035)
10. Fixed an issue with alerts from OSX/Linux 4.*x* sensors that resulted in invalid process links. (CB-4037)
11. Fixed an issue with redundant syslog events from feed searcher job every time a MD5 matches a feed. (CB-4045)
12. Corrected invalid report id errors from watchlist searcher. (CB-4048)
13. Fixed an issue persistence of global feed alert settings on the UI across multiple users. (CB-4058)
14. Corrected an issue with total blocks counter not being updated for banned hashes. (CB-4059)
15. Corrected an issue with ignore status on feed report being nullified on feed full-sync. (CB-4062)
16. Corrected an issue with cbcluster start hangs while CbTools continues to run. (CB-4065)
17. Corrected an issue with disabling "Process user context" event collection not being reflected in the systemsettings.ini file. (CB-4066)

18. Removed sensor purge functionality in favor of filtering Sensor Details page results in the API. (CB-4069)
19. Added parent_unique_id field to the results returned by the rest API search() endpoint. (CB-4074)
20. Fixed an issue with Process Analyze page feed facets. (CBUI-1036)
21. Corrected a discrepancy in sensor queue values reported by UI versus the rest API. (CBUI-1130)
22. Corrected “Email Me” option not being persisted after watchlist creation issue. (CBUI-1532)
23. Corrected an issue with "Export All to CSV" action on Sensors page failing to export all sensors. (CBUI-1575)
24. Improved how drag and drop on "team settings" UI page works. (CBUI-1576)
25. Search Binaries page now correctly displays "ago" in the first-seen field on result rows. (CBUI-1578)
26. Corrected an issue with incorrect search being performed when clicking on “Publisher” field in Process Analyze page. (CBUI-1582)
27. Corrected an issue on selection of a facet for process analysis. (CBUI-1600)
28. Corrected an issue with “sensor filter by node” facet, which resulted in incorrect selections on the Sensors page. (CBUI-1601)
29. Fixed an issue with hyperlinks on UI notifications drop down. (CBUI-1602)
30. Improved sensor "yield" tooltip messaging when the issue is health score related. (CBUI-1612)
31. Corrected an issue with custom threat feed dialog not correctly disappearing after adding a feed url manually. (CBUI-1686)

Windows Sensor (5.1.1.151030.0948)

1. Fixed an issue with kernelSocketConnect in cbk7.sys that resulted in system crash in some machines. (WIN-306)
2. Fixed a potential memory leak in cbtdiflt close completion handling. (WIN-340)
3. Fixed an issue with sensors not honoring "collect binaries" checkbox in sensor group settings. (WIN-349)
4. Fixed an issue with sensor dropping network connections on Win 2K3 endpoints. (WIN-352)
5. Resolved an issue that resulted in sensors not communicating to server on isolate. (WIN-360)
6. Resolved a potential deadlock due to holding FAST_MUTEX while calling ZwSetValueKey().

(WIN-362)

OS X Sensor (5.1.1.151217.0244)

1. Sensor now correctly rotates/expunges log files so that /var partition is not filled. (OSX-251)
2. Reduced excess error events in system.log with OS X 10.11. (OSX-281)

Linux Sensor (5.1.1.151215.1153)

1. Sensors now correctly rotate/expunge log files so that /var partition is not filled. (LNX-194)
2. Sensor now correctly honors Binary/Eventlog collection limits (1GB or 2% each) with small partitions. (LNX-196)
3. Fixed a kernel panic on systems running *named* linux service. (LNX-206)

Known Issues and Limitations

OS X Sensor Upgrade Limitation

Customers who have previously upgraded to OS X version 5.2.8.170419.1312 won't be able to upgrade to OS X version 6.0.4.170328.1642 included in this package due to an installer issue that does not correctly allow for upgrade to a higher build version if the build timestamp is not newer. This does not impact any earlier OS X version built before March 28th, 2017.

OS X 10.12 Sierra Support with 5.2.0 Patch 3 and Later Sensors

If you have *already* upgraded to OS X 10.12 while running 5.2.0 Patch 2 or earlier versions of the sensor, the sensor will continue to operate, however certain events may not be reported as expected (for example, module loads) or some features might be unavailable (such as banning).

At this point, if the sensor is upgraded to 5.2.0 Patch 3 or later sensors, a reboot will be necessary to restore full functionality.

If 5.2.0 Patch 3 or a later sensor is installed *before* upgrading to OS X 10.12 or a fresh install of 5.2.0 Patch 3 or a later sensor on 10.12 Sierra **will not** require a reboot to begin functioning fully.

Changes to nginx Configuration Directory

Customers upgrading to 5.2.0 from earlier versions will find that nginx proxy configuration directory (`/etc/cb/nginx/conf.d`) layout has changed in this version. Custom nginx server configuration that is contained in `cb.server.custom` file is now located under `/etc/cb/nginx/conf.d/includes`. Customers may need to edit their nginx `cb.conf` file to update the include path of this file to reflect the new directory hierarchy following the upgrade.

For additional troubleshooting information and configuration examples, see the following knowledgebase articles:

https://community.carbonblack.com/docs/DOC-5430

https://community.carbonblack.com/docs/DOC-5441

Installations Using Single Sign-On

Customers upgrading to 5.1.1 Patch 2 from earlier releases may need to edit their SSO configuration file to ensure proper operation after upgrading. The following steps should be taken:

1. Verify the name of the current sso configuration file being used. This is defined in `/etc/cb/cb.conf` with the `SSOConfig` parameter, for example: `SSOConfig=/etc/cb/sso/sso.conf`

2. In the sso configuration file, find the entry for the assertion_consumer_service. It will look similar to the following:

```
"endpoints": {
          "assertion_consumer_service": [
            [
              "https://<IP Address>/api/saml/assertion",
             "urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
            ]
          ]
  },
```

3. If the assertion_conumer_service is defined using square-bracket syntax as in the example above, change it to use curly-brace and replace the comma to a colon in its syntax, as follows:

```
      "endpoints": {
      "assertion_consumer_service": {
            "https://<IP Address>/api/saml/assertion":
           "urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
       }
},
```

Using Boolean OR with Negated Query Terms

Cb Response server query language relies on the query syntax of the underlying database architecture that uses SOLR/Lucene. This query syntax has limitations when dealing with negated terms in queries that contains Boolean OR, for example, A OR -B.

In such cases, negated term is OR'ed with the result set of the terms that are not negated, instead of being applied first over the entire document set and then OR'ed with the result set of the other terms. This may return confusing search results, for example:

```
netconn_count:[20 TO *] OR -process_name:chrome.exe
```

This query is expected to return processes that have more than 20 network connections OR processes not named *chrome.exe*, regardless of their network connection count. However, the results set will be a set of processes that are not named *chrome.exe* in the set of processes that have more than 20 network connections.

In order to work around this shortcoming, the logical OR could be translated into a logical AND by using the equivalent negated version of the entire query, for example, A OR -B → -(-A AND B)

```
-(-netconn_count:[20 TO *] AND process_name:chrome.exe)
```

Alternatively, the negated term can be replaced with a term that includes logical AND to a term that would match all documents, for example:

```
netconn_count:[20 TO *] OR (process_id:* AND -process_name:chrome.exe)
```

A comprehensive fix to this limitation will be included in an upcoming release.

Tracking and Isolation of Network Connections That Existed Before the OS X Sensor Was Installed

In the OS X sensor version included in 5.1.1 Patch 2, we have made a design change to improve sensor interoperability with a number of other endpoint applications, for example, Symantec Endpoint Protection agent and LittleSnitch. This resulted in a modified behavior in tracking and isolation of network connections. In 5.1.1 Patch 2, network connections and sockets that are established *before* the sensor is installed will not be tracked for monitoring and isolation. If the machine is rebooted after installation, the sensor will continue to monitor and successfully isolate all network connections.

Automatic Pruning of Inactive Sensors

In version 5.1.0 Patch 1, we have added configuration logic to prune out sensors that are dormant or inactive. This would include systems that are offline, uninstalled or otherwise not communicating with the Cb Response server for a given number of days. The following configuration has been added to the `cb.conf` file to control pruning of such inactive sensors:

```
DeleteInactiveSensors=True

DeleteInactiveSensorsDays=10
```

By default, the value is set to *False*.

In 5.1.1 Patch 1, we modified the configuration to filter out sensors that are dormant or inactive, rather than pruning them from the database to preserve the historical context of process activity stored by the server. The configuration option in `cb.conf` has also been modified to reflect the change in implementation:

```
SensorLookupInactiveFilterDays
```

If this value is unset (default), all sensors are returned. When SensorLookupInactiveFilterDays set to > 0, only sensors that checked in the past SensorLookupInactiveFilterDays days will be returned.

Important Note:

Users upgrading to 5.1.1 Patch 1 or Patch 2 from earlier releases may need to update their `cb.conf` file to reflect this change. ***The new setting supersedes both previous settings and the legacy settings are ignored by the system****.*

Other Issues

1. Cbssl command line throws a KeyError exception when run on the server, even though its execution correctly completes (CB-12622)
2. OS X and Linux sensors do not support excluding certain hashes from being banned via restrictions.conf. This feature is only supported for Windows platform.

3. Version 5.1.0 implementation of sensor purging has a known issue. If a sensor has been purged prior to its process data being purged, the Process Analyze page will return a 404 error for that sensors processes. All searching capabilities and process events are still present, searchable, and will be alerted. To reduce the chances of this scenario if you choose to enable DeleteInactiveSensors, we recommend setting your DeleteInactiveSensorsDays equal to or greater than your desired storage retention period. *This issue has been addressed in 5.1.1 Patch 1*
4. Negated terms in queries with Boolean OR logic have some limitations (see section under upgrading the server). (CB-4068)
5. In order for sensor upgrades to work properly, McAfee EPO may need to be configured to exclude c:\windows\carbonblack\cb.exe from its "Prevent creation of new executable files in the Windows folder" option. (CB-7061)
6. The power state of a Linux sensor is not displayed correctly on the Host Details page. When a Linux sensor is powered off, the icon next to the Computer Name does not change to the correct state. (CB-6671)
7. Some outbound UDP network connections are not reported on Linux platforms. (CB-6630)
8. ICMP traffic is allowed when sensor is isolated on Linux and OS X platforms. (CB-6483/CB-6623)
9. Non-binary file write event collection cannot be disabled on Linux platforms. (CB-6686)
10. On OS X platforms, the UI setting to turn all “event collections” off is not honored. (CB-6389)
11. Binary execution of a file can still be banned if the file reuses the same inode on Linux and OS X platforms. (CB-6647/CB-6402)
12. If a sensor’s system clock is wrong and in the future, the start time for processes from that sensor are not displayed correctly in the Carbon Black console. (CB-6257)
13. On the Carbon Black server, when a sensor is moved out of a group with a user on a team that has only "Viewer" access to that particular group, results for that group are still searchable for the time period it was in that group, but the Process Details page links get 405 errors. If the sensor is put back into the group, the 405 errors for those processes go away. (CB-3704)
14. The Reshard tool can fail with “File Not Found” exception, in turn causing a corrupt index. If a re-shard is necessary please contact support for a potential work around. (CB-3743)
15. The Linux sensor fails to properly cache observed events after the disk quota is reached and connection to the server is lost. (CB-6722)
16. The Linux sensor may fail to generate an MD5 and collect a binary image of file on a network share or user-space file system. (CB-6749)
17. CbEP enforcement fails after the Linux Sensor is uninstalled. A restart of CbEP is required to restore enforcement. (CB-7674)

Contacting Carbon Black Support

Carbon Black Technical Support provides the following channels for resolving support questions:

Technical Support Contact Options
Web: www.carbonblack.com
Email: support@carbonblack.com
Phone: 877.248.9098 (877.BIT9.098)
Fax: 617.393.7499
Hours: 8:00 a.m. to 8:00 p.m. EST

Reporting Problems

When contacting Carbon Black Technical Support, be sure to provide the following information:

Required Information	Description
Contact	Your name, company name, telephone number, and e-mail address
Product version	Product name (Cb Response server and sensor version)
Hardware configuration	Hardware configuration of the Cb Response server (processor, memory, and RAM)
Document version	For documentation issues, specify the version of the manual you are using. The date and version of the document appear after the copyright section of each manual.
Problem	Action causing the problem, error message returned, and event log output (as appropriate)
Problem severity	Critical, serious, minor, or enhancement