Carbon Black EDR User Guide

12 April 2024

VMware Carbon Black EDR 7.8.0

vmware by Broadcom

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

https://docs.vmware.com/

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

Contents

Preface

This guide describes how to use Carbon Black EDR. It is written for both Carbon Black EDR and Carbon Black Hosted EDR administrators.

This is your guide to managing Carbon Black EDR and sensors and using Carbon Black EDR to monitor file activity and threats on your endpoints. The content includes Carbon Black EDR concepts, architecture, and terminology.

Intended Audience

This documentation is written for administrators, Security Operations Center (SOC), and Incident Response (IR) personnel. It is intended for people who set up and maintain security for endpoints and networks, and for users who assess potential vulnerabilities and detect advanced threats. Staff who manage Carbon Black EDR activities should be familiar with:

- Linux, Microsoft Windows, and macOS operating systems
- Web applications
- Desktop infrastructure (especially in-house procedures for software roll-outs, patch management, and antivirus software maintenance)
- Effects of unwanted software

Additional Documentation

- *Carbon Black EDR Release Notes* – Provides information about new and modified features, issues resolved, general improvements in this release, and known issues and limitations. It also includes required or suggested preparatory steps before installing the Carbon Black EDR server.
- *Carbon Black EDR Server Operating Environment Requirements Guide* – Describes performance and scalability considerations in deploying a Carbon Black EDR server.
- *Carbon Black EDR Sensor OERs* – These five documents describe the operating environment requirements for Carbon Black EDR Windows, macOS, and Linux sensors.
- *Carbon Black EDR Sensor Installation Guide* – Describes how to install, upgrade, uninstall, and troubleshoot Carbon Black EDR sensors.
- *Carbon Black EDR Server Configuration Guide* – Describes the Carbon Black EDR server configuration file (`cb.conf`), including options, descriptions, and parameters.

- *Carbon Black EDR Server Cluster Management Guide* – Describes how to install, manage, and backup/restore a Carbon Black EDR non-containerized server/cluster.
- *Carbon Black EDR Containerized Server Guide* – Describes how to install, manage, and backup/restore a Carbon Black EDR containerized server/cluster.
- *Carbon Black EDR Unified View User Guide* – Describes how to install and manage Carbon Black EDR Unified View.
- *Carbon Black EDR Integration Guide* – Provides information for administrators who are responsible for integrating Carbon Black EDR with various tools and applications, such as Carbon Black App Control, EMET, VDI, SSO, and more.
- Carbon Black EDR API – Documentation for the Carbon Black EDR REST API is located at https://developer.carbonblack.com/reference/enterprise-response . Documentation for the Python module for easy access to the REST API is hosted at https://cbapi.readthedocs.io .
- Carbon Black EDR connectors – Documentation describing how to install, configure and maintain various connectors is located at https://developer.carbonblack.com/reference/enterprise-response/connectors/ . A connector enables communication between a third-party product and a Carbon Black EDR server.

Document History

For a list of changes made to this guide, see Chapter 25 Document History.

Copyrights and Notices

This topic describes Carbon Black copyright notices for Carbon Black EDR.

WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

We acknowledge the use of the following third-party software in the Carbon Black EDR software product:

- Antlr python runtime - Copyright (c) 2010 Terence Parr
- Backbone routefilter - Copyright (c) 2012 Boaz Sender
- Backbone Upload - Copyright (c) 2014 Joe Vu, Homeslice Solutions
- Backbone Validation - Copyright (c) 2014 Thomas Pedersen (http://thedersen.com)
- Backbone.js - Copyright (c) 2010–2014 Jeremy Ashkenas, DocumentCloud
- Beautifulsoup - Copyright (c) 2004–2015 Leonard Richardson
- Canvas2Image - Copyright (c) 2011 Tommy-Carlos Williams (http://github.com/devgeeks)
- Code Mirror - Copyright (c) 2014 by Marijn Haverbeke marijnh@gmail.com and others
- D3js - Copyright 2013 Mike Bostock. All rights reserved
- FileSaver - Copyright (c) 2011 Eli Grey
- Font-Awesome - Copyright Font Awesome by Dave Gandy (http://fontawesome.io)
- Fontello - Copyright (c) 2011 by Vitaly Puzrin
- Freewall - Copyright (c) 2013 Minh Nguyen
- FullCalendar - Copyright (c) 2013 Adam Shaw
- Gridster - Copyright (c) 2012 Ducksboard
- Heredis - Copyright (c) 2009–2011, Salvatore Sanfilippo and Copyright (c) 2010–2011, Pieter Noordhuis
- Java memcached client - Copyright (c) 2006–2009 Dustin Sallings and Copyright (c) 2009–2011 Couchbase, Inc.
- Javascript Digest Auth - Copyright (c) Maricn Michalski
- Javascript marked - Copyright (c) 2011–2014, Christopher Jeffrey (https://github.com/chjj/)
- Javascript md5 - Copyright (c) 1998 - 2009, Paul Johnston & Contributors All rights reserved
- Javascript modernizr - Copyright (c) 2009 - 2013 Modernizr
- Javascript zip - Copyright (c) 2013 Gildas Lormeau. All rights reserved
- Jedis - Copyright (c) 2010 Jonathan Leibiusky
- Jmousewheel - Copyright (c) 2013 Brandon Aaron
- Joyride - Copyright (c) 1998 - 2014 ZURB, Inc. All rights reserved.

- JQuery - Copyright (c) 2014 The jQuery Foundation
- JQuery cookie - Copyright (c) 2013 Klaus Hartl
- JQuery flot - Copyright (c) 2007–2014 IOLA and Ole Laursen
- JQuery Foundation - Copyright (c) 2013–2014 ZURB, Inc.
- JQuery placeholder - Copyright (c) Mathias Bynens (http://mathiasbynens.be/)
- JQuery sortable - Copyright (c) 2012, Ali Farhadi
- JQuery sparkline - Copyright (c) 2009–2012 Splunck, Inc.
- JQuery spin - Copyright (c) 2011–2014 Felix Gnass [fgnass at neteye dot de]
- JQuery tablesorter - Copyright (c) Christian Bach
- JQuery timepicker - Copyright (c) Jon Thornton, thornton.jon@gmail.com (https://github.com/jonthornton)
- JQuery traffic cop - Copyright (c) Jim Cowart
- JQuery UI - Copyright (c) 2014 jQuery Foundation and other contributors
- jScrollPane - Copyright (c) 2010 Kelvin Luck
- Libcurl - Copyright (c) 1996 - 2014, Daniel Stenberg, daniel@haxx.se
- libfreeimage.a - FreeImage open source image library
- Meld3 - Supervisor is Copyright (c) 2006–2015 Agendaless Consulting and Contributors
- moment.js - Copyright (c) 2011–2014 Tim Wood, Iskren Chernev, Moment.js contributors
- MonthDelta - Copyright (c) 2009–2012 Jess Austin
- Mwheelintent.js - Copyright (c) 2010 Kelvin Luck
- nginx - Copyright (c) 2002–2014 Igor Sysoev and Copyright (c) 2011–2014 Nginx, Inc.
- OpenSSL - Copyright (c) 1998–2011 The OpenSSL Project. All rights reserved
- PostgreSQL - Portions Copyright (c) 1996–2014, The PostgreSQL Global Development Group and Portions Copyright (c) 1994, The Regents of the University of California
- PostgreSQL JDBC drivers - Copyright (c) 1997–2011 PostgreSQL Global Development Group
- Protocol Buffers - Copyright (c) 2008, Google Inc.
- pyperformance - Copyright 2014 Omer Gertel
- Pyrabbit - Copyright (c) 2011 Brian K. Jones
- Python decorator - Copyright (c) 2008, Michele Simionato
- Python flask - Copyright (c) 2014 by Armin Ronacher and contributors
- Python gevent - Copyright Denis Bilenko and the contributors (http://www.gevent.org)

- Python gunicorn - Copyright 2009–2013 (c) Benoit Chesneau benoitc@e-engura.org and Copyright 2009–2013 (c) Paul J. Davis paul.joseph.davis@gmail.com
- Python haigha - Copyright (c) 2011–2014, Agora Games, LLC All rights reserved
- Python hiredis - Copyright (c) 2011, Pieter Noordhuis
- Python html5 library - Copyright (c) 2006–2013 James Graham and other contributors
- Python Jinja - Copyright (c) 2009 by the Jinja Team
- Python kombu - Copyright (c) 2015–2016 Ask Solem & contributors. All rights reserved.
- Python Markdown - Copyright 2007, 2008 The Python Markdown Project
- Python netaddr - Copyright (c) 2008 by David P. D. Moss. All rights reserved.
- Python ordereddict - Copyright (c) Raymond Hettinger on Wed, 18 Mar 2009
- Python psutil - Copyright (c) 2009, Jay Loden, Dave Daeschler, Giampaolo Rodola'
- Python psycogreen - Copyright (c) 2010–2012, Daniele Varrazzo daniele.varrazzo@gmail.com
- Python redis - Copyright (c) 2012 Andy McCurdy
- Python Seasurf - Copyright (c) 2011 by Max Countryman
- Python simplejson - Copyright (c) 2006 Bob Ippolito
- Python sqlalchemy - Copyright (c) 2005–2014 Michael Bayer and contributors. SQLAlchemy is a trademark of Michael Bayer
- Python sqlalchemy-migrate - Copyright (c) 2009 Evan Rosson, Jan Dittberner, Domen Kozar
- Python tempita - Copyright (c) 2008 Ian Bicking and Contributors
- Python urllib3 - Copyright (c) 2012 Andy McCurdy
- Python werkzeug - Copyright (c) 2013 by the Werkzeug Team, see AUTHORS for more details
- QUnitJS - Copyright (c) 2013 jQuery Foundation (http://jquery.org/)
- redis - Copyright (c) by Salvatore Sanfilippo and Pieter Noordhuis
- Simple Logging Facade for Java - Copyright (c) 2004–2013 QOS.ch
- Six - Copyright (c) 2010–2015 Benjamin Peterson
- Six - yum distribution - Copyright (c) 2010–2015 Benjamin Peterson
- Spymemcached / Java Memcached - Copyright (c) 2006–2009 Dustin Sallings and Copyright (c) 2009–2011 Couchbase, Inc.
- Supervisord - Supervisor is Copyright (c) 2006–2015 Agendaless Consulting and Contributors
- Switchery - Copyright (c) 2013–2014 Alexander Petkov
- Toastr - Copyright (c) 2012 Hans Fjallemark & John Papa

Carbon Black

3421 Hillview Ave., Palo Alto, CA 95131 USA

Tel: 877.248.9098

Email: cb-support@vmware.com

Carbon Black EDR User Guide

Product Version: 7.8.0

Document Revision Date: 12 April 2024

Carbon Black EDR Overview

1

This topic introduces Carbon Black EDR, explains key concepts and terminology, and suggests operating strategies for managing sensors and data to provide the visibility, detection, and response capabilities in the Carbon Black solution.

Read the following topics next:

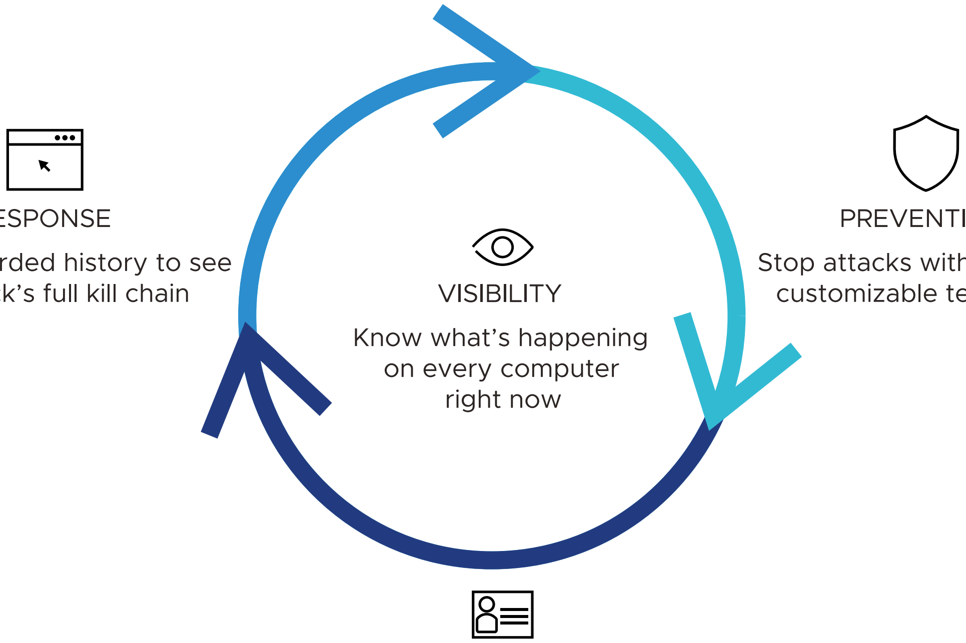
- What is Carbon Black EDR?
- Carbon Black EDR Terminology
- System Architecture
- Data Flow Diagrams
- Workflow Overview

What is Carbon Black EDR?

Carbon Black EDR provides endpoint threat detection and a rapid response solution for Security Operations Center (SOC) and Incident Response (IR) teams.

With Carbon Black EDR, enterprises can continuously monitor and record all activity on endpoints and servers. The combination of Carbon Black EDR's endpoint visibility with Carbon Black Threat Intel helps enterprises to proactively hunt for threats, customize their detection, and respond quickly. The following diagram shows how Carbon Black EDR features work together to help you answer these questions:

- How did the problem start?
- What did the threat do?
- How many machines are infected?
- How can we resolve the threat?



RESPONSE

Use a recorded history to see an attack's full kill chain

VISIBILITY

Know what's happening on every computer right now

PREVENTION

Stop attacks with proactive, customizable techniques

DETECTION

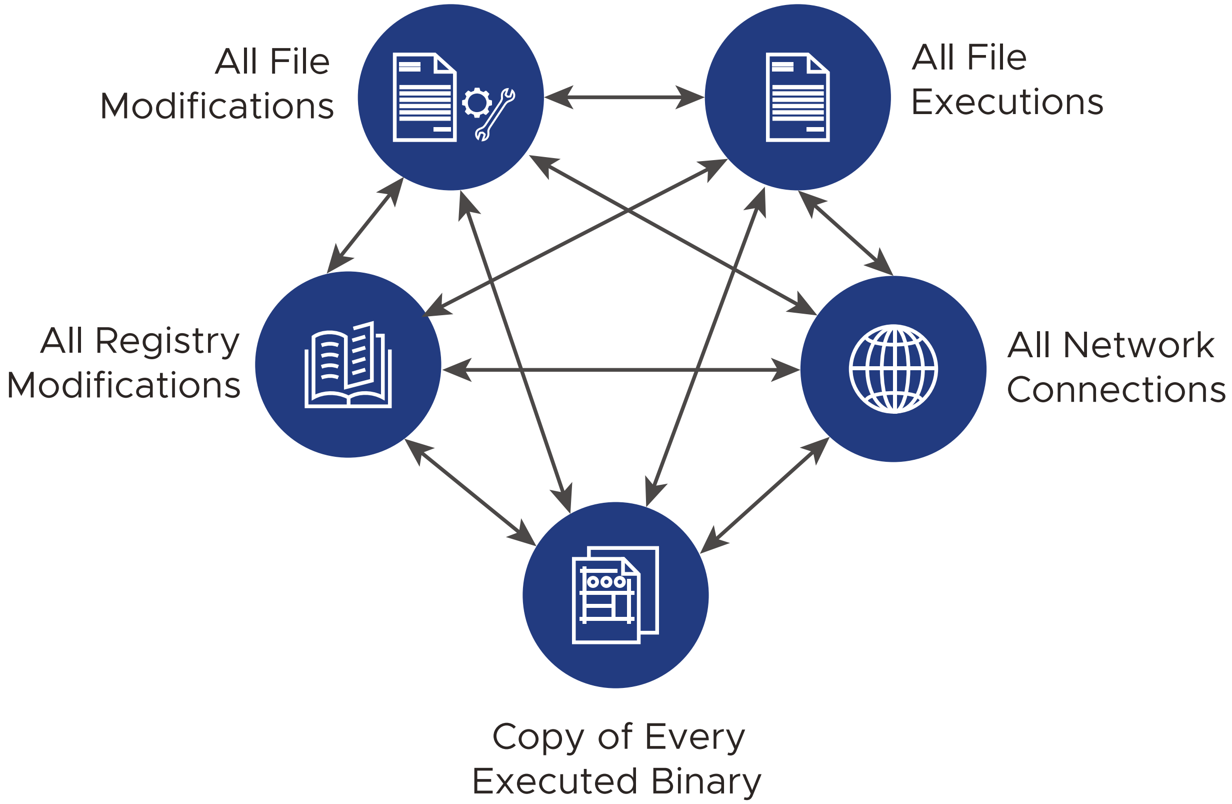
Detect attacks in real-time without signatures

Carbon Black EDR provides these solutions:

- Visibility – Know what's happening on every computer at all times. With Carbon Black EDR, you have immediate real-time visibility into the files, executions, network connections, and critical system resources on every machine, and the relationships between them. You can see how every file got there, what created it, when it arrived, what it did, if it made a network connection, if it deleted itself, if a registry setting was modified, and much more.
- Detection – See and record everything; detect attacks in real time without signatures. Carbon Black EDR's threat research team analyzes threat techniques and creates Advanced Threat Indicators (ATIs) to alert you to the presence of an attack. These ATIs look for threat indicators and are not based on signatures. You can detect advanced threats, zero-day attacks, and other malware that evades signature-based detection tools—in real time. There is no wait for signature files, no testing and updating .dat files, and no sweeps, scans or polls. You get immediate, proactive, signature-less detection.
- Response – Use a recorded history to see the full "kill chain" of an attack, and contain and stop attacks. When you need to respond to an alert or threat, you will instantly have the information you need to analyze, scope, contain, and remediate the problem. With the recorded details about every machine, you can "go back in time" to see what happened on any of your machines to understand the full "kill chain" of an attack. You will also have a copy of any binary that ever executed, so you can analyze it yourself, submit it to a third party, and so on. You can also contain and stop attacks by globally blocking the execution of any file automatically or with a single click.
- Prevention via Carbon Black App Control – Stop attacks with proactive, signatureless prevention techniques by integrating Carbon Black App Control with Carbon Black EDR. With Carbon Black App Control, you can choose from different forms of advanced endpoint protection to match your business and systems. Carbon Black App Control's proactive "Default-Deny" approach ensures that only software you trust can run on your machines. Carbon Black App Control's "Detect-and-Deny" technology uses ATIs to detect malware and stop its execution, and Carbon Black App Control's unique "Detonate-and-Deny" approach automatically can send every new file that arrives on any endpoint or server to leading network security tools for "detonation." If a tool reports finding malicious files, Carbon Black App Control can automatically stop them from running on all of your machines.

Carbon Black EDR accelerates detection by going beyond signatures, and reduces the cost and complexity of incident response. Using a real-time endpoint sensor, Carbon Black EDR delivers clear and accurate visibility and automates data acquisition by continuously recording and maintaining the relationships of every critical action on all machines, including events and event types such as executed binaries, registry modifications, file modifications, file executions, and network connections.

Carbon Black EDR provides a cross-process event type that records an occurrence of a process that crosses the security boundary of another process. While some of these events are benign, others can indicate an attempt to change the behavior of the target process by a malicious process.


All File Modifications
All File Executions
All Registry Modifications
All Network Connections
Copy of Every Executed Binary

Unlike scan-based security solutions, Carbon Black EDR can expand detection beyond the moment of compromise with its robust endpoint sensor and access to the information provided by Carbon Black Threat Intel.

Carbon Black Threat Intel provides three types of intelligence:

- Carbon Black Threat Intel Reputation – A cloud-based intelligence database that provides highly accurate and up-to-date insight into known-good, known-bad, and unproven software. It provides IT and security teams with actionable intelligence about the software installed in their enterprise. The capabilities of the reputation service are further enhanced by feeds from third party partners.
- Carbon Black EDR Threat Indicators – Search for patterns of behavior or indicators of malicious behavior. Unlike signature-based detection, threat indicators can recognize distinct attack characteristics, based on the relationships between network traffic, binaries, processes loaded, and user accounts. Carbon Black EDR also offers watchlists that are fully customizable saved searches that you can use to look for specific threat indicators.
- Third Party Attack Classification – Uses intelligence feeds from third-party sources to help you identify the type of malware and the threat actor group behind an attack. This enables security teams to have a better understanding of attacks so that they can respond more quickly and effectively. You can also leverage your own intelligence feeds to enhance response capabilities.

Carbon Black EDR compares endpoint activity with the latest synchronization of Carbon Black Threat Intel feeds as it is reported. You can add intelligence feeds that you already have set up to give you zero-friction consumption of threat intelligence in Carbon Black EDR, regardless of the source.

Carbon Black EDR’s sensor is lightweight and can be easily deployed on every endpoint, requiring little to no configuration. This enables endpoint security analysts and incident responders to deploy thousands of sensors across their environment to immediately answer key response questions.

Carbon Black EDR’s continuously-recorded sensor data is stored in a central server, which lets your team see and understand the entire history of an attack, even if it deleted itself.

Carbon Black EDR integrates with leading network security providers. This integration enables you to prioritize alerts that are detected on the network by correlating them with events that occurred on endpoints and servers. This enables you to fully investigate your entire enterprise instantly to accelerate detection, reduce dwell time, minimize scope, and immediately respond to and contain advanced threats.

You can use Carbon Black EDR’s APIs to customize or integrate with existing security technologies that you are using, and Security Information and Event Management systems (SIEMs).

Carbon Black Hosted EDR includes extensive support for programmatic access to the underlying data and configuration through APIs.

Documentation, example scripts, and a helper library for each of these libraries is available at https://developer.carbonblack.com .

Carbon Black EDR Terminology

This section defines some key terms you will encounter when using Carbon Black EDR.

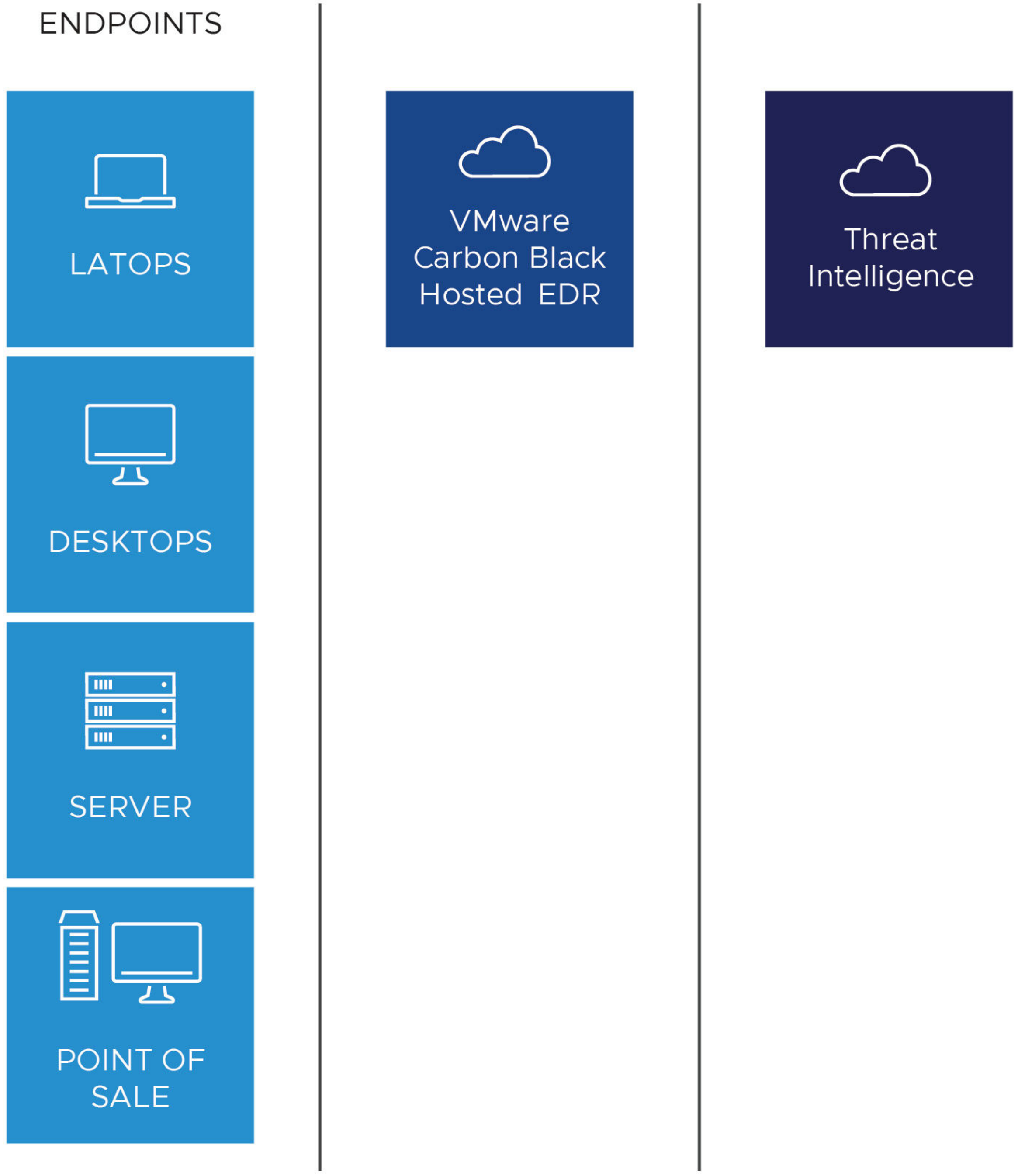
Term	Definition
Binary	Executable file (for example, PE Windows file, ELF Linux file, or Mach-O Macintosh file) that is loaded onto a computer file in binary form for computer storage and processing purposes. Carbon Black EDR only collects binaries that execute. It does not collect scripts, batch files, or computer files that are created or modified. ■ Carbon Black EDR collects the script or batch file names from command prompts and command lines. ■ Carbon Black EDR collects file names and paths as they are created or modified.
Carbon Black EDR Sensor	Lightweight data gatherers installed on hosts on the deployed network. They gather event data on the hosts and securely deliver it to the Carbon Black EDR server for storage and indexing.
Carbon Black EDR Server	A CentOS server that exists on the deployed network. It receives data from sensors, stores and indexes that data, and provides access to the data through the Carbon Black EDR console.
Carbon Black Threat Intel Feeds	Pre-configured threat intelligence feeds. These feeds contain threat intelligence data. These feeds come from various sources: ■ Carbon Black ■ Our MSSP/IR partners ■ Our customers ■ Open-source Carbon Black Threat Intel feeds provide a list of Indicators of Compromise (IOCs) and contextual information based on binary/process attributes and events (MD5, SHA-256, IP, domain). These attributes and events are scored and rated, and then correlated with any matching files in your environment. For more information, see Chapter 14 Threat Intelligence Feeds.
Carbon Black Threat Intel Server	A server that is managed by Carbon Black and augments the functionality of the Carbon Black EDR server.
Data File	A computer file that is a resource for storing information that requires a computer program (executable or binary file) to run. Data files are not captured by the Carbon Black EDR sensor.
Indicators of Compromise (IOCs)	Carbon Black EDR sensors constantly monitor your computers for IOCs and send alerts to the Carbon Black EDR console when detected. Queries are dynamic indicators that look at behaviors that are continuously recorded by sensors on endpoints and centrally recorded for analysis. Hashes (MD5, SHA-256), IP addresses, and domain names are static indicators that are similar to signatures. They are used to identify suspected malicious activity.
MD5	Unique cryptographic hash identifier for a binary instance in Carbon Black EDR.
Process	An instance of the execution of a binary file.
Watchlist	Fully customizable searches that contain lists you can use to track specific IOCs. Watchlists are saved searches that are visible to all users. They can be used for searching either processes or binary executable files.

System Architecture

This section provides a system architecture overview for both Carbon Black Hosted EDR and Carbon Black EDR. In both systems, the server records events that are related to file changes. Copies of files and the data that changed are not recorded

Carbon Black Hosted EDR

This section illustrates the components of a Carbon Black Hosted EDR installation.



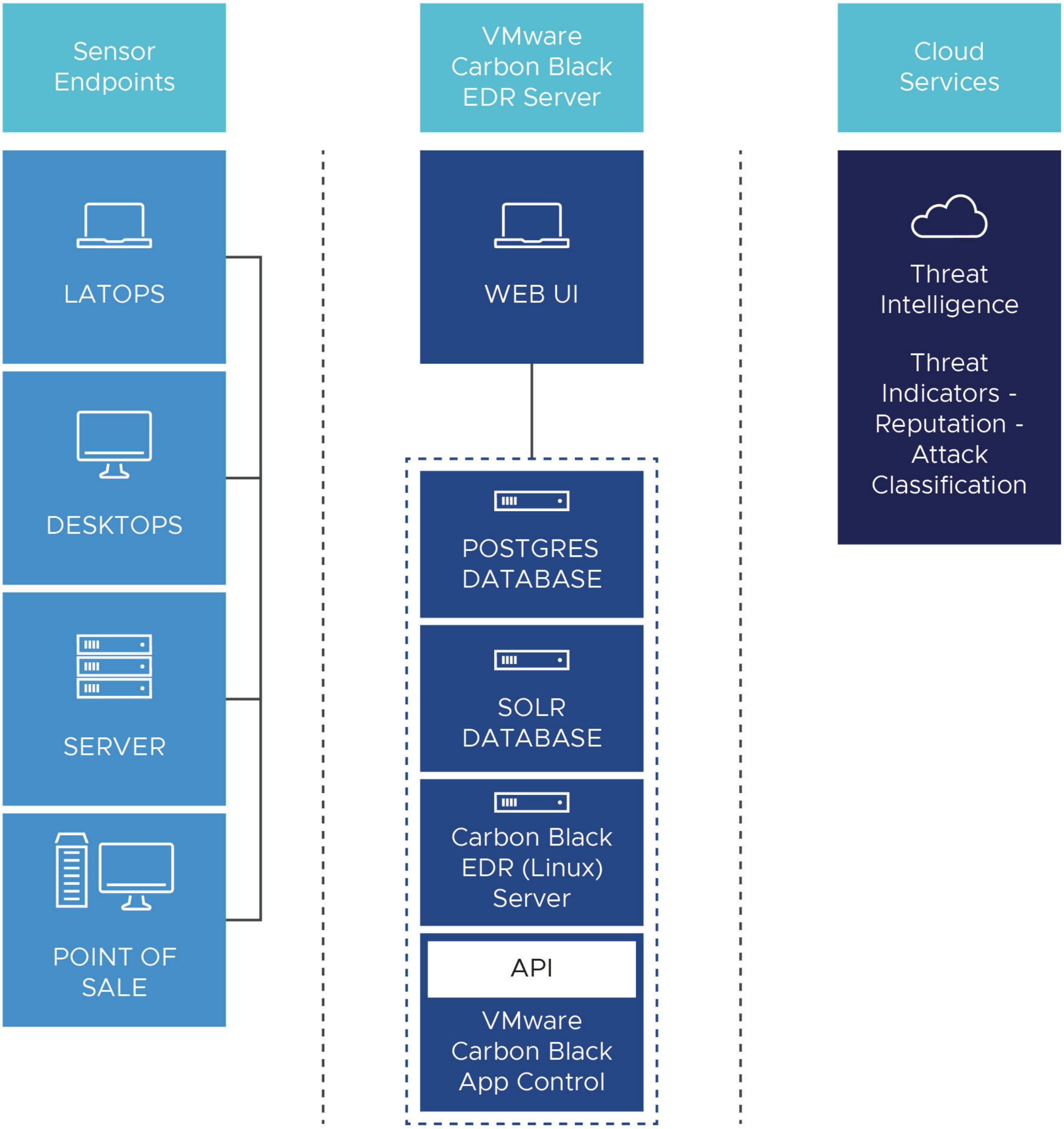
Components are:

- Sensors that can be installed on various endpoints such as laptops, desktops, servers, and point of sale (POS) machines.
- A cloud service that collects sensor data and makes it accessible with a web user interface or API.
- The threat intelligence that includes the Carbon Black Threat Intel Reputation, Carbon Black App Control, and Carbon Black EDR threat indicators, and third-party attack classification using Threat Intel partner feeds.

If your company is also using Carbon Black App Control, you can integrate it with Carbon Black Hosted EDR. By leveraging Carbon Black App Control, you can contain advanced threats by globally blocking or banning them through Carbon Black App Control's customizable prevention techniques in the midst of a response. See the *Carbon Black EDR Integration Guide*.

Carbon Black EDR

Carbon Black EDR server software is installed on a Linux server. The following diagram illustrates the components of a Carbon Black EDR installation.


Sensor Endpoints
LATOPS
DESKTOPS
SERVER
POINT OF SALE
VMware Carbon Black EDR Server
WEB UI
POSTGRES DATABASE
SOLR DATABASE
Carbon Black EDR (Linux) Server
API
VMware Carbon Black App Control
Cloud Services
Threat Intelligence
Threat Indicators - Reputation - Attack Classification

Components are:

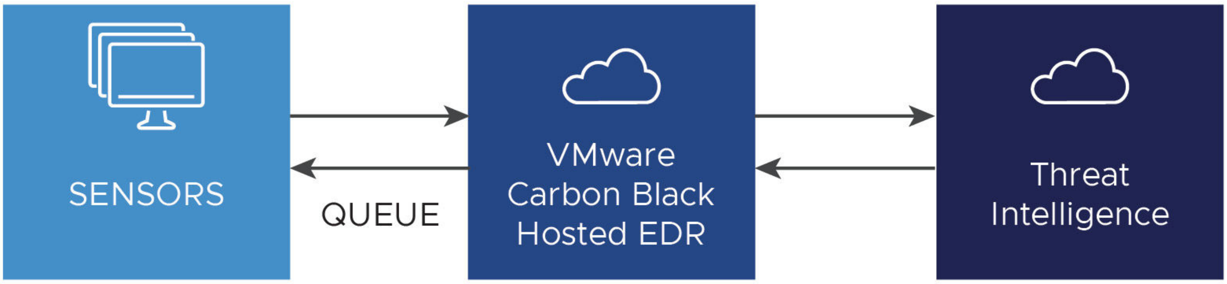
- Sensors that can be installed on various endpoints such as laptops, desktops, servers, and point of sale (POS) machines.
- A server that collects sensor data and makes it accessible with a web user interface or an API.
- The threat intelligence that includes the Carbon Black Threat Intel Reputation, Carbon Black App Control, and Carbon Black EDR threat indicators, and third-party attack classification using Carbon Black Threat Intel partner feeds.

If your company is also using Carbon Black App Control, you can integrate it with Carbon Black EDR. See the *Carbon Black EDR Integration Guide*. With Carbon Black App Control, you can contain advanced threats by globally blocking or banning them through Carbon Black App Control's customizable prevention techniques in the midst of a response.

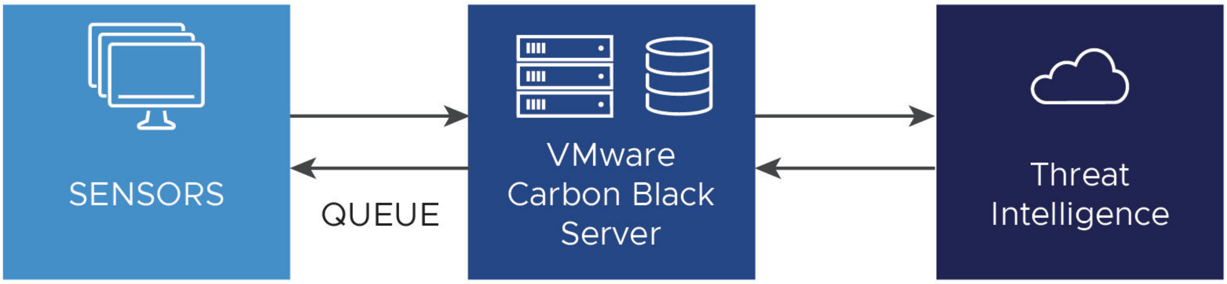
Data Flow Diagrams

This section describes Carbon Black Hosted EDR and Carbon Black EDR data flows.

The following diagram illustrates the Carbon Black Hosted EDR data flow:



The following diagram illustrates the Carbon Black EDR data flow:



As soon as a sensor is installed, it begins buffering activity to report to the cloud service. This includes:

- Currently running processes that create events
- Binary executions
- File executions and modifications
- Network connections
- Registry modifications
- Cross-process events (events that cross the security boundaries of other processes)
- PowerShell fileless scriptload events

Every few minutes, sensors check in with the cloud service to report what they have buffered, even if they are reporting that they have nothing buffered. When a sensor checks in, the cloud service responds, letting the sensor know when to send the data and how much data to send.

As the cloud service records data from sensors, the data is compared with the latest synchronization from any enabled Carbon Black Threat Intel feed partner. In most cases, incremental synchronizations occur hourly. Full synchronizations occur once every 24 hours by default.

Some Carbon Black Threat Intel feeds provide a list of all of the IOCs they track. Some feeds only include reports on files (identified through their MD5 or SHA-256 hashes) that are observed in your enterprise.

If you enable data sharing with the Carbon Black Threat Intel partners, Carbon Black EDR pushes MD5 hashes that are observed by sensors and binaries originating from your enterprise to their cloud services. If there is a corresponding report or record, the feed is updated to include that information. If there is no corresponding third party-report, one is requested and when available, included in the feed.

When information about a specific binary is included in these feeds, the information remains there, even if the binary it is associated with is deleted from your endpoints and is no longer present in your environment.

The following table provides key additional information about data flows:

Data Flow	Description
Sensor to Server	■ All communications are through HTTPS. ■ The TCP port is 443 by default, but is configurable. ■ Communications are always initiated from sensor to server (never from server to sensor). ■ By default, communications are mutually authenticated by statically pinned TLS certificates, both client and server. There is also an option to substitute user-provided certificates and use stricter validation. Sensors have the server's certificate embedded, and the server has all client certificates embedded. See Chapter 7 Managing Certificates for more information. ■ All communications require a minimum of TLSv1+; only allow FIPS-compliant ciphers and use a 2048-bit Diffie Hellman key. ■ Sensor communication through a proxy is unsupported, unless the proxy is deployed in a transparent, in-line configuration. ■ Sensor communication is supported through transparent proxies. Due to certificate pinning, communication is not supported through traffic inspection proxies, or any other device that would affect SSL certificates. ■ The Windows sensor honors settings that are configured via a `proxy.pac` file. (This does not change the requirement that any proxy that is used must not modify SSL certificates or otherwise attempt to bypass the secure communications between sensor and server.) ■ Sensor communication through an TLS intercept/decryption device is not currently supported, even for in-line proxy configurations. ■ The server's sensor-facing interface can be configured in a DMZ to support endpoints outside the corporate LAN
Server to Alliance Server and Carbon Black Threat Intel	■ All communications are explicitly opt-in. ■ All communications are HTTPS. ■ This connection is required for threat intelligence that is provided by Carbon Black EDR. ■ TCP is 443 to api.alliance.carbonblack.com and threatintel.bit9.com. ■ Proxies are supported.
Server to yum Repository	■ TCP is 443 for HTTPS to yum.distro.carbonblack.io. ■ TCP is 80 to a CentOS or RHEL.

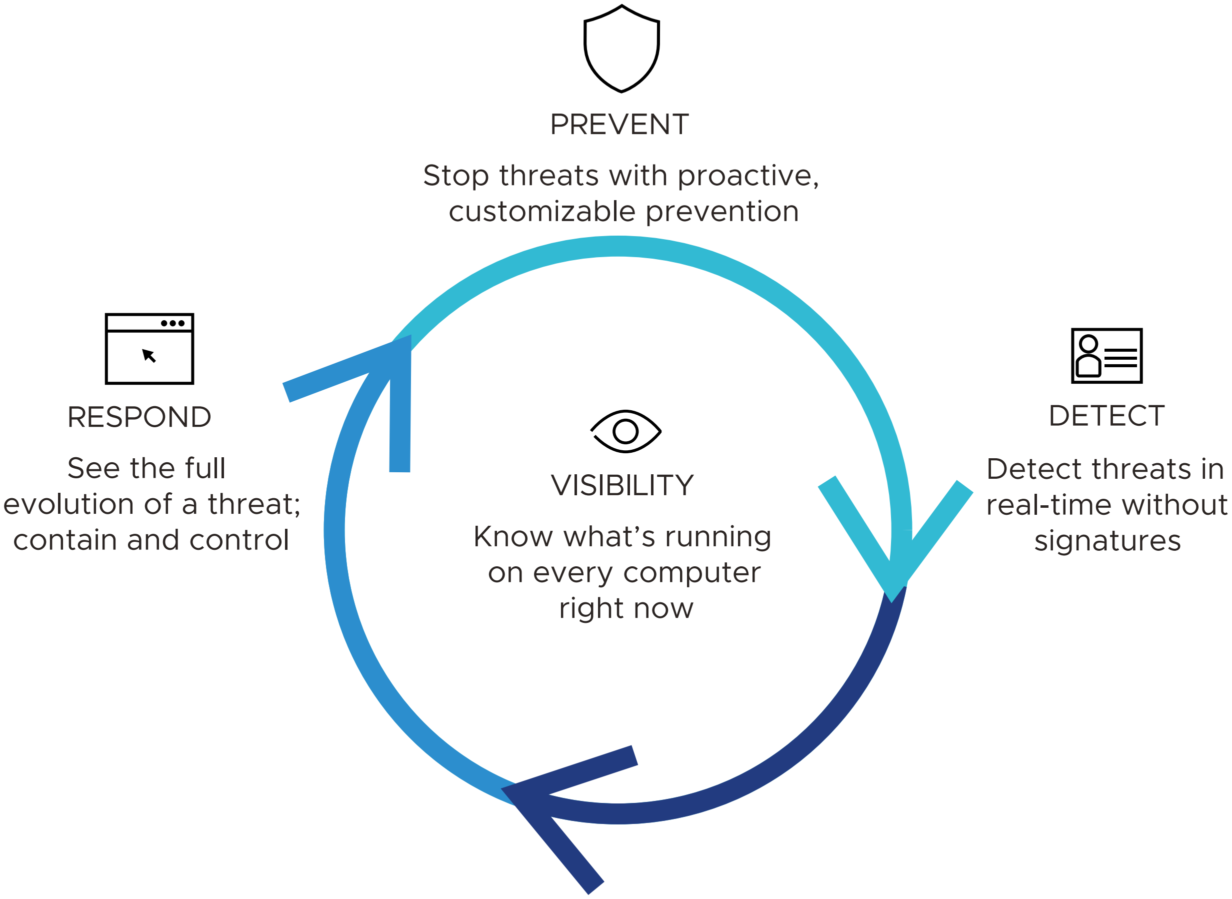
Workflow Overview

After sensors are installed and configured, your IT and security teams can perform basic tasks on a regular basis to ensure that there are no threats on any endpoint in your enterprise. Access to the Carbon Black EDR user interface is via browser, although you can perform some functions through an API.

Note Google Chrome is the only supported browser for this release. Although Firefox can be used, it causes rendering issues on some pages and is not recommended. Other browsers should not be used for console access.

The basic workflow is continuous: you search for threats, analyze them, resolve then, and using the tools of your choice, prevent them from happening again. As you search, you can tag any items that seem unusual or that merit further investigation and then drill down further to find out more details about those items.

Carbon Black EDR provides you with tools to help you detect and fix threats to your system. The following diagram shows the basic Carbon Black EDR workflow:



PREVENT

Stop threats with proactive, customizable prevention

DETECT

Detect threats in real-time without signatures

RESPOND

See the full evolution of a threat; contain and control

VISIBILITY

Know what's running on every computer right now

The following table shows how Carbon Black EDR provides solutions to problems.

Problem	Solution
What is the entry point of the threat?	Find out how the attacker got into your systems. Get oriented with visibility into everything that is running on every computer in your enterprise using the **Process Search** feature.
What did the attacker do?	Look deeper into suspicious processes and events to detect evidence of damage. Select processes that look suspicious and drill deeper using the **Process Analysis** feature.
How many machines were compromised?	Find out the scope of the damage by digging deeper into details about detected threats by using the **Process Details** and **Binary Details** pages. Set up Carbon Black Threat Intel feeds and **Watchlists** by defining characteristics of interesting activity that you want to be notified about and receiving notifications as you need them. Create **Investigations** of suspicious processes to keep track of key events during a given response.
How do we respond to threats?	Find out how bad the threat is, and then determine how to respond to it by seeing its full evolution, containing the threat, and then controlling it.
How do we stop the threat from happening again?	Use the **Go Live** feature to directly access content on endpoints that are running sensors. Set up **Watchlists** and Carbon Black Threat Intel feeds that identify specific issues, and use the feeds and watchlists to perform continuous searches. This can provide immediate detection to help you stop the threat from happening again, and ensures that you know of any new related activity.
How do we isolate threats?	You can isolate one or more Windows endpoints from the rest of your network and the Internet through the Carbon Black EDR console. For more information, see Isolating an Endpoint.

Note Access to Carbon Black EDR features is determined by the permissions that a logged-in user has. See Chapter 3 Managing User Accounts (Carbon Black EDR) and Chapter 4 Managing User Accounts (Carbon Black Hosted EDR).

Getting Started with Carbon Black EDR

2

This section explains how to log in and out of Carbon Black EDR, and includes instructions for using two-factor authentication for Carbon Black Hosted EDR accounts. It also introduces the Carbon Black EDR console controls that are available through the navigation bar and top menu, and summarizes the features that are accessible from those locations.

.

Read the following topics next:

- Logging in to Carbon Black EDR
- Logging out of the Carbon Black EDR console
- Carbon Black EDR Console Controls

Logging in to Carbon Black EDR

The Carbon Black EDR console is a browser-based user interface for accessing the Carbon Black EDR server and the information it collects from sensors and Carbon Black Threat Intel feeds. You log into the Carbon Black EDR console from a supported web browser on any computer that has access to your server.

.

Note Google Chrome is the only supported browser for this release. Although Firefox can be used, it causes rendering issues on some pages and is not recommended. Other browsers should not be used for Carbon Black EDRCarbon Black EDR console access.

To log into the console:

1. From a supported web browser, enter the path to the Carbon Black EDR server.
2. If your browser displays a warning about the certificate, you can safely ignore the warning and click through the remaining confirmation windows.

   Note To avoid future certificate warnings, accept the certificate permanently.

3. In the Login dialog box, enter your user name and password.
4. Click the **Login** button to display the **Head Up Display (HUD)** page.

Logging in and Configuring Two-Factor Authentication

This section explains how to log in to Carbon Black Hosted EDR and configure two-factor authentication.

Logging In for the First Time from an Email Invitation (Carbon Black Hosted EDR)

If you have received an email inviting you to access Carbon Black Hosted EDR, use the link in the email to either sign in with an existing account or create a new account.

Note The email link expires seven days after receipt.

To log in from an email invitation:

1 Click the link in your invitation email to open the **Login** dialog box.

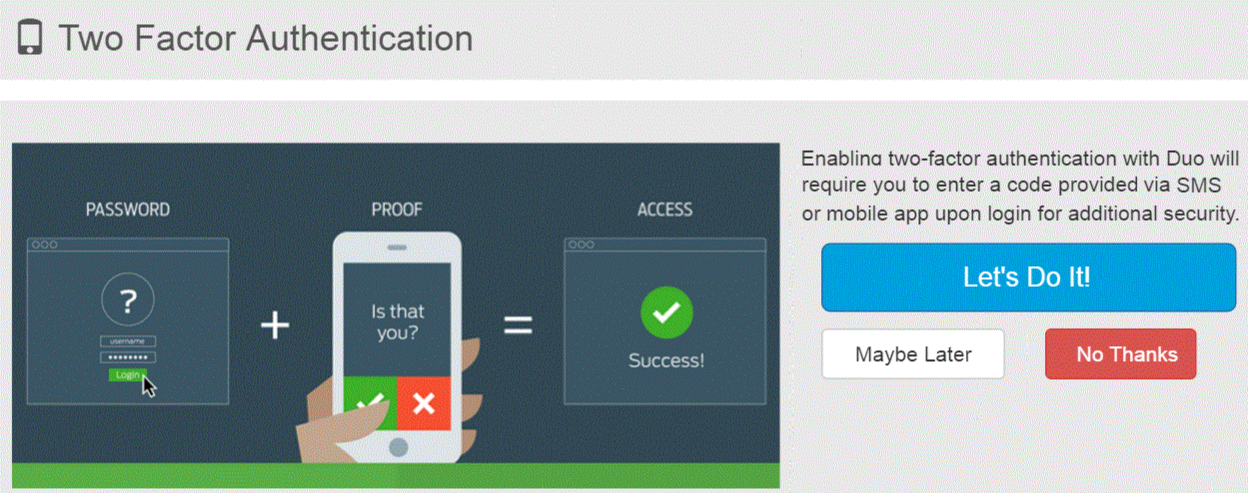
2 Do one of the following:

- If you already have an account, click **Sign In** and follow the procedures detailed in Logging in After Initial Login (Carbon Black Hosted EDR).
- To create a new account, enter values in the **Username** , **First** , **Last** , **Password** , and **Confirm Password** fields, and click Sign up.

3 If you are creating a new account, read the terms and conditions, and click **Accept** .

> **Note** This page only appears the first time you access the Carbon Black Hosted EDR or when the terms and conditions are updated.

The **Two Factor Authentication** wizard opens where you can optionally configure two-factor authentication. Two-factor authentication adds a second authentication factor to your server and facilitates authentication management and security monitoring.



Two Factor Authentication

Enabling two-factor authentication with Duo will require you to enter a code provided via SMS or mobile app upon login for additional security.

Two-factor authentication is available through Duo and requires that you download the Duo Mobile application on a device. (For more information, see https://duo.com).

If you change your mind later about using two-factor authentication, you can disable it. (See Enabling/Disabling Two-Factor Authentication.)

Configuring Two-Factor Authentication (Carbon Black Hosted EDR)

This topic describes how to configure two-factor authentication for Carbon Black Hosted EDR.

To configure two-factor authentication:

1 Log into the console.

If you are logging in for the first time, or if you logged in previously and temporarily bypassed enrollment, the **Two-Factor Authentication** wizard appears.

2 On the first page of the Two Factor Authentication wizard, the following options are available:

- **Maybe Later** – Bypass enrollment for now and proceed with logging in. You will have the option to configure two-factor authentication at your next login.
- **No Thanks** – Do not enable two-factor authentication.
- **Let's Do It!** – Enable two-factor authentication.

If you decline to set up two-factor authentication, you are logged into the Carbon Black Hosted EDR.

3 To set up two-factor authentication, select **Let's Do It!** and then click **Start setup** .

The **What type of device are you adding?** screen appears. You can add any of several device types for two-factor authentication, including mobile phone devices, tablets, and landlines.

4 Select the type of device you are adding and click **Continue** .

5 Provide your phone number information and click **Continue**.

6 Select the type of phone device you are using and click **Continue**.

Note This procedure uses an iPhone as an example, but you can add other types of devices.

7 Follow the instructions to install the Duo Mobile application on your device, and then click **I have Duo Mobile installed** .

8 With your phone, scan the bar code presented in the **Activate Duo Mobile** page.

9 When the check mark appears on the bar code indicating success, click Continue.

10 On the **My Settings & Device** page, make the following selections:

a From the **My default device is** drop-down list, select your default device. This is useful when you use multiple devices for two-factor authentication.

b Select the **Automatically send me a** check box and select either **Duo Push** or **Phone Call** as your preferred communication mode with Duo Mobile.

c Click **Save**. When your device is successfully added, click **Done** (you might need to scroll down to see the **Done** button).

11 On the **Choose an authentication method** page, select an authentication method:

- **Send me a Push** – Select this recommended option to receive a Duo push notification to authenticate. Tap **Approve** on the Duo login request that is received on your phone.
- **Call Me** – Select this option to receive a phone call to authenticate.
- **Enter a Passcode** – Select this option to enter a Duo Mobile passcode to authenticate. Open the Duo Mobile application on your phone and click the key icon to generate a new passcode.

12 When authentication is successful, you are logged into Carbon Black Hosted EDR. The **HUD** page appears.

Logging in After Initial Login (Carbon Black Hosted EDR)

This topic describes how to log in to Carbon Black Hosted EDR after your initial login.

To log into the console after initial login:

1 In a supported web browser on a computer with access to your server, enter the path to the Carbon Black Hosted EDR service. In the initial dialog, click **Login with CB Cloud** .

The **Login dialog box** appears.

2 Do one of the following:

- If **Username** and **Password** fields are pre-populated, click **Sign in** .
- If the fields are not pre-populated, enter your **Username** and **Password** and click **Login** .

The system responds in one of the following ways, depending on your two-factor authentication selection:

- If you selected **No Thanks**, the HUD page appears.
- If you selected **Maybe Later** , the **Two Factor Authentication** wizard opens, prompting you to enroll. You can either decline, or decide to configure two-factor authentication.
- If you enabled two-factor authentication, the system contacts your configured device. Follow the prompts to authenticate.

After you authenticate successfully, you are logged in to the Configure Server page.

On the Configure Server page, you can do the following:

- Go to the My Servers page, where you view a list of all servers to which you have authorized access. Click a server link to access the HUD page for that server.
- Click your user name to manage account details, such as changing your password, and enabling or disabling two-factor authentication.
- Click Invite user to invite new or existing users.
- Click an existing user to manage their account and permissions. For more information, see Chapter 4 Managing User Accounts (Carbon Black Hosted EDR)

Enabling/Disabling Two-Factor Authentication

You can enable/disable two-factor authentication for your Carbon Black Hosted EDRaccount at any time.

To enable or disable two-factor authentication:

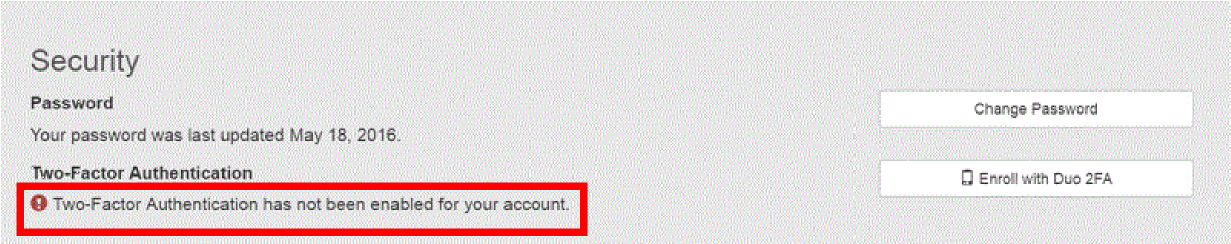
1 From the menu bar, select **Your Username > Account**.

2 In the **Security** panel of the Account page, do one of the following:

- If enabled, disable two-factor authentication by clicking **Disable Duo 2FA**.
- If disabled, enable two-factor authentication by clicking **Enroll with Duo 2FA**.

3 Follow the prompts to complete the procedure to either enable or disable Duo 2FA.

The **Security** panel of the Account page updates to display the changed status for Two-Factor Authentication.



Security

Password

Your password was last updated May 18, 2016.

Two-Factor Authentication

Two-Factor Authentication has not been enabled for your account.

Logging out of the Carbon Black EDR console

This topic describes how to log out of the Carbon Black EDR console.

The top-right corner of the Carbon Black EDR console displays your user name. Click it to display a drop-down list from which you can:

- View your profile information
- Logout of Carbon Black EDR
- For administrators only, view and modify Settings and Shared Settings.

To log out of the console:

1 Click your user name.

The user drop-down list appears.

2 Click **Logout** to log out of the console.

Carbon Black EDR Console Controls

This topic introduces the Carbon Black EDR console and describes how to navigate in it.

Navigation Bar

You use the Carbon Black EDR navigation bar to access console pages. The following table describes the options that are available for users with full Administrator/Global Administrator privileges – other users will see options that are appropriate to their privilege level. The **Teams** option appears for cloud instances only.

Link	Description
CB Logo	Opens the HUD page, which is a customizable page that provides a summary of alerts on hosts that report to your Carbon Black EDR server. See Chapter 21 Head-Up Display Page.
Threat Intelligence	Provides intelligence feeds. You can set up watchlists, incremental synchronizations, and full synchronizations with these feeds. You can also access information about process and binary matches found by each feed. See Chapter 14 Threat Intelligence Feeds
Triage Alerts	Shows events that match queries that are defined by watchlists and indicators of compromise (IOCs) that are defined by feeds. The information provides criteria that is available to search for specific events. See Managing Alerts on the Triage Alerts Page.
Watchlists	Saved queries that are performed on process events and binary data stores. The queries contain lists you can use to track specific IOCs. See Chapter 19 Watchlists
Process Search	Provides an overview of the sensor process data collection that is received from currently installed sensors. See Chapter 10 Process Search and Analysis
Binary Search	Shows the metadata of binary files that have been executed. Binary file data is tracked at the moment of execution, and is identified by MD5 hash name. See Chapter 11 Binary Search and Analysis
Go Live	This icon appears if you have enabled Go Live in **Username > Settings > Advanced Settings**. It opens a command line page that provides direct access to sensors. You can directly access content on endpoints. See Chapter 8 Responding to Endpoint Incidents
Live Query	This icon appears if you have enabled Live Query in **Username > Settings > Advanced Settings**. It allows you to run direct SQL queries against targeted endpoints. See Chapter 9 Live Query.
Investigations	A collection of tagged process events that are products of search results from searching your networks and endpoints for threats. See Chapter 18 Investigations.
Sensors	Shows data for sensors and sensor groups. Sensor groups categorize sensors that share the same configuration. You can view, define and update sensors and sensor groups on this page. See Chapter 5 Managing Sensors.
Yara Manager	Carbon Black Yara Manager provides a web-based user interface that is integrated with the Carbon Black EDR server to configure, control, and assess the status of the Yara Connector. See Chapter 17 YARA Manager.
Users	■ Carbon Black EDR displays the User Management page. Administrators can add and configure new users who can view user activity, and create and manage teams of users. Non-administrator users can use this menu item to view their user profile details. See Chapter 3 Managing User Accounts (Carbon Black EDR). ■ Carbon Black Hosted EDR displays user accounts that are authorized to access the server. See Chapter 4 Managing User Accounts (Carbon Black Hosted EDR).
Teams	Carbon Black Hosted EDR only. This link goes to the Team Management page for your instance. Administrators can configure users, view user activity, and create and manage teams of users. See Chapter 4 Managing User Accounts (Carbon Black Hosted EDR).

Link	Description
Event Forwarder	Displays only if enabled. This link opens the Event Forwarder Settings page, which lets you configure the Event Forwarder from within the Carbon Black EDR console. See Chapter 16 Event Forwarder.
Server Dashboard	Shows server statistics such as sensor statistics and server communication status. See Monitoring Sensor and Server Information.
Banned Hashes	Opens the Manage Banned Hashes page, which shows process hashes for which a ban has been created. Banned processes are blocked from running on hosts that are managed by a Carbon Black EDR sensor. See Chapter 8 Responding to Endpoint Incidents

Username Menu

The top right corner of the Carbon Black EDR console shows the name of the currently logged-in user. This topic describes the options for viewing and managing this user's experience.

A dropdown menu includes the following options:

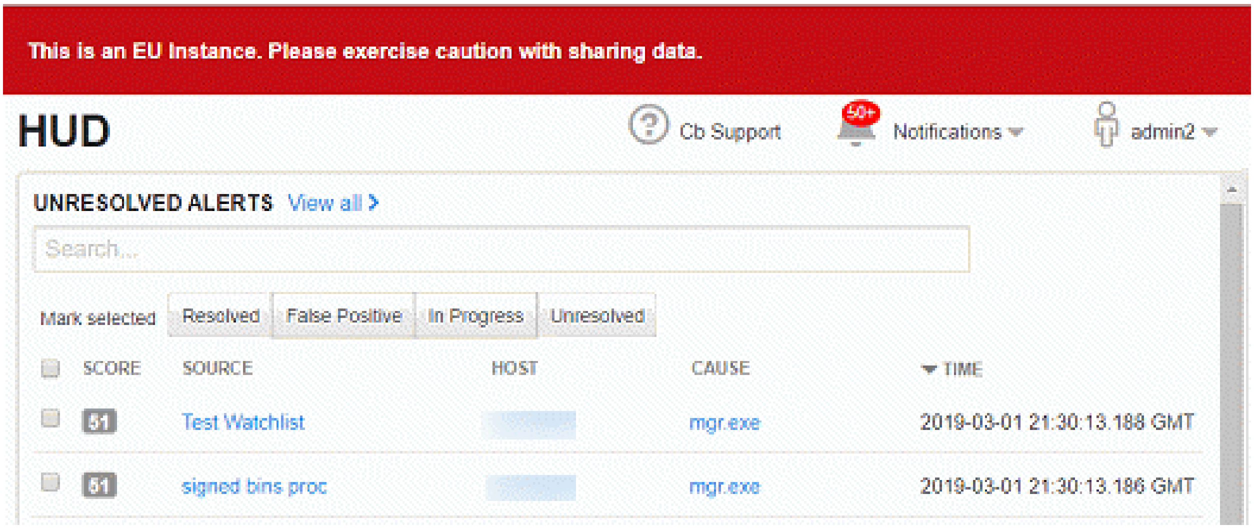
Menu Choice	Description
My Profile	Shows and allows editing of the current user’s first and last name and email address. Shows teams in which the user is a member. Provides access to dialogs for changing the user’s password and API token. For Carbon Black Hosted EDR users, the My Profile page includes a link to the user’s profile (account) page.
Sharing Settings	For administrators only, this option shows a page that allows administrators to determine whether to share different types of information with Carbon Black and its partners. See Threat Intelligence Data Sharing Settings.

Menu Choice	Description
Settings	For administrators only, this page allows administrators to view and change settings that affect the operation of Carbon Black EDR: ■ Sites – Provides a menu of sites and the ability to throttle sites by time and day of the week. ■ E-Mail – Lets you configure your own alert email server (recommended), use Carbon Black's email server, or not receive alert emails. ■ License – Shows the Carbon Black EDR server's license and allows you to apply a new license. ■ Server Nodes – Shows all of the server nodes in your cluster, their Node ID, Name, Hostname (with domain) and full URL with port. ■ Server Certificates – Shows all of the sensor-server certificates that are available on the current server. It also shows the validation method that is being used for these certificates. See Chapter 7 Managing Certificates. ■ Ingress Filters – Lets you define and view ingress filters. ■ VDI Settings – When enabled in `cb.conf` by using the `VDIAPIEnabled` setting, you can define the VDI attribute-mapping configuration in the console. Selectable attributes are Hostname, DNS Name, Computer SID, IP Address, and MAC Address. For more information about VDI support, see the *Carbon Black EDR Integration Guide* . For more information about `cb.conf,` see the *Carbon Black EDR Server Configuration Guide*. ■ Carbon Black App Control Server – Shows configuration information if this Carbon Black EDR server is integrated with Carbon Black App Control . See the *Carbon Black EDR Integration Guide*. ■ Login Customization – Allows you to display customized text upon login. ■ Advanced Settings – ■ Process Search Settings – Allows you to block certain process searches that could cause significant performance issues. See Chapter 10 Process Search and Analysis. ■ EU Data Sharing Banner – Allows you to enable and disable the display of a red banner at the top of console pages that warns users to be cautious when sharing screenshots or other data. This capability can be overriden in the `cb.conf` file by the `ShowGdprBanner` setting. See EU Data Sharing Banner for details. For more information about `cb.conf,` see the *Carbon Black EDR Server Configuration Guide*. ■ Live Response – Allows you to enable or disable Live Response, which opens a command interface for direct access to any connected host running the Carbon Black EDR sensor. This can be overriden in the `cb.conf` file by the `CbLREnabled` setting. For more information about `cb.conf,` see the *Carbon Black EDR Server Configuration Guide*. This section also includes configuration settings that can be adjusted to help improve Live Response performance. See Using Live Response for details.
Logout	Logs the current user out of the Carbon Black EDR console.

EU Data Sharing Banner

Carbon Black EDR displays information from all endpoints that report to a server through the sensor. Because some of this information can be sensitive, you might need to take extra steps to avoid exposing it in the wrong places.

As an extra precaution, Carbon Black EDR provides administrators with the ability to display a red banner at the top of console pages that warns users to be cautious. This banner is enabled and disabled on the **Advanced Settings** tab of the Settings page.

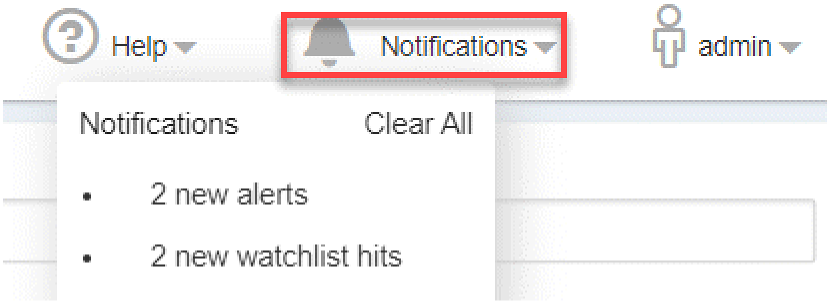


This is an EU Instance. Please exercise caution with sharing data.

☐	SCORE	SOURCE	HOST	CAUSE	TIME
☐	51	Test Watchlist	[redacted]	mgr.exe	2019-03-01 21:30:13.188 GMT
☐	51	signed bins proc	[redacted]	mgr.exe	2019-03-01 21:30:13.186 GMT

Notifications

The **Notifications** menu includes a count of new notifications and a dropdown menu that shows the number of each type of notification. Clicking any item on the menu takes you to the page that provides details for the item.



For example, if you click **2 new watchlist hits**, you go to the Watchlists page. When you click into details for all of the new notifications, the counter resets to zero and the menu then displays **No new notifications**. You can click **Clear All** to clear the menu.

Help: User Guide and Customer Support

The **Help** menu in the top right area of the console provides access to two sources of assistance for Carbon Black EDR.

- Click **User Guide** to open a new browser window or tab showing the in-product HTML version of the *Carbon Black EDR User Guide* (this document). If it displays as a tab, you can drag the tab off the current browser to display the User Guide in its own window. The online version of the *Carbon Black EDR User Guide* is compatible with Chrome browsers, preferably the latest release. You might be able to display help in other browsers, but these are not supported and they might have issues that interfere with display or performance.

- The Broadcom Carbon Black Support option opens a new browser window or tab showing the Broadcom Carbon Black Support page.

To display online documentation from the console:

1. In the top console menu, click the question mark button and choose **User Guide** from the menu. This opens the Carbon Black EDR Help home page and table of contents in a new window or tab. The controls on the help page vary depending on the size of the window, but all pages provide access to the table of contents and search features.
2. To view the table of contents if it is not visible, either expand the browser window to display the contents on the left, or click the **Table of Contents** button.
3. In the table of contents, click a right arrow icon or the name next to it in the table of contents to expand the table to show subtopics. Click the down arrow icon to collapse the items below it.
4. To search for topics using key words, enter the words in the **Search** box if it is visible, or if not, click the **Search** button to display the field.

Managing User Accounts (Carbon Black EDR) 3

This section explains how to manage user access to the Carbon Black EDR console for servers that are installed on your premises. This includes managing teams to determine user roles and manage user accounts.

For information on managing users for Carbon Black Hosted EDR, see Chapter 4 Managing User Accounts (Carbon Black Hosted EDR).

Read the following topics next:

Overview of User Management (Carbon Black EDR)

The Carbon Black EDR console is the user interface for access to Carbon Black EDR features. Each console user logs in to the system with a user name and password. Login accounts provide administrators, analysts, and others to access the features that are appropriate to their role, and allows administrators to limit unnecessary access to features or sensors.

During the Carbon Black EDR installation process, a default user account is created and assigned Global Administrator status. This user has full access to all sensors and all features. After you log in by using the default account, you can set up additional users, including other Global Administrators and users with other roles that can vary by sensor group.

The capabilities of a user are determined by the following factors:

- **Is the user a Global Administrator?** – A checkbox on the User Details page determines whether a user has **administrator privileges** (for all sensor groups and features). If a user is a Global Administrator, the following factors are not relevant.
- **What teams does the user belong to?** – The privileges of users who are not Global Administrators depend upon the **teams** to which they belong. Global Administrators can also be assigned to teams, although team membership does not affect them unless their administrator status is disabled.

- **What roles do team members have for each Sensor Group?** – Teams specify a *role* , which determines the level of privileges that their members have for each **sensor group** . There are three roles: **Analyst** , **Viewer** , and **No Access**. Teams can (and usually will) have different roles for different sensor groups. See Role-Based Privileges for Teams.
- **What is the highest role for any team this user belongs to?** – Access to some features is not restricted by sensor group, but is still controlled by the roles that are assigned to a team. These features become available to a user if the user is on at least one team that has a high enough role for at least one sensor group .

  This helps control access to features that are not specific to sensor groups, but to which you might want to restrict access. For example, threat feeds are not specific to any sensor group, but are an important tool for Carbon Black EDR threat monitoring. If a user is an Analyst on any team, that user can take actions that are available on the Threat Intelligence Feeds page.
- **Is the user an Analyst with enhanced permissions?** – For Analysts, access to especially sensitive features (such as Live Response, sensor isolation, uninstalling sensors, file banning, and tamper protection levels) is controlled by enhanced permissions. These are added on a per-user basis. See Adding Enhanced Permissions for Analysts.

A table with more specific details about team privileges is displayed in Managing User Access with Teams.

Important Creation and management of user accounts and teams is available only to Global Administrators in Carbon Black EDR installations, and by Administrators in Carbon Black Hosted EDR installations.

Managing User Access with Teams

If a Carbon Black EDR user is a Global Administrator or Carbon Black Hosted EDR Administrator, that user has access to all functionality for all computers in all sensor groups. For all other users, access to features is granted through membership on teams.

Endpoints running the Carbon Black EDR sensor are members of *sensor groups* . Each team has a defined role in each sensor group, and this role defines what it can see and do with sensors and their information. Team specifications also control access to some features that are not group-specific.

During Carbon Black EDR installation, a default sensor group (called *Default Group*) is created. You can put all sensors in the Default Group, but in order to use teams to limit access to certain sensors, we recommend that you create additional sensor groups. See Chapter 6 Sensor Groups.

You might want one team to manage endpoints in one region, and another team to manage endpoints in another region. Or, you might let all teams manage most endpoints but create a special team to manage the endpoints of your executive staff. You can also create teams that can view but not modify information and settings.

If a user is assigned to multiple teams with permission to access the same sensor group, and these teams have different rules, the user has the privileges of the highest role that is available from any of the teams.

Although you can assign a user to teams at any time, it is helpful to have teams set up before you create non-Global Administrator users because the capabilities of most users are determined by their teams.

Role-Based Privileges for Teams

This topic describes the roles you can assign to a team for each sensor group.

- **Analyst** – This role allows the user to monitor and respond to suspicious or malicious activity on endpoints in sensor groups for which it has the role.

  Analysts can be given additional, enhanced privileges on a per-user basis so that they can use special features. See Adding Enhanced Permissions for Analysts.

  Unless they are Global Administrators, Analysts do not have access to data or functions for managing the server itself, such as managing users and teams, viewing and changing server settings (including sharing settings), and viewing the server dashboard.

- **Viewer** – This role allows the user to access information, including suspicious or malicious activity, on endpoints in sensor groups for which it has the role.

  Unless they are Global Administrators, Viewers cannot access Live Response (Go Live), investigations, sensor isolation or file banning. They also cannot access server management functions.

- **No Access** – This role gives the user no access to information or management functions for the specified sensor groups. If the user does not have any higher role for any team, the only page available to them is My profile.

  Some access control is applied on the page level – for example, certain pages are only visible to Global Administrators or Administrators. In other cases, access control determines the data that appears on a page and the actions that can be taken there. If users enter a URL for a page they do not have permission to view, they are redirected to the HUD page.

  The following table provides more detail about privileges and access types that are available for each role.

Feature or Page	Permissions by Role
Server Dashboard	Only available to Carbon Black EDR Global Administrator or Carbon Black Hosted EDR Administrator.
Sensors	**Viewers:** Can view tables and details of sensors in sensor groups for which the user has Viewer access. **Analyst:** Can perform actions on a sensors in sensor groups for which the user has Analyst access. Additional enhanced user permissions are necessary for isolating and uninstalling sensors and using Live Response. Analysts can also move sensors between sensor groups if they are Analysts for both the source and destination sensor groups.
Sensor Groups	**Viewers:** Can view tables and details of sensor groups for which the user has Viewer access. **Analyst:** Can perform certain actions involving sensor groups for which the user has Analyst access: ■ Can toggle tamper detection if the user also has enhanced permissions for tamper levels. ■ Can toggle process banning if the user also has enhanced permissions for process banning. ■ Can edit other General, Sharing, Advanced, Event Collection, Upgrade Policy settings for the group. An Analyst cannot add or delete a sensor group.
Uninstall Sensors	**Viewers:** No Access **Analyst:** Can uninstall sensors from the console in sensor groups for which the user is an Analyst if the user also has the enhanced permission for uninstalling sensors .
Users, Teams and Activity Audit	Only available to Carbon Black EDR Global Administrator or Carbon Black Hosted EDR Administrator.
Tamper Level	**Viewer:** No Access **Analyst:** Can configure tamper settings for sensor groups for which the user is an Analyst if the user also has the enhanced permission for tamper levels .
HUD page	**Viewer:** Can view the page that is filtered to show alerts and sensors in sensor groups for which the user is a Viewer. **Analyst:** Can take action on alerts.
Threat Intel Feeds	**Viewer:** No Access **Analyst:** Can view and modify the page, including enabling and disabling actions on hit (Email Me, Create Alert, or Log to Syslog).

Feature or Page	Permissions by Role
Triage Alerts	**Viewer:** Can view all binary alerts, and can view other alerts in sensor groups for which the user is a Viewer. **Analyst:** Can view and take action on all binary alerts; can view and take action on other alerts in sensor groups for which the user is an Analyst.
Watchlists	**Viewer:** Can view watchlist results for binary searches and other searches that involve sensor groups for which the user is a Viewer. **Analyst:** In addition to view access, can add, modify, and delete watchlists, and take actions including enabling and disabling email notification, log to Syslog, and alerts.
Process Search	**Viewer and Analyst:** Can view process search results for sensor groups for which the user has at least Viewer access.
Process Analysis	**Viewer:** Can view process analysis results for sensor groups for which the user has at least Viewer access. **Analyst:** Can take actions for processes in sensor groups for which the user is an Analyst if the user also has the enhanced permission for that action. Actions include Isolate host, Go Live, and Ban Hash.
Binary Search (results) & Analysis (details)	**Viewer:** Can view all binary search results on the Search Binaries page and also details about one binary (Binary Analysis), regardless of the sensor group of the binary instance. **Analyst:** Can ban hashes in the search results if the user also has the enhanced permission to ban hashes.
Live Response	**Viewer:** No Access. **Analyst:** Can use Live Response to access and take actions on the endpoints in sensor groups for which the user is an Analyst if the user also has the enhanced permission for Live Response .
Investigations	**Viewer:** Can view the Investigations page. Actions are limited to Export events to CSV and Export timeline to PNG. **Analyst:** Can create, delete, and modify investigations.
Isolation	**Viewer:** No Access. **Analyst:** Can isolate endpoints and restore them from isolation in sensor groups for which the user is an Analyst if the user also has the enhanced permission for isolating sensors .
Banned Hashes	**Viewer:** No Access. **Analyst:** Can ban hashes and remove bans if the user has the enhanced permission for banning hashes. Not restricted by sensor group.
Notifications	**Viewer and Analyst:** All users can view notifications on the Notifications menu and receive notification emails.

Feature or Page	Permissions by Role
Sharing Settings	Only available to Carbon Black EDR Global Administrator or Carbon Black Hosted EDR Administrator.
Settings	Only available to Carbon Black EDR Global Administrator or Carbon Black Hosted EDR Administrator.
Profile info	All users can view and edit their own profile.

Adding Enhanced Permissions for Analysts

The Analyst role allows access to features for monitoring and investigation of suspicious or malicious activity on endpoints. You might allow some Analysts to take certain actions to remediate threats or vulnerabilities. Carbon Black EDR provides an interface for adding special permissions to Analysts on a per-user basis.

When enabled, these enhanced features allow a user to take action in sensor groups where the user is on a team with Analyst privileges:

Enhanced Permission	Description
Ban hashes	Can ban files by hash and remove bans. These bans are applied to all sensors.
Isolate sensor	Can isolate a sensor in that group from the network and restore the sensor from isolation. See Isolating an Endpoint.
Live Response	Can connect to and act on a sensor in that group using Live Response. See Using Live Response.
Tamper	Can set tamper level for sensor groups for which the user is an Analyst if the user also has the enhanced permission for tamper .
Uninstall sensors	Can use the console to uninstall a Carbon Black EDR sensor in the group. See also the *Carbon Black EDR Sensor Installation Guide*.
Execute Live Queries	Can run queries against endpoints. See Chapter 9 Live Query.

Note You can add enhanced Analyst permissions to any user, but these permissions are unnecessary for a Carbon Black EDR Global Administrator or Carbon Black Hosted EDR Administrator. They have no affect on users who are not on a team with the Analyst role in at least one sensor group.

Add Enhanced Permissions to Analysts

Perform the following procedure to provide enhanced Analyst permissions to a user.

The following procedures instruct you to navigate to **Users** from the navigation bar; for Carbon Black Hosted EDR administrators, navigate to **Teams** instead.

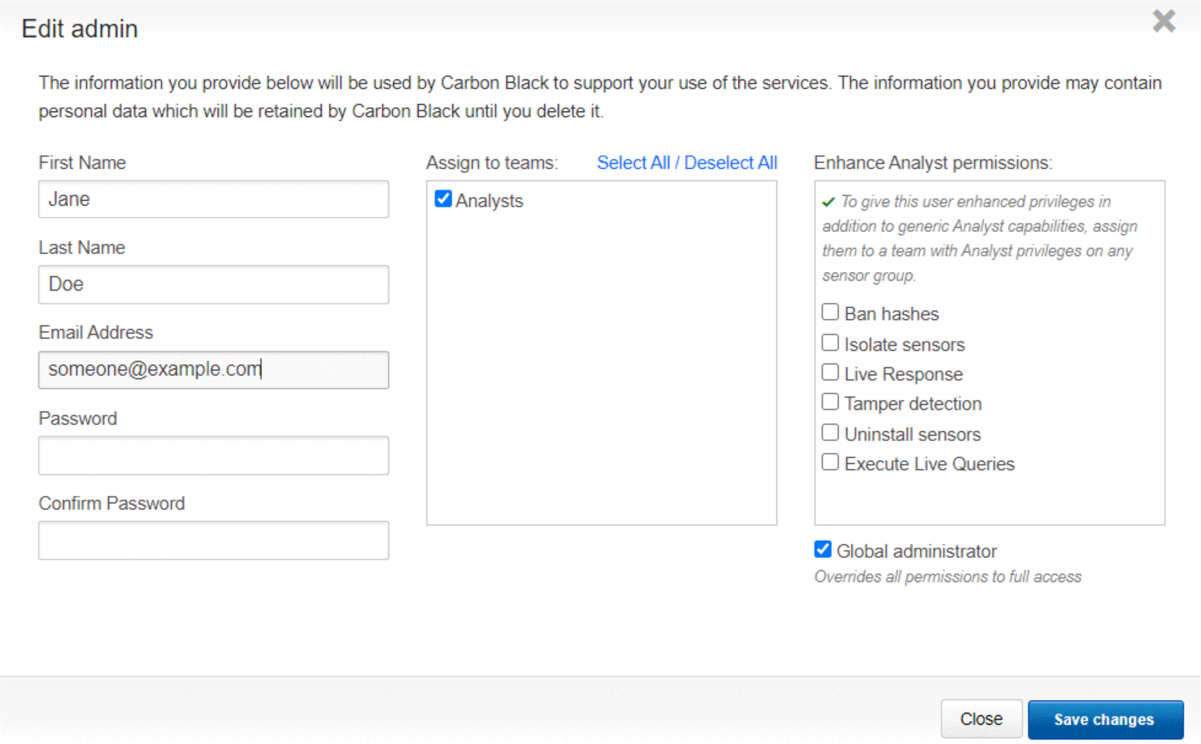
Procedure

1 On the navigation bar:

- For Carbon Black EDR, click **Users**.
- For Carbon Black Hosted EDR, click **Teams** and then click the **Users** tab.

2 Locate the user to whom to give enhanced permissions and click the **Edit user** button. If you are providing enhanced Analyst permissions for Carbon Black EDR that you have not created, use the **Add User** button and provide all necessary information.

3 In the **Enhance Analyst permissions** panel, check the box next to each permission to give this user.



Edit admin ✖

The information you provide below will be used by Carbon Black to support your use of the services. The information you provide may contain personal data which will be retained by Carbon Black until you delete it.

First Name: Jane

Last Name: Doe

Email Address: someone@example.com

Password:

Confirm Password:

Assign to teams: Select All / Deselect All

☑ Analysts

Enhance Analyst permissions:

✓ *To give this user enhanced privileges in addition to generic Analyst capabilities, assign them to a team with Analyst privileges on any sensor group.*

☐ Ban hashes
☐ Isolate sensors
☐ Live Response
☐ Tamper detection
☐ Uninstall sensors
☐ Execute Live Queries

☑ Global administrator
Overrides all permissions to full access

If the user is not a member of an Analyst team, a gray triangle icon appears in the upper left of the **Enhance Analyst permission** panel. If the user is already a member of a team with Analyst permission for a sensor group, the icon is a green checkmark.

4 If necessary, add the user to a team with Analyst permissions.

5 Click **Save**.

User/Team Permissions Example

The following example shows how you can set up user accounts and teams. This is an simplified example — not a recommendation.

Suppose that a division of your company is based in Europe, with sites in France, Germany, and Italy. Also assume that all endpoints in these countries have sensors that are managed by one Carbon Black EDR cluster.

- **Create Administrators** : You can make two Global Administrators in each country — they can set up their users and user teams. These Global Administrators can also monitor system performance and change settings that control the server. In this example, their primary responsibility is for the sensors in their own country; however, each Global Administrator can perform any Carbon Black EDR activity on the any country's endpoints if necessary.
- **Create country-specific Analysts:** Create four additional users in each country that are assigned as Analysts to teams for the sensor groups that correspond to the endpoints in their own country. They can monitor data from the sensors in these groups.
- **Enhance permissions for some Analysts:** For two of the Analysts in each country, add enhanced permissions that allow them to take actions that affect sensors.
- **Let Analysts be Viewers for other sensor groups:** So that Analysts can be aware of activities or trends that could affect all countries, give them the Viewer role for the sensor groups in countries in which they are not Analysts.
- **Create Viewer (only) users:** There might be a third, larger group of users that you assign to teams that make them Viewers for the sensor groups in their own country so they can monitor but not alter the sensors in those groups.

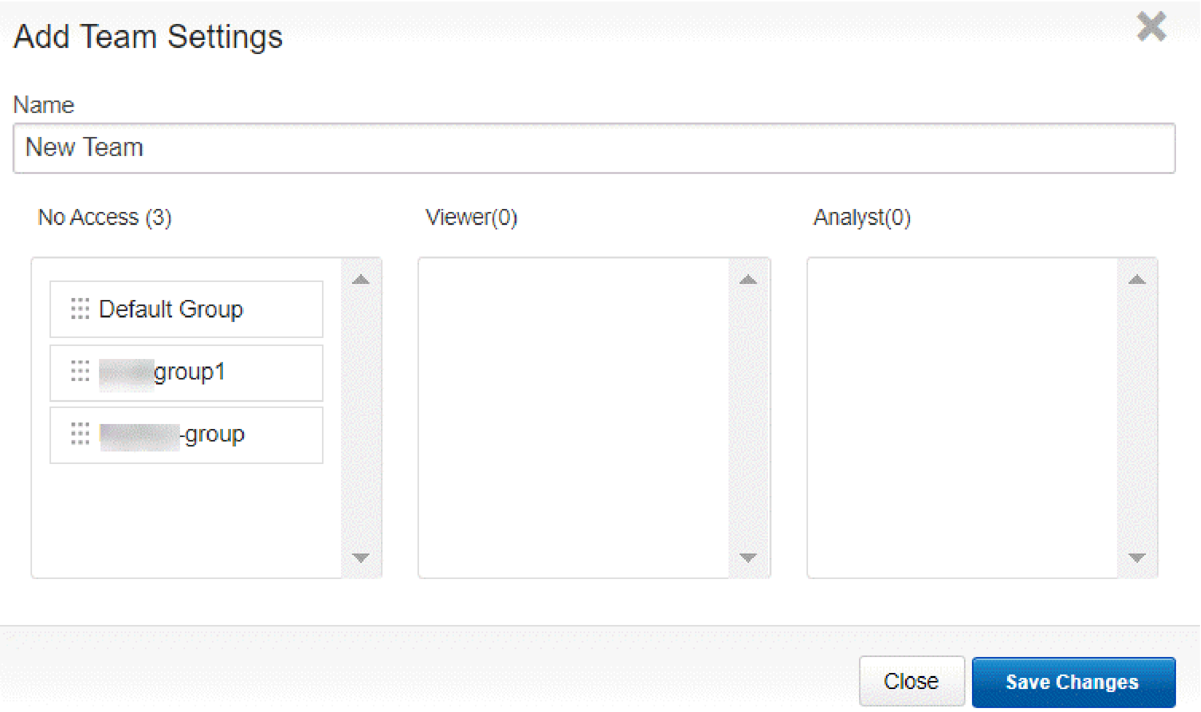
Create Teams

Use the following procedure to create teams.

Procedure

1. On the navigation bar:
   - For Carbon Black EDR, click **Users** and then click **Teams**.
   - For Carbon Black Hosted EDR, click **Teams** and then click **Teams**.

2 Click the **Create Team** button to display the **Add Team Settings** page:



Add Team Settings

Name

New Team

No Access (3)

- Default Group
- [redacted]group1
- [redacted]-group

Viewer(0)

Analyst(0)

3 In the **Name** field, enter a name for the team.

4 Drag and drop the sensor groups with the appropriate permissions. For example, for this team to have no access to the sensor group named **Default Group** , you would drag the **Default Group** box to the **No Access** list.

You can assign roles to users by adding them to teams that are set up with the type of privileges that are appropriate for the role (Analyst, Viewer, or No Access).

5 Click **Save Changes**.

Modify Teams

Use the following procedure to modify teams.

Procedure

1 On the navigation bar:

- For Carbon Black EDR, click **Users** and then click **Teams**.
- For Carbon Black Hosted EDR, click **Teams** and then click **Teams**.

2 In the list of teams, click the **Edit** icon to the far right of the team name.

3 In the **Edit Team Settings** page, modify the team settings as needed and then click **Save Changes** .

Delete Teams

When you delete a team, references to the team in user accounts are deleted, but the user accounts remain active. Use the following procedure to delete a team.

Procedure

1. On the navigation bar:
   - For Carbon Black EDR, click **Users** and then click **Teams**.
   - For Carbon Black Hosted EDR, click **Teams** and then click **Teams**.
2. In the list of teams, click the **Delete (x)** icon to the far right of the team name.
3. Click **OK** to confirm the deletion.

Managing Carbon Black EDR User Accounts

Although you can assign a user to teams later, it is helpful to have teams set up before you create non-Global Administrator users because the capabilities of most users are determine by their teams.

See Managing User Access with Teams.

Create a Carbon Black EDR User Account

To create user accounts, log in to the Carbon Black EDR server console by using an account that has Global Administrator status. If no other users have been created yet, use the administrative account and password that were established during the server installation process.

Procedure

1. From a supported web browser, enter the path to your Carbon Black EDR server. `https://<your server address>/`.
2. Enter the username and password for a Global Administrator account.
3. On the navigation bar, click **Users**.
4. Click **Add User** in the top-right corner.
5. Enter the following information and then click **Save changes**.

Field	Description
Username	Name that the user enters to log in to the console. User names are case-sensitive and restricted to standard Latin alphanumeric characters. Symbols and punctuation characters are not allowed. If you try to create a user account that contains an illegal character, the console displays a warning message.
First Name	First name of the user.
Last Name	Last name of the user.

Field	Description
Email address	Email address for the user.
Password	Password that authenticates this user. Enter any combination of letters, numbers, or special characters. Passwords are case-sensitive. This field changes to **New Password** when you are editing existing accounts.
Confirm Password	Retype the password for confirmation.
Assign to teams	Select the teams to which the user will belong. The default team is Analysts. Users can belong to more than one team. See Managing User Access with Teams.
Enhance Analyst permissions	For a user that is an Analyst on any team, check one or more boxes to give the user permission to use additional features. See Adding Enhanced Permissions for Analysts.
Global Administrator	Check this box to give the user Global Administrator privileges. **Important:** A Global Administrator has full access to all Carbon Black EDR features and data, including server management and response tools. This includes access to every endpoint with an active sensor, without needing to be assigned to teams.

Change a Carbon Black EDR User Password

Carbon Black recommends that new users change their passwords after logging in for the first time. Use the following procedure to change your password.

Procedure

1 In the Carbon Black EDR console, click ***Username*** > **My Profile** .

2 Click **Change Password** . Enter the current password, the new password, and then confirm the new password.

3 Click **Save changes** .

Note Global Administrators can change the password of any user through the User Management page.

Reset a Carbon Black EDR API Token

A unique API token is assigned to each Carbon Black EDR user. It serves as the key authentication mechanism when making calls to the APIs. Users can reset their API token at any time.

Carbon Black EDR has RESTful APIs to create custom scripts for interactions. These are described at https://developer.carbonblack.com/reference/enterprise-response/.

Note When a user's API token is reset, any affected custom scripts or integrations that use the API token must also be updated.

Procedure

1 Log in to the Carbon Black EDR console with the account whose API token will be changed.

2 Click *Username* > **My Profile**.

3 Click **API Token**.

4 Click the **Reset API Token** button.

A notification briefly appears notifying you that the API token has been reset.

Delete a Carbon Black EDR User Account

A user account can be removed from the system when that user no longer needs access to the Carbon Black EDR console.

Users who have Global Administrator privileges can delete any account except their own and the built-in administrator account. If a user with a deleted account belongs to a team, the user is automatically removed from the team when the user account is deleted.

Note A Global Administrator can delete any account except the one you are logged in as, including the administration account that was created during the server installation.

Procedure

1 On the navigation bar, click **Users**.

2 Locate the user’s name and click the **Delete (x)** icon to the far right of the user name. (The delete icon next to the currently logged in user is grayed out,because a user cannot be deleted while logged in.)

3 Click **OK** to confirm the deletion.

A popup message reports the success or failure of this action. The Activity Audit for this user also shows the delete action.

View Carbon Black EDR User Activity

Carbon Black EDR keeps an audit trail of user activity. Use the following procedure to view and export this data.

Procedure

1 On the navigation bar, click **Users**, click **Teams**, and then click **Activity Audit**.

The following information is displayed:

Field	Description
Username	The user name of the user who accessed the console.
Timestamp	The date and time that the user logged in.
Remote IP	The IP address of the computer from which the user logged in.
Request Information	The request (POST, GET, DELETE, etc.) being sent to the server.

Field	Description
Result	The HTTP response code when the user accesses a resource. For example, a successful authentication shows an HTTP 200 code response. A request to access a resource to which the user does not have permission usually results in redirection to the HUD page or displays an HTTP 403 code.
Description	The HTTP response description. For example, an HTTP 200 response shows OK, while an HTTP 403 response shows a Requires Authentication response.

2 Click **Export to CSV** to export the activity results in a CSV format with the filename `UserActivity.csv` .

Note If you have access to the Carbon Black EDR server, you can directly view the log for user activity in the following file: `/var/log/cb/coreservices/debug.log`.

Carbon Black EDR User Activity API Audit Logging

You can enable API audit logging for a server by setting `EnableExtendedApiAuditLogging=True` in the `cb.conf` configuration file (see the *Carbon Black EDR Server Configuration Guide*). In this case, Carbon Black EDR logs all REST API requests from either the console or other sources (such as scripts).

API audit log information is stored in the `/var/log/cb/audit/useractivity.log` file, and also appears as follows:

- In the User Management section of the Carbon Black EDR console, under **Request Information** on the **Activity Audit** tab.
- In a CSV file downloaded from the **Activity Audit** tab, as in the following example:

```
2017-12-22 11:30:54:  username='bill' userid='1' ip='::ffff:111.111.1.1' status='200'
method='GET' path='/api/v2/sensor'
2017-12-22 11:30:54:  username='bill' userid='1' ip='::ffff:111.111.11.1' status='200'
method='GET' path='/api/v1/alert'
2017-12-22 11:30:55:  username='bill' userid='1' ip='::ffff:111.111.11.1' status='200'
method='GET' path='/api/v1/detect/report/currentmonitoringstatus'
```

Managing User Accounts (Carbon Black Hosted EDR)

4

This section describes how to manage Carbon Black Hosted EDR user accounts.

Important Users are also affected by the user and team configurations that are described in Managing User Access with Teams.

To manage user accounts for Carbon Black EDR, see Chapter 3 Managing User Accounts (Carbon Black EDR).

Read the following topics next:

- Overview of User Management (Carbon Black Hosted EDR)
- Managing User Accounts (Carbon Black Hosted EDR)

Overview of User Management (Carbon Black Hosted EDR)

Carbon Black Hosted EDR users access their server console using a Carbon Black Hosted EDR account. User accounts allow system management professionals, threat responders, and other console users to access and manage Carbon Black Hosted EDR features.

User accounts are initiated when an administrator sends an email invitation to a new user, who can then respond to the invitation and create the account. Users can access one or more servers for which they have been authorized. In Carbon Black Hosted EDR, separate accounts are not created for each authorized server.

The capabilities of a user are determined by the following factors:

- **For which servers is the user authorized?** – The administrator who sends out an account invitation is inviting the user to create an account (if they don’t already have one) and authorize that account for a particular server.
- **Is the user an Administrator?** – The administrator who sends out an account invitation determines whether the new user will have administrator privileges. This can be changed later. If a user is an Administrator, the next three factors are not relevant.
- **What teams does the user belong to?** – The privileges of users who are not administrators depend upon the teams to which they belong. Administrators can also be assigned to teams, although team membership does not affect them unless their administrator status is disabled.

- **What roles do team members have for each sensor group?** – Teams specify a role , which determines the level of privileges their members have for each sensor group . There are three roles: **Analyst** , **Viewer**, and **No Access**. Teams can (and usually will) have different roles for different sensor groups.
- **What is the highest role for any team this user belongs to?** – Access to some features is not restricted by sensor group, but is controlled by the roles that are assigned to a team. These features become available to a user if the user is on at least one team that has a high enough role for at least one sensor group .

  This mechanism helps control access to features that are not specific to sensor groups, but to which you can restrict access. For example, threat feeds are not specific to any sensor group, but are an important tool for threat monitoring. If a user is an Analyst on any team, that user can take any of the actions available on the Threat Intelligence Feeds page.
- **Is the user an Analyst with enhanced permissions?** – For Analysts, access to sensitive features (Live Response, sensor isolation, uninstalling sensors, file banning, etc.) is controlled by supplemental enhanced permissions.

See Managing User Access with Teams.

Important Creation and management of user accounts and teams is available to Administrators only.

Managing User Accounts (Carbon Black Hosted EDR)

Carbon Black Hosted EDR users are assigned to one of two classes when their account is created.

- **Administrator** – Administrators have full privileges on the Carbon Black Hosted EDR server, including adding and removing other users.
- **User** – Users can access non-administrative functions of the Carbon Black Hosted EDR server. Access is determined by their team membership.

An administrator can authorize user access to a server in one of two ways:

- By inviting a new user through email
- By inviting an existing Carbon Black Hosted EDR user to become authorized on a new server

User invitations are created from the Users page.

Invite a User to Access a Carbon Black Hosted EDR Server

This topic explains how to invite a new user to create an account with access to a particular server, or authorize an existing Carbon Black Hosted EDR user to access a server to which they do not currently have access.

Procedure

1 In a browser, enter the URL for the Carbon Black Hosted EDR server and log in as an Administrator.

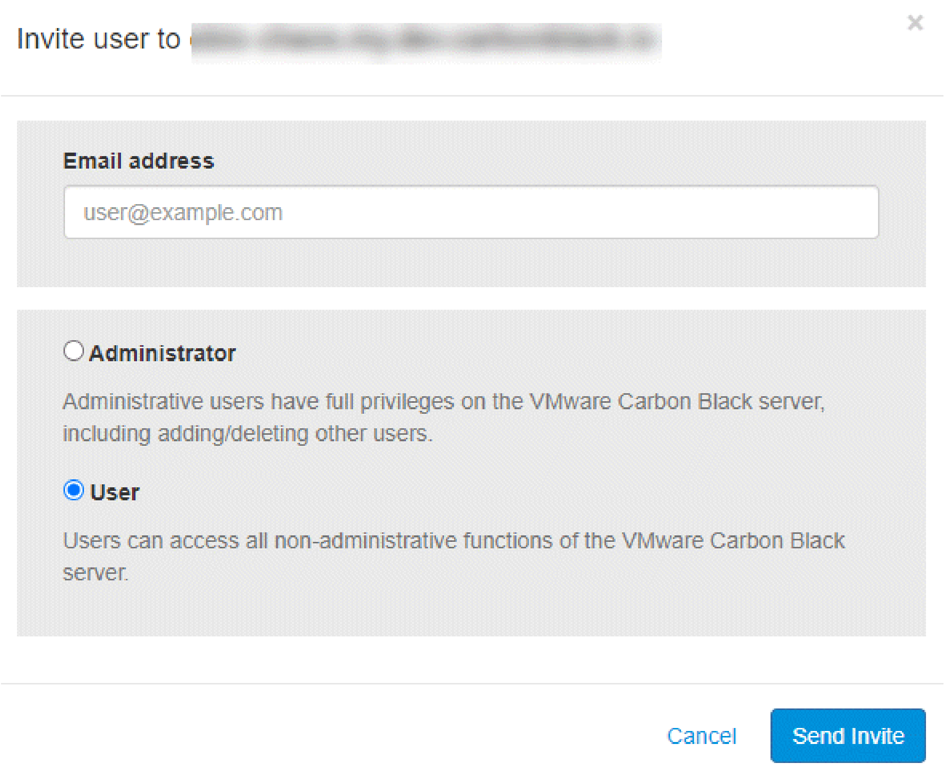
If no other administrator accounts have been created, use the administrator account that Carbon Black provided when you initiated your use of Carbon Black Hosted EDR.

2 On the navigation bar, click **Users**.

The Users page shows the Carbon Black Hosted EDR user accounts that are authorized to access the server. This page also shows users whose invitation has expired without their account being activated. If you find that the user you want to invite is listed on the page but the box for user account is grayed out, a previous email invitation to register for this server has expired. You have three options:

- Re-invite the user by double-clicking the account and clicking the **Resend Invite** button.
- Remove the invitation (making any links in the email sent to this user candidate unusable) by double-clicking the account box and clicking the **Revoke Invite** button.
- Leave the user's invitation in the expired state.

3 If the user you want to invite does not appear on the page, click **Invite user**.



Invite user to [redacted]

Email address

user@example.com

○ **Administrator**

Administrative users have full privileges on the VMware Carbon Black server, including adding/deleting other users.

● **User**

Users can access all non-administrative functions of the VMware Carbon Black server.

4 Type the email address to which to send the invitation.

5 Select **Administrator** or **User**.

Keep in mind that the Administrator role gives the user full privileges on this server. Administrators can add and delete other users.

6 Click **Send Invite**.

An email is sent to the user that contains a link to create a new account on the server. This link also authorizes an existing user who has access to a different server to log into this server by using the same account.

Note The email invitation link expires after seven days of no activity.

Activating a Carbon Black Hosted EDR Account

The invitation to activate a Carbon Black Hosted EDR account arrives in an email that is sent to the address that is provided when an administrator creates the invitation.

Activate a New Carbon Black Hosted EDR Account

Follow this procedure to active a new Carbon Black Hosted EDR account from an invitation.

Procedure

1 In the invitation email, click on the link to Carbon Black Hosted EDR.

2 In the **Create an account** section, choose a username for your account. User names are restricted to standard Latin alphanumeric characters without symbols or punctuation characters. Then enter your first and last names.

3 Click the **Sign up** button.

You are immediately logged in to the HUD page for the server you were invited to access. Confirmation that the account was created is sent to the same email address that received the initial invitation.

Access a Carbon Black Hosted EDR Server from an Existing Account

Follow this procedure to activate access to a new Carbon Black Hosted EDR server from an existing account.

Procedure

1 In the invitation email, click the link to Carbon Black Hosted EDR.

2 Under the **Already have an account?** label, click the green **Sign In** button and log in with your existing account name and password.

You are immediately logged in to the HUD page for the new server.

Access Authorized Carbon Black Hosted EDR Servers

You can view the Carbon Black Hosted EDR servers to which your account has access by using the **Cloud** option in the console.

Procedure

1 On the navigation bar, click **Users**.

2 If you have servers in different regions, choose the region (for example, U.S.) for which you want to view servers.

3 In the header area of the Users page, click the **Cloud** link.

The My Servers page lists the servers to which you have access.

4 Click a servername link to access the HUD page for that server.

5 Click the **Configure** button to manage users on that server.

Carbon Black Hosted EDR User Account Lockout

To protect against brute force login attacks, Carbon Black Hosted EDR locks a user account after seven consecutive, unsuccessful login tries within a period of 15 minutes.

An account can remain locked for up to 15 minutes. Attempts to log in during the lockout period, with or without the correct credentials, have no effect.

A user can unlock an account before the lockout expires by clicking **Forgot your password?** and following the prompts to reset the account password.

Note Carbon Black does not recommend using a group email address for a user account. For such an account:

- Any person in the group can lock the account with too many failed login attempts. In that case, none of the group members can log in during the lockout period.
- If someone unlocks the account by changing the password, all other group members must be informed of the password change.

Viewing and Modifying Carbon Black Hosted EDR User Accounts

There are several places in the console where Carbon Black Hosted EDR user information can be viewed or changed.

- **Users page** – The Users page shows each account holder, their last login, whether they have two-factor authentication enabled, and their top-level account status (User or Administrator). To access this page, click **Users** on the navigation bar.
  - Click the tile for any user to show the User Permissions for < *servername* > page, which allows an administrator to change the user from an Administrator to User, or vice versa.
  - An Administrator can also remove a user account from this page.

- **Account page** – On the Account page, individual users can manage their own account details, including:
  - Resetting their passwords
  - Enabling or disabling two-factor authorization
  - Changing the email address associated with the account
  - Editing their first and last names

  To access the Account page, click **View profile on carbonblack.io** on the My Profile page or click **Account**on the *<Username>* menu.
- **Team Management Users page** – The **Users** view on the Team Management page shows a table of all users on the current server and their teams. The **View Details** button next to a user name opens an Edit < *user* > page, where you can add or delete a user from teams. See Managing User Access with Teams for more details.
- **My Profile page** – For the currently logged-in user, the My profile page shows the user's teams, provides access to the API Token page where the API Token can be changed, and includes a **View profile on carbonblack.io** button that opens a new browser window showing the Account page for this user.

  You can access the My profile page through the *<Username>* menu in the top right of the console.

View Carbon Black Hosted EDR User Activity

Carbon Black Hosted EDR keeps an audit trail of user activity. Use the following procedure to view and export this data.

Procedure

1 On the navigation bar, click **Users** and then click **Activity Audit**.

The following information is displayed:

Field	Description
Username	The user name of the user who accessed the console.
Timestamp	The date and time that the user logged in.
Remote IP	The IP address of the computer from which the user logged in.
Request Information	The request (POST, GET, DELETE, etc.) being sent to the server.
Result	The HTTP response code when the user accesses a resource. For example, a successful authentication shows an HTTP 200 code response. A request to access a resource to which the user does not have permission usually results in redirection to the HUD page or displays an HTTP 403 code.
Description	The HTTP response description. For example, an HTTP 200 response shows OK, while an HTTP 403 response shows a Requires Authentication response.

2 Click **Export to CSV** to export the activity results in a CSV format with the filename `UserActivity.csv` .

> **Note** If you have access to the Carbon Black Hosted EDR server, you can directly view the log for user activity in the following file: `/var/log/cb/coreservices/debug.log`.

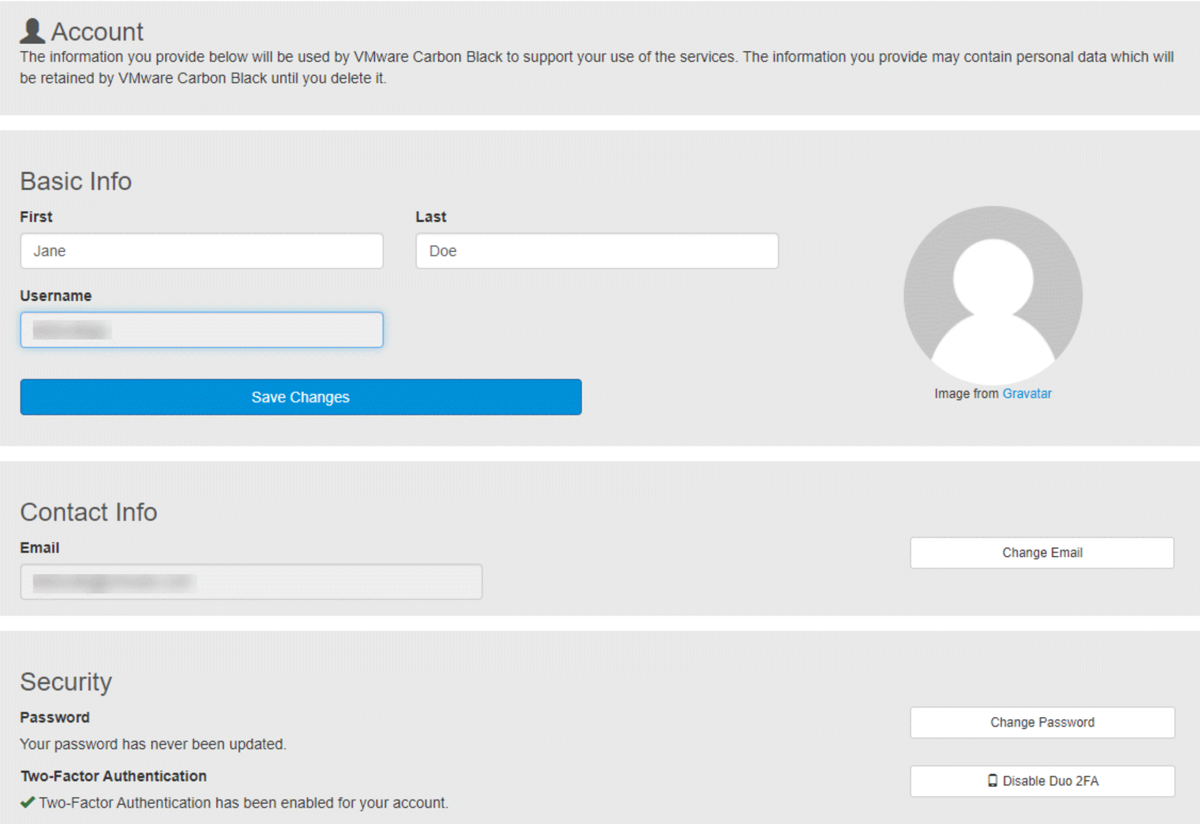
Change Carbon Black Hosted EDR User Information

Using the Account page, Carbon Black Hosted EDR users can manage their account details, including resetting their passwords and enabling or disabling two-factor authorization. Users can also change the email address associated with an account and edit their first and last names.

Procedure

1 On the navigation bar, click **Users**.

2 Click **Username > Account.** Your Account page appears:



Account

The information you provide below will be used by VMware Carbon Black to support your use of the services. The information you provide may contain personal data which will be retained by VMware Carbon Black until you delete it.

Basic Info

First: Jane

Last: Doe

Username: [redacted]

Image from Gravatar

Contact Info

Email: [redacted]

Security

Password

Your password has never been updated.

Two-Factor Authentication

✓ Two-Factor Authentication has been enabled for your account.

3 In the **Basic info** section, you can modify your first name and last name, and you can upload an image. You cannot change your username. After making your modifications, click **Save Changes**.

4 In the **Contact info** section, you can change the email address that is associated with your account by clicking **Change Email**.

In the **Change Email Address** dialog, enter the new email address and click **Change email** . You will receive an email notification to verify the new email address.

5 In the **Security** section, you can:

- **Change password** – Click to display the **Change Password** dialog, where you can change your password by entering it twice and clicking **Change password**.
- **Enroll/Disable Duo 2FA** – Click this button to enable or disable two-factor authentication. See Logging in and Configuring Two-Factor Authentication.

Change Carbon Black Hosted EDR Administrator / User Status

You can change a user's status to either an administrator or a non-administrative user. Only administrators can perform this task.

Procedure

1 On the navigation bar, click **Users**.

The Users page appears and shows the user accounts that are authorized to access the server.

2 Click the user account to modify.

The User Permissions for *<server name>* page appears for that user account.

3 To change the user account status, select an option:

- **Administrator** – Administrators are users with full privileges on the Carbon Black Hosted EDR server, including adding/removing other users.
- **User** – Users that are not administrators can access all non-administrative functions of the Carbon Black Hosted EDR server.

4 Click **Save Changes**.

Reset a Carbon Black Hosted EDR API Token

A unique API token is assigned to each Carbon Black Hosted EDR user. It serves as the key authentication mechanism for making calls to the APIs. A user can reset their API token at any time.

Carbon Black Hosted EDR has RESTful APIs that can be used to create custom scripts for interactions with its features. These are described at https://developer.carbonblack.com/reference/enterprise-response/.

Note When a user's API token is reset, any affected custom scripts or integrations that use the API token must also be updated.

Procedure

1 On the navigation bar, click **Username > My Profile**.

2 In the **My Profile** window, click **API Token**.

3 Click the **Reset API Token** button.

Delete a Carbon Black Hosted EDR User Account

You can remove a user account from accessing the Carbon Black Hosted EDR server, thereby terminating access for that account.

Procedure

1 On the navigation bar, click **Users**.

The Users page displays the user accounts that are authorized to access the server.

2 Click the user account to remove.

The User Permissions for *<server name>* page appears for that user.

3 Click **Remove User.**

Managing Sensors

5

This section describes how Carbon Black EDR sensors work, the information they provide, and how to search for and monitor sensors.

- See Managing User Access with Teams for information about the roles and permissions that are required to view and modify sensors and their information.
- See the *Carbon Black EDR Sensor Installation Guide* for information on installing, upgrading, troubleshooting, and uninstalling sensors.
- See Chapter 6 Sensor Groups for information on managing sensor groups.
- See Chapter 7 Managing Certificates for information about certificate options.
- See Chapter 23 Sensor Parity for information on features supported on sensor operating systems.

Read the following topics next:

Overview of Sensor Management

Installed sensors gather event data on host computers (endpoints) and securely deliver the data to the Carbon Black EDR server for storage and indexing. This enables your team to see and understand the history of an attack, even if the attacker deleted artifacts of its presence.

A sensor checks in with the Carbon Black EDR server every five minutes to report the activity that it detects. The server responds and notifies the sensor about how much data to send. To aid in detecting IOCs, the server compares the data it records from sensors with the latest data that is synchronized from the threat intelligence feed partners that you have enabled.

We recommend you employ Network Time Protocol (NTP) on the sensors and the Carbon Black EDR server. Carbon Black EDR does not have a technical requirement to maintain coordinated time between sensors and servers, but event correlation depends on a common understanding of when things occurred in time to determine if the events are strongly coincidental and therefore likely to be related.

Employing NTP ensures that the times reported by the various sensors coincide with the time as understood by the Carbon Black EDR server. In this way, queries executed on the Carbon Black EDR server can present relevant, related events in a manner that analysts can readily correlate. Additionally, it ensures remote processing systems like SIEMs can perform the same time-based event correlation. Accurate time keeping through use of NTP or NTP-like services is essential for the proper operation of Carbon Black EDR.

Each sensor belongs to a sensor group that defines the configuration and security characteristics for the sensor. For example, sensor groups define the upgrade policy and types of event information that sensors in the group collect. One sensor group can contain many sensors, but a single sensor can only belong to one sensor group. See Chapter 6 Sensor Groups for more information.

To secure communication between sensors and the server, Carbon Black EDR uses HTTPS and TLS. You can use the default server certificate or add your own server certificates and assign different certificates to different sensor groups. See Chapter 7 Managing Certificates for details.

Collected Data Types

Sensors collect information about the following data types:

- Currently running parent and child processes
- (macOS and Linux only) Fork and posix_exec processes
- Modules loaded by processes
- Processes blocked as the result of a Carbon Black EDR hash ban
- Binaries
- File executions
- File modifications
- Network connections
- (Windows only) Registry modifications
- (Windows only) Cross-processes (an occurrence of a process that crosses the security boundary of another process)
- (Windows only) Enhanced Mitigation Experience Toolkit (EMET) events and configuration

Incident-Response Features

To help you manage sensors and work with the information they capture, Carbon Black EDR provides incident-response features that provide the following capabilities:

- Directly respond to a threat detected on an endpoint through a command interface
- Isolate an endpoint with a suspicious process or threat
- Ban process hashes to prevent known malware from running in the future
- Set watchlists to monitor suspicious activity on endpoints

For information on these incident-response features, see Chapter 8 Responding to Endpoint Incidents and Chapter 19 Watchlists

Tamper Protection of Windows Sensors

The 7.2.0-win sensor release includes a Tamper Protection feature that protects the Carbon Black EDR Windows sensor against external attempts to stop Carbon Black EDR services, or to modify the sensor's binaries, disk artifacts, or configuration.

While in a Tamper Protected state, the sensor only accepts actions that are requested through the Carbon Black EDR server console.

We encourage you to review the knowledge base article EDR: Which Sensor directories need exclusion from third-party anti-virus scans to make sure that the latest Carbon Black EDR Windows sensor exclusions are in place before enabling Tamper Protection.

Apply Tamper Protection to a Sensor Group

Follow this procedure to apply Tamper Protection to a sensor group.

Prerequisites

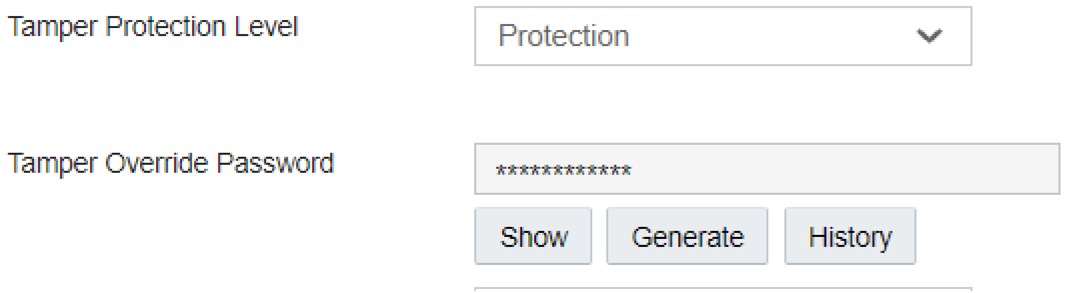
Requirements:

- Minimum OS Versions of Windows 10 v1703 (Desktop) or Windows Server v1709 (Windows build 15163)
- Minimum Carbon Black EDR versions of v7.2.0 Windows sensor and v7.4.0 Carbon Black EDR Server
- You must be one of the following: a Global Administrator (Carbon Black EDR), an Administrator (Carbon Black Hosted EDR), or a user who is an Analyst for the applicable sensor group and who also has permission for Tamper Level.

Procedure

1. On the navigation bar, click **Sensors**.
2. Click the gear icon next to the sensor group for which to apply Tamper Protection.
3. In the **Edit Group** panel, click **Advanced**.

4 Change the **Tamper Protection Level** to **Protection** and create a **Tamper Override Password**.



Tamper Protection Level Protection

Tamper Override Password *************

5 Click **Save Group**.

For more information about this setting, see Advanced Settings.

- Any Windows sensor in a sensor group that has Tamper Protection applied and that does not meet the minimum OS requirements will default to *Tamper Detection*. Carbon Black App Control Tamper Protection is recommended in these cases. We recommend that you update the tamper rule settings for Carbon Black App Control to the latest Carbon Black EDR Tamper Protection Rapid Config.
- Enabling Tamper Protection on both Carbon Black App Control and Carbon Black EDR does not provide extra protection. We recommend that you disable the Carbon Black App Control "Carbon Black EDR Tamper Protection" Rapid Config after Carbon Black EDR Tamper Protection enforcement is in place.

Use the Tamper Protection CLI Tool

In case of disrupted communication between the Carbon Black EDR server and the sensor, you can manage the sensor directly by using the `CbEDRCLI` tool.

Procedure

1 Run `CbEDRCLI.exe` as an admin on the Windows endpoint.

2 Enter the tamper override password. Note that three incorrect password attempts incur a lockout period of one minute.

The `CbEDRCLI` tool will be effective for one hour.

Monitoring Sensor Status and Activity

The Carbon Black EDR console provides multiple views into sensor activity on your endpoints.

- On the HUD page, the Sensors panel gives a snapshot of sensor health, status, and activity.
- On the Sensors page, you can search for sensors and manage sensor groups.
- From anywhere in the console (Process Search or Watchlists pages, for example), you can click a hostname to get detailed information about a particular sensor.

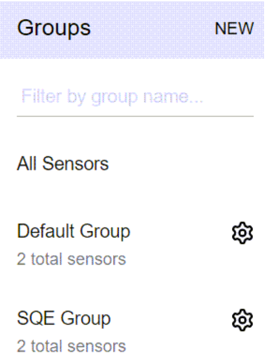
Getting Started using the Sensor Page

The Sensors page in the Carbon Black EDR console provides information about sensors and their host computers (endpoints).

- On the navigation bar, click **Sensors**.

Groups Panel

The **Groups** panel on the left side of the Sensors page displays the sensor groups in which the sensors belong:



Groups NEW

Filter by group name...

All Sensors

Default Group
2 total sensors

SQE Group
2 total sensors

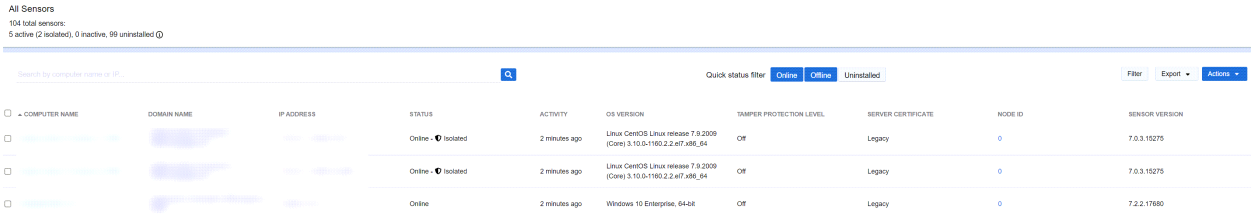
You can filter the groups that display by group name.

Click the name of the group in the **Groups** panel to view all sensors in a particular sensor group. The group name appears at the top of the **Sensors** panel.

Click **All Sensors** at the top of the **Groups** panel to view all sensors in all groups.

Sensors panel

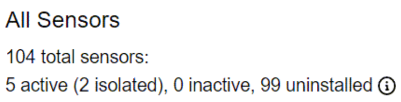
Computers (endpoints or hosts) that have installed or uninstalled Carbon Black EDR sensors display in the **Sensors** panel:



All Sensors
104 total sensors:
5 active (2 isolated), 0 inactive, 99 uninstalled ⓘ

□	▲ COMPUTER NAME	DOMAIN NAME	IP ADDRESS	STATUS	ACTIVITY	OS VERSION	TAMPER PROTECTION LEVEL	SERVER CERTIFICATE	NODE ID	SENSOR VERSION
□	[redacted]	[redacted]	[redacted]	Online - Isolated	2 minutes ago	Linux CentOS Linux release 7.9.2009 (Core) 3.10.0-1160.2.2.el7.x86_64	Off	Legacy	0	7.0.3.15275
□	[redacted]	[redacted]	[redacted]	Online - Isolated	2 minutes ago	Linux CentOS Linux release 7.9.2009 (Core) 3.10.0-1160.2.2.el7.x86_64	Off	Legacy	0	7.0.3.15275
□	[redacted]	[redacted]	[redacted]	Online	2 minutes ago	Windows 10 Enterprise, 64-bit	Off	Legacy	0	7.2.2.17680

In the top left of the **Sensors** panel, a count of all sensors per state is displayed:



All Sensors
104 total sensors:
5 active (2 isolated), 0 inactive, 99 uninstalled ⓘ

The following information appears for all sensors on the page:

Field	Description
Computer Name	The hostname corresponding to the endpoint on which the sensor is installed.
Domain Name	The registered DNS name for the IP address of the endpoint on which the sensor is installed.
IP Address	The IP address of the endpoint on which the sensor is installed.
Status	Describes the status of sensor connectivity as follows: ■ Online – Sensor communicated with the server within the previous expected check-in interval. ■ Offline – Sensor did not communicate with the server for more than a five-minute period after the expected check-in interval provided during the previous check-in. If known, offline status might include one of the following reasons: ■ Offline (Suspended) – Sensor detected that an OS-level suspend operation occurred before the sensor went offline. ■ Offline (Restarting) – Sensor detected that an OS-level restart operation occurred before the sensor went offline. ■ Offline (Isolate configured) – Sensor is offline and marked for isolation upon next check-in. ■ Offline Uninstalled – Appears when an uninstall was requested for an offline sensor. If a sensor is being uninstalled, one of the following status descriptions might appear: ■ Uninstall uninstalled – Requested uninstall operation has completed, and the sensor was successfully uninstalled. ■ Uninstall pending uninstalled – Uninstall operation was requested but has not yet completed.
Activity	The time that updated data is expected from the sensor; for example, “Was expected 2 seconds ago” or “Last seen about 3 months ago.”
OS Version	The operating system version of the endpoint on which the sensor is installed.
Server Certificate	The server certificate being used to secure communications with this sensor.
Node Id	In a clustered environment, the server ID to which a sensor sends data. For a standalone instance, the value is 0 (zero).
Sensor Version	Version of the currently installed Carbon Black EDR sensor.

Define the List of Displayed Sensors

- Click the **Online**, **Offline**, and/or **Uninstall** buttons to display matching sensors.

  Note The `SensorLookupInactiveFilterDays` setting determines whether sensors that have not checked in for a specified number of days are filtered out of the Sensors page.

  This setting has no effect when set to the default value of zero. When set to any value greater than zero, the Sensors page filters out any sensors that have not been checked in during the specified past number of days. This setting filters the results of the API call `GET /api/v1/sensor` .

  This setting interacts with the setting for `MaxEventStoreDays`, which controls how old warm (mounted) partitions can become before they are unmounted or deleted. If `SensorLookupInactiveFilterDays` is not zero but less than the value of `MaxEventStoreDays` (30 days by default), process data for inactive computers is included in search results.

- To search for one or more sensors, see Search for Sensors.

Navigation

When the defined list of sensors cannot fit on a single page, use the controls at the bottom of the page to navigate through multiple pages:

- Enter the number of **Items per Page**.
- Enter a number to **Jump to a Page**.
- Click the forward and back arrows to navigate pages sequentially.
- Click a number between the arrows to go to a specific page.

Search for Sensors

On the Sensors page, you can search for sensors using either the **Search** box, or a search based on filtered criteria.

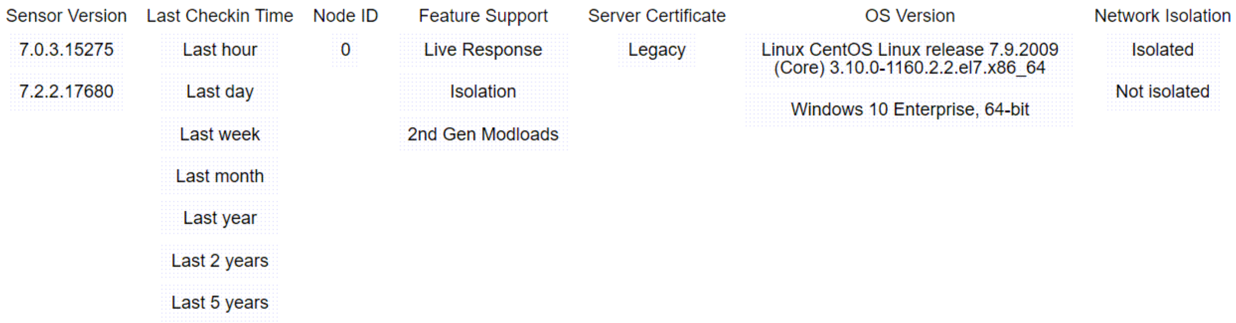
Procedure

1 On the Sensors page, do one of the following:

  - In the **Search** box, enter characters in the name of the endpoints to find. Searching commences incrementally as you type, and is case-insensitive.

    Search results include computers with installed (or with the **Uninstalled** quick filter selected, uninstalled) sensors that match the search criteria that one or more selected filters specifies

  - Click **Filter** and select any of the following criteria:



Sensor Version	Last Checkin Time	Node ID	Feature Support	Server Certificate	OS Version	Network Isolation
7.0.3.15275	Last hour	0	Live Response	Legacy	Linux CentOS Linux release 7.9.2009 (Core) 3.10.0-1160.2.2.el7.x86_64	Isolated
7.2.2.17680	Last day		Isolation		Windows 10 Enterprise, 64-bit	Not isolated
	Last week		2nd Gen Modloads			
	Last month					
	Last year					
	Last 2 years					
	Last 5 years					

Filter Criteria	Description
Sensor Version	Installed version of a sensor.
Last Checkin Time	Timespan in which a sensor last checked into the Carbon Black EDR server (last hour, last day, last week, and so on).
Node ID	In a clustered environment, the server ID to which a sensor sends data. For a standalone instance, the value is 0 (zero).

Filter Criteria	Description
Feature Support	One of the following features a sensor reports as supporting: ■ Live Response – CBLR ■ Isolation – The sensor can be isolated ■ 2nd Gen Modloads – (macOS only) Binary modules the sensor reports as being loaded by a process
Server Certificate	The server certificate that is being used to secure communications with the sensor. The certificate is assigned to the sensor group and gets applied to all sensors belonging to that group. See Chapter 7 Managing Certificates.
OS Version	The operating system version of the endpoint on which the sensor is installed.
Network Isolation	Isolation state of the sensor; this is either Isolated or Not isolated.

The list of sensors updates dynamically according to the filters selected.

Search results include computers with installed, or with the **Uninstalled** quick filter selected, uninstalled sensors that match the search criteria that one or more selected filters specifies.

2 To clear all filters and search-box criteria and reset the Sensors page to an unfiltered list of sensors, click **Reset Filters**.

Exporting Sensor Data

From the Sensors page, you can download detailed sensor data to a CSV file.

From the **Export** drop-down list on the Sensors page, click one of the following options:

- **Export All** – Downloads data either for all installed sensors, or with the **Uninstalled** quick filter selected, for both installed and uninstalled sensors on endpoints in your environment.
- **Export Visible** – Downloads data only for sensors that are visible on the current page. For example, if there are 40 sensors in a list, and the list displays 20 items per page, the CSV file only contains data for the 20 sensors that are currently visible.

Sensor Actions

On the Sensors page, you can select sensors by selecting the check boxes next to the sensor names. Use the **Actions** drop-down list to perform the following actions on one or more selected sensors.

- **Sync** – Forces the sensor to send all the data that it has collected to the Carbon Black EDR server immediately, ignoring any bandwidth throttles that might be configured.
- **Restart** – Restarts the sensor process.
- **Move to group** – Moves the sensor to another sensor group.
- **Uninstall** – Uninstalls the sensor from the host computer.

- **Isolate** – Isolates an endpoint from the rest of the network, leaving only the connections necessary for the Carbon Black EDR server to access its sensor. The action presents an optional description text box where you can note the reason for the activity. The console provides the following cues for an isolated host:
  - On the Sensors page, the word **Isolated** appears in the **Status** column.
  - On the Sensor Details page, the word **Isolated** appears with the sensor status. A **Remove host isolation** button is available next to the **Actions** button.
  - On the Process Analysis page (if a process belonging to the isolated host is being analyzed), the message “This host has been isolated from the rest of the network” appears at the top, and a **Remove host isolation** button is available beside the **Actions** button.

    See Isolating an Endpoint.
- **Remove isolation** – From an isolated state, rejoins an endpoint to the network. The action presents an optional description text box where you can note the reason for the activity.

Monitoring Sensor and Server Information

This topic describes how to view this information in the Server Dashboard, and the details that display there.

The Server Dashboard provides an overview of the following sensor and server details:

- Sensor statistics
- Server communication status
- License information

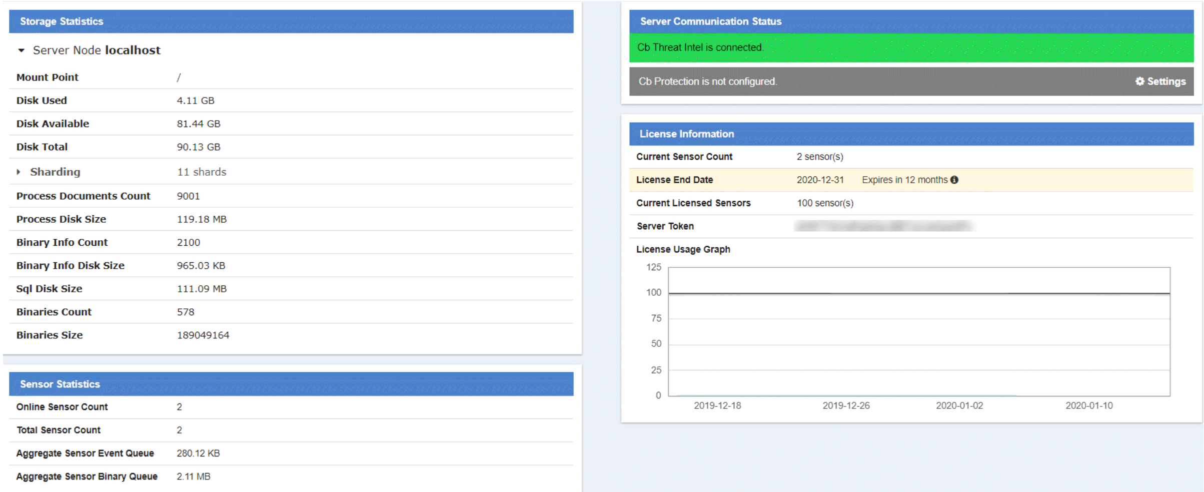
Note The Server Dashboard is only available to Carbon Black EDR Global Administrators and Carbon Black Hosted EDR Administrators.

View Server and Sensor Information in the Server Dashboard

Perform the following procedure to view sensor and server information in the Server Dashboard.

Procedure

1 On the navigation bar, click **Server Dashboard**.



Storage Statistics

▾ Server Node **localhost**

Mount Point	/
Disk Used	4.11 GB
Disk Available	81.44 GB
Disk Total	90.13 GB
▸ Sharding	11 shards
Process Documents Count	9001
Process Disk Size	119.18 MB
Binary Info Count	2100
Binary Info Disk Size	965.03 KB
Sql Disk Size	111.09 MB
Binaries Count	578
Binaries Size	189049164

Sensor Statistics

Online Sensor Count	2
Total Sensor Count	2
Aggregate Sensor Event Queue	280.12 KB
Aggregate Sensor Binary Queue	2.11 MB

Server Communication Status

Cb Threat Intel is connected.

Cb Protection is not configured.

License Information

Current Sensor Count	2 sensor(s)
License End Date	2020-12-31 Expires in 12 months ℹ
Current Licensed Sensors	100 sensor(s)
Server Token	[redacted]

License Usage Graph

2 Review information in the **Storage Statistic** panel:

Field	Description
Mount Point	The mount point for the Carbon Black EDR server data directory.
Disk Used	The amount of the disk space used for storage.
Disk Available	The amount of the disk space still available for storage.
Disk Total	The total amount of disk space.
Sharding	The number of shards on the disk. Expand to see associated ID, size, document count, and max document count.
Process Documents Count	The number of process documents uploaded to the database on the server. This is the same number as the total number of processes on the Process Search page .
Process Disk Size	The disk space used by the process documents.
Binary Info Count	The number of binaries that are seen by the sensor. This is the same number as the total number of binaries on the Binary Search page .
Binary Info Disk Size	The total number of bytes of binary information that is uploaded to the server.
Sql Disk Size	The psql database disk utilization.
Binaries Count	The number of binaries stored on the server.
Binaries Size	The total number of bytes of binaries stored on the server.

3 Review information in the **Sensor Statistics** panel:

Field	Description
Online Sensor Count	The number of sensors that are detected as being online by the server.
Total Sensor Count	The total number of sensors installed and registered for this server.
Aggregate Sensor Event Queue	The total size of queued events needing to be pushed to the server for all online sensors.
Aggregate Sensor Binary Queue	The total size of queued binaries needing to be pushed to the server for all online sensors.

4 Review information in the **Server Communication Status** panel:

Field	Description
Carbon Black Threat Intel is connected	Shows whether or not communication between the Carbon Black EDR server and Carbon Black Threat Intel has been established.
Carbon Black App Control is not configured	Shows whether or not communication between the Carbon Black EDR server and a Carbon Black App Control server has been established.

5 Review the information in the **License Information** panel:

Field	Description
Current Sensor Count	Total number of unique online sensors in the last 24 hours (active).
License End Date	The date when the license terminates.
Current Licensed Sensors	The total number of sensors on the current license.
Server Token	The token for the Carbon Black EDR server. This token is primarily used for support purposes.
License Usage Graph	A weekly depiction of how many sensors are active for the license.

Viewing Sensor Details

The Sensor Details page provides detailed information about each sensor.

To view sensor details:

- From the Process Search page, click anywhere within a row to open the Process Preview page. After opening Process Preview, click the **Hostname** link.
- From the Process Search page, HUD Sensors widget, or Sensors page, click the name of the endpoint.

Sensor Details Heading and Options

The heading in the **Sensor Details** panel displays the following information and options.

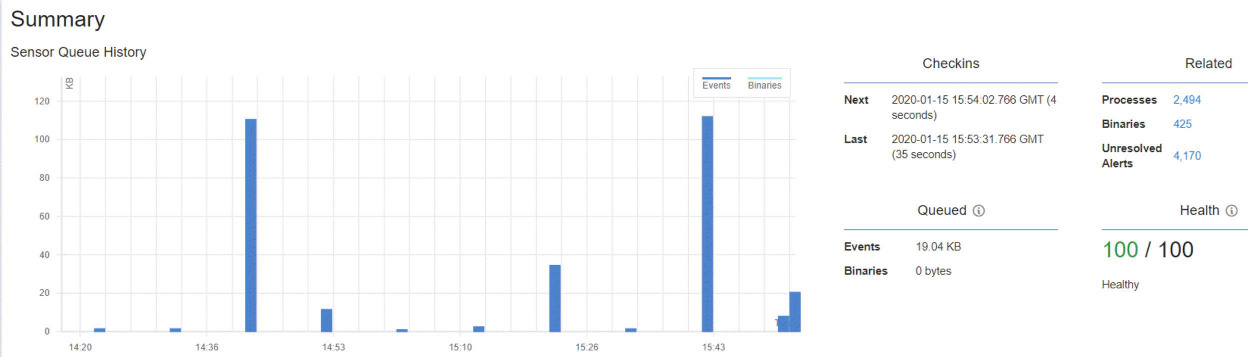
- An icon that represents the operating system of the host computer.
- The name of the host computer.
- Status of the sensor. This can be **Online**, **Offline**, **Uninstalled**, and whether the sensor is restarting, isolated, or syncing. If Carbon Black EDR is not running or if there is a communication problem, the status is **Offline.**
- A **Go Live** button starts a Live Response session with the endpoint if Live Response is enabled.

  Note To use Live Response, you must be a Carbon Black EDR Global Administrator, a Carbon Black Hosted EDR Administrator, or a user on a team that has Analyst privileges.

- **Isolates**: Isolates a host computer from the rest of the network, leaving only the connections to access the Carbon Black EDR server. If the sensor is isolated, this button displays **Remove isolation**. The action presents an optional **Description** text box where you can note the reason for the activity. See Isolating an Endpoint.
- The **Actions** menu provides the following options:
  - **Sync** – Forces the sensor to immediately send all the data that it has collected to the Carbon Black EDR server, ignoring any configured bandwidth throttles.
  - **Restart** – Restarts the sensor process.
  - **Move to group** – Moves the sensor to another sensor group.
  - **Uninstall** – Uninstalls the sensor from the host computer.

Sensor Details Summary

This topic describes the Sensor Details page **Summary** panel.



Checkins

Next 2020-01-15 15:54:02.766 GMT (4 seconds)

Last 2020-01-15 15:53:31.766 GMT (35 seconds)

Related

Processes 2,494

Binaries 425

Unresolved Alerts 4,170

Queued ⓘ

Events 19.04 KB

Binaries 0 bytes

Health ⓘ

100 / 100

Healthy

The **Summary** panel displays the following information about the sensor's health and activity:

Item	Description
Sensor Queue History	A graph that displays the recent history of activity in the sensor's queue. Click **View Details** to view this data in tabular form.
Checkins	Shows the **Next** and **Last** times (GMT) that the sensor checked in with the Carbon Black EDR server.

Item	Description
Related	Processes, binaries, and alerts that are associated with the host.
Queued	Events and binaries that the sensor has not yet sent to the Carbon Black EDR server. The maximum queue size can be managed in Sensor Group Settings.
Health	Sensor health score, and a health message from the sensor if there is one. See Chapter 24 Sensor Health Score Messages.

Vitals and Configuration

This section describes the **Vitals and Configuration** panel on the Sensor Details page.

Sensor Vitals

The **Sensor Vitals** section of the **Vitals and Configuration** panel displays the following information.

Field	Description
Sensor Id	The internal Carbon Black EDR sensor GUID of the host computer.
Node Id	The server ID to which the sensor sends data in a clustered environment.
Node Address	The address of the server to which the sensor sends data in a clustered environment.
Node Hostname	The host name of the server to which the sensor sends data in a clustered environment.
Shard Id	The shard ID to which the sensor submits event and binary metadata.
Registration Time	The date and time that the sensor registered (the start time of the sensor) with the Carbon Black EDR server.
Sync-mode	Shows if the sensor is currently synchronizing with the server.
Restart Pending	Shows if the sensor host is restarting.
Uninstall Pending	Shows if the sensor is being uninstalled from the host.
Sensor Version	The current version of the sensor.
Sensor Uptime	The time since the Carbon Black service started. **Note** ■ Because the `Host Uptime` value might not increase when the endpoint is asleep or hibernating, the `Sensor Uptime` value can be higher than `Host Uptime`. ■ If the `Sensor Uptime` is consistently a low number, it can indicate an issue with service stability.
Network Isolation	Shows if the host is isolated from the rest of your network and the Internet. See Isolating an Endpoint.

Computer Vitals

The **Computer Vitals** section of the **Vitals and Configuration** panel shows the following details about the host.

Field	Description
Hostname	Hostname of the endpoint on which the sensor is installed.
OS Version	Version of the operating system on the host.
IP Address/MAC Info	IP and MAC address of the host.
Computer Domain Name	Name of the endpoint on which the sensor is installed. This can be the same name as the hostname.
Computer SID	Unique security identifier for the computer.
Amount of RAM	Amount of available RAM.
Free Disk Space	Amount of free disk space.
Total Disk Space	Amount of total disk space.
Host Uptime	The time that the operating system has been running as reported by the operating system itself. **Note** This number might not increase when the endpoint is asleep or hibernating.
Power State	Indicates whether the host endpoint is running.
Clock Delta	The difference between the sensor clock and the server clock. If the delta is greater than 5 seconds, an alert is displayed.

Configuration

The **Configuration** section of the **Vitals and Configuration** panel shows the following sensor information.

Field	Description
Group	Sensor group to which this sensor belongs.
Site	Site to which the sensor group that contains this sensor belongs. For information about assigning sensor groups to sites, see General Settings.
Server Name	URL of the Carbon Black EDR server.
Sensor Upgrade Policy	Upgrade policy for this sensor. See Upgrade Policy Settings for details of sensor group upgrade policies.
EP Agent Installed	Indicates whether the Carbon Black App Control agent is installed.
EP Agent Host Id	If the Carbon Black App Control agent is installed, displays the Carbon Black App Control agent host Id.

Team Access

The **Team Access** section of the **Vitals and Configuration** panel shows the teams that have access to this sensor and the permissions that team members have. This information is defined in **Sensors** > **Edit Settings** > **Permissions**.

See Permissions Settings.

Recent Activity

This topic describes the **Recent Activity** panel on the Sensor Details page.

The **Activity** section of the **Recent Activity** panel shows the activity types that the sensor has been engaged in, and the date and time of each activity. This data is applicable for the duration that the sensor has been up and running.

The **Resource Status** section of the **Recent Activity** panel shows information for tracking internal performance metrics for sensors. If the performance of a sensor is degrading, the values in this table can help diagnose the cause. The panel shows the following details:

Field	Description
Timestamp	The date and time in one-hour intervals for which the sensor resource status is tracked.
Page Faults	The number of page faults that occurred on the date and time in the `Timestamp` field.
Commit Charge	The total memory used by all applications on the host endpoint, including memory that has been temporarily paged to disk at the date and time that is displayed in the `Timestamp` field.
Handles	The number of handles in use at the date and time displayed in the `Timestamp` field.

Sensor Diagnostics

This topic describes sensor diagnostics data on the Sensor Details page. For additional sensor troubleshooting information, see the *Carbon Black EDR Sensor Installation Guide*.

Communication Failures

The **Communication Failures** section of the **Diagnostics** panel shows the timestamp and failure code of communication failures between the sensor and the server.

You can locate the correct failure code and cross-reference it with the information provided at https://curl.haxx.se/libcurl/c/libcurl-errors.html. For example, if you see error code *0x80c80013*, locate 13 on this page.

Driver Diagnostics

The **Driver Diagnostics** section of the **Diagnostics** panel shows diagnostic information about the sensor driver.

Carbon Black EDR macOS sensors have the following components:

- **CbSystemProxy** – A core kernel driver that improves interoperability with third-party products. When the macOS sensor is uninstalled, the next two kernel drivers are immediately removed and unloaded. The core kernel driver remains until the system reboots. Immediately unloading the core kernel driver can cause system instability if other products (typically security) are running in the system that integrate in the same way as Carbon Black EDR.
- **CbOsxSensorProcmon** – A kernel driver to capture all other events on macOS 10.15 and earlier.
- **CbOsxSensorNetmon** – A kernel driver to capture network events on macOS 10.15 and earlier.
- **CbOsxSensorService** – A user-mode service to communicate with the Carbon Black EDR server.
- **es-loader.es-extension** – A user-mode driver to capture all events on macOS 11.0 and later.

Carbon Black EDR Windows sensors have the following components:

- **CoreDriver**
  - For Windows XP/2003/Vista/2008 (Vista server version), the driver binary name is `carbonblackk.sys` .
  - For Windows 7 and later, the binary name is `cbk7.sys` .
  - In all cases, the core driver is a mini-filter driver with the service name `carbonblackk` .
  - The core driver captures all events except for network connection events and passes all events, except tamper detection events, to the user-mode service.
  - The core driver attempts to directly send Tamper detection events to the Carbon Black EDR server. If this fails, then the core driver attempts to send the Tamper detection events to the user-mode service.
- **Network Filter Driver**
  - For Windows XP/2003, the network filter driver is a Transport Driver Interface (TDI) filter driver with the binary name `cbtdiflt.sys` and service name `cbtdiflt` .
  - For Windows Vista and later, the network filter driver is a Windows Filter Platform (WFP) driver with the binary name `cbstream.sys` and service name `cbstream` .
  - The network filter driver is responsible for collecting network connection events and implementing the network isolation feature of the Windows sensor.
- **User-mode Service**
  - The sensor uses a user-mode service with the binary name `cb.exe` and service name `CarbonBlack` .
  - This service communicates with the core and network filter drivers to gather and process events from the kernel and send those to the server.

Carbon Black EDR Linux sensors have the following components:

- **Kernel Module –** This module does the following:
  - Uses a binary named `cbsensor.ko.<kernel version>` where the `<kernel version>` is a currently supported kernel.
  - Captures all system events and makes them available to the user mode daemon to process.
  - Exposes performance statistics in the `/proc/cb` directory.
- **User Mode Daemon** – This user-mode daemon uses a binary named `cbdaemon` . This service communicates with the kernel module to gather and process events to be sent to the server.

The sensor starts recording activity as soon as the core driver is loaded. It queues up the activity for the user mode service to receive as soon as it starts. This occurs early in the sensor boot process.

The network driver is loaded after the core driver, but it also starts recording as soon as it is loaded, and it also queues events for the user mode service.

These kernel driver components usually work in sync with each other, but it is possible for the sensor to be communicating with the server while one of the drivers is inoperable.

The Driver Diagnostics section of the Diagnostics panel shows the following information about the status of these drivers:

Field	Description
Timestamp	The date and time that the driver was loaded.
Name	The name of the driver.
Version	The version of the driver.
Is Loaded	Shows whether the driver is loaded (true or false).
Load Status	The load status of the driver.

Reducing the Impact of Netconn Data Collection (Windows)

On systems that have a large number of network connections (for example, DHCP/DNS servers, domain controllers, build servers, etc.), netconn data collection by the sensor can cause significant CPU utilization by the Carbon Black service. If this is an issue but you want to continue collecting netconn data, Windows sensors beginning with v6.1.4 let you disable the DNS name resolution in data collection for network connections, thereby reducing the amount of netconn traffic on these systems. This is done by configuring the following Windows registry key:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\CarbonBlack\config]
"DisableNetConnNameResolution"=dword:00000001
```

Event Diagnostics

The **Event Diagnostics** panel shows the date and timestamp (in GMT) of sensor events, together with the number of each of the following event elements:

- Messages Generated
- Messages Logged
- Raw Events Observed
- Raw Events Throttled
- Raw Events in Process
- Raw Events Filtered Out
- Raw Events Discarded

Sensor Status History

This topic describes the **Status History** panel on the Sensor Details page.

- The **Component Status** section shows information about threads and the sensor component.
- The **Isolation Audit** section shows details about the sensor's isolation/remove isolation activity. The data presented is sorted by timestamp in descending order. Isolation Audit history data retention time is 12 weeks (~3 months).
- The **Upgrade Status** section shows details about the sensor upgrade process.
- The **Uninstall Status** section shows data if the sensor has been recently uninstalled.

Sensor Groups

6

This section describes how to create, move, edit, and delete sensor groups.

Carbon Black EDR sensors are lightweight data gatherers installed on network endpoints (such as laptops, desktops, and servers). They gather event data on the endpoints and securely deliver it to the Carbon Black EDR server for storage and indexing. Each sensor is associated with a sensor group that defines its configuration and security characteristics. One sensor group can contain many sensors, but a single sensor can only belong to one sensor group.

Sensor groups can be based on your security and organizational requirements. For example, you might base sensor groups on functional groups (such as marketing, customer service, or IT) or by location.

If you move sensors from one sensor group to another, the sensors will receive security settings from the new group the next time they check back into the server. In most cases, you do not have to re-install the sensors when you move them.

See also:

- The *Carbon Black EDR Sensor Installation Guide* for information on installing, uninstalling, and troubleshooting sensors.
- Chapter 5 Managing Sensors for information on managing sensors.
- Chapter 7 Managing Certificates for information about certificate options.
- Chapter 23 Sensor Parity for information on supported operating systems. This topic indicates whether a supported configuration is available on a sensor and configurable on a sensor group.

Read the following topics next:

- Create or Edit a Sensor Group (macOS or Windows Sensors)
- Move Sensors to Another Group
- Delete Sensor Groups

Create or Edit a Sensor Group (macOS or Windows Sensors)

Use the following procedure to create or edit a sensor group for macOS or Windows sensors. You can create sensor groups before or after you install sensors.

Procedure

1. On the navigation bar, click **Sensors**.
2. In the **Groups** panel of the Sensors page, do one of the following:
   - To create a new group, click **NEW** at the top of the **Groups** panel.

     The **Create Group** panel appears.
   - To edit an existing group, do one of the following:
     - Select a group and at the top of the Sensors page, click **Edit**.
     - Next to a group name, click the gear icon(⚙).

       The **Edit Group** panel appears.

   In the **Create Group** or **Edit Group** panel, sensor group settings are organized in sections that you can expand or collapse.

   Note To quickly open or close all sections of sensor group settings at one time, click **Display All Sections** or **Collapse All Sections** at the top right of the **Create Group** or **Edit Group** panel.

General Settings

The **General** section of the **Create Group** or **Edit Group** panel includes the following settings.

Setting	Description
Name	The name of the sensor group; alphanumeric characters only.
Sensor Process Name	(Optional, Windows-only) An alternate name for the sensor group process. The default name of the process is `cb.exe` . For example, you can change the default name if Operations Security (OPSEC) policies require sensors to run with a non-standard or obfuscated executable name. If you change the name of the sensor process, the process will run with this name instead of the default `cb.exe` . This will not change the Windows service display name, but it will change the name of the actual executable that is run.
Server URL	The URL that the sensor group uses to communicate with the Carbon Black EDR server. This URL is the same one that is used to log into the Carbon Black EDR server, prefixed with sensors. Use HTTPS and specify the secure port in the URL.

Setting	Description
Site Assignment	Select a site to assign to this sensor group. You can use site definitions to define throttle settings to manage bandwidth for groups of computers. These settings are applied per site, not per sensor group. If bandwidth is an issue for this group of sensors, create or configure a site with the appropriate bandwidth settings in ***Username*> Settings > Sites**, and then assign the site to this sensor group by selecting the site in this field. **Note:** Modifying bandwidth settings for sites requires Carbon Black EDR Global Administrator status or Carbon Black Hosted EDR Administrator status. Additional information about site throttling is available in the *Carbon Black EDR Server Operating Environment Requirements Guide*.
Assign Server Certificate	Assign a server certificate to all sensors in the group. Only sensors that check in receive this update. This field includes a **Manage certificates** link that goes to the **Server Certificates** tab of the Settings page. See Chapter 7 Managing Certificates.

Sharing Settings

The **Sharing** section of the **Create Group** or **Edit Group** panel includes the following settings.

Setting	Description
Share Binary hashes with Carbon Black	Select this option to be notified of any binary that is flagged by Carbon Black Collective Defense Cloud. For more information about this choice, do the following from the **Sharing** section of the **Create Group** or **Edit Group** page: 1 Click **Share Settings** to open the Sharing page for the Carbon Black EDR server. 2 On the Sharing page, scroll to **Endpoint Activity Sharing**. 3 In the **Carbon Black** column next to **Binary Hashes & Metadata**, click the current setting (**Enabled**, **Disabled**, or **Partial**) for a description.
Send events to Carbon Black	Select this option to: ■ Allow advanced analysis of your aggregated process execution events by the Carbon Black Threat Research Team. ■ Give your enterprise access to enhanced Carbon Black Threat Intel information that is only available to those who participate in the community program. The Carbon Black Threat Research Team receives process events (as shown on the Process Analysis page) for more detailed analysis of the behavior of a process as it executes at the customer site. For more information, see Chapter 10 Process Search and Analysis.
Allow Carbon Black to analyze unknown binaries	Select this option t o get advanced analysis of binary content from the Carbon Black Threat Research Team. By sharing binaries with Carbon Black, our researchers will perform advanced static analysis of your binaries to alert you to suspicious activity. Select this option to detect new variants of known malware. For more information about this choice, do the following: 1 Click **Share Settings**. 2 Scroll to **Endpoint Activity Sharing**. 3 Click the current setting (**Enabled**, **Disabled**, or **Partial**) for a full description.

Default settings for the sensor group sharing settings are defined on the global Sharing page, which you can access in the following ways:

- Click the **Share Settings** link in the sensor group **Sharing** section.
- Click ***Username*> Sharing Settings** .

For more information, see Threat Intelligence Data Sharing Settings.

Advanced Settings

The **Advanced** section of the **Create Group** or **Edit Group** panel includes the following settings.

Setting	Description
Sensor-side Max Disk Usage	This setting contains two options to limit sensor disk consumption on clients either by raw available space (in megabytes) or percentage of the total space available. The sensors will limit the amount of space they use on clients based on the smaller of these two values: ■ In the **MB** field, enter the maximum available space on the client (between 2 and 10240 megabytes). ■ In the **%** field, enter the maximum percentage (between 2% and 25%) of total disk space on the client.
Filter known modloads (Windows and macOS only)	When selected, Carbon Black EDR will not report the module load events of known good Windows and macOS modules that reside on the operating system. This helps reduce the number of known good events that are reported to the server.
Process Banning	When selected, this setting enables process hash bans in this group. By default, this setting is disabled and process hash bans prevent banned processes from running. For more information, see Banning Process Hashes.
Tamper Protection Level (Windows only)	This setting determines the tamper detection or protection level for the sensor on the endpoint. When set to **None**, no tamper detection or protection exists. This is the default setting. When set to **Detection**, the sensor identifies attempts to modify the sensor configuration or memory and alerts on these attempts. This setting is only applicable for Windows sensors version 5.0.0 and higher. With the 7.2.0 Windows sensor release, Tamper Detection protects against process injection attempts. When set to **Protection**, the sensor blocks local admin attempts to inject, remove, modify, or delete the sensor by protecting the sensor service, drivers, files, folders, registry settings and other sensor components. This setting is only applicable for Windows sensors version 7.2.0 and higher. To change this setting you must be one of the following: a Global Administrator (Carbon Black EDR), an Administrator (Carbon Black Hosted EDR), or a user who is an Analyst for this sensor group and who also has permission for Tamper Level. For more information about Tamper Protection, see Tamper Protection of Windows Sensors.
Tamper Override Password	This field contains a password to temporarily disable tamper protection on the sensor from the CLI, in case the sensor cannot reach the server.
VDI Behavior Enabled	When selected, this setting enables Virtual Desktop Infrastructure (VDI) for sensors on virtual machines. Use VDI when endpoints that are virtual machines are re-imaged. Sensor IDs are maintained across re-imaging by hostname, MAC, or other determining characteristics. **Note** VDI support must be globally enabled to use this feature. See the *Carbon Black EDR Integration Guide*.

Setting	Description
Retention Maximization	These settings change how sensor process data that contains only modload processes or only modload and cross processes is recorded on the server. Minimum Retention makes this data more easily searchable but leaves a bigger footprint and can lead to a reduction in data retention time. Recommended and Maximum Retention consolidate data under parent processes, reducing the data footprint and helping increase the retention time. Data consolidated in this way is still searchable, as child processes. ■ **Minimum Retention** – All process activity is recorded and available for search. ■ **Recommended Retention** – The processes that contain only modload events are available under the parent processes and are searchable as child processes. You can search metadata, such as command line and user context, under the parent process. ■ **Maximum Retention** – The processes that contain only modload and cross processes are available under the parent processes and are searchable as child processes. You can search metadata, such as command line and user context, under the parent process. **Note** Recommended and Maximum Retention can result in false positives in the results of cmdline searches. See Retention Maximization and cmdline Searches. **Note** This setting was called **Data Suppression Level** in pre-6.5 versions of Carbon Black EDR.
Alerts Critical Severity Level	Select a value from the menu to alter the critical level for alerts on a per-sensor-group basis. This directly effects the severity rating for alerts generated by this sensor group. On the Triage Alerts page, the severity score of an alert (located in the **Severity** column of the Results table) is determined by three components: ■ Feed rating ■ Threat intelligence report score ■ Sensor criticality. For example, server sensors can have a higher criticality than engineering workstations. If two sensor groups have different alert criticalities, and they receive alerts from the same feed and for the same report, the sensor group that has the higher alert criticality will have a higher severity score on the Triage Alerts page, and servers in that group will appear at the top of the queue. For more information about alerts, see Chapter 20 Console and Email Alerts. For more information about threat intelligence feed scores, see Chapter 14 Threat Intelligence Feeds.

Permissions Settings

In the **Permissions** section of the **Create Group** or **Edit Group** panel, you can define user team permissions for sensors groups.

Available permission levels are as follows:

- **Analyst** – Users can configure the sensor host and group details.
- **Viewer** – Users can view the data collected from hosts in this sensor group. Users cannot make any configuration changes to this group or hosts that belong to it.
- **No Access** – When users in a team try to access or view details on a host in this sensor group, the system generates the following HTTP 405 response: “The method you are using to access the file is not allowed.”

For information about user teams and access levels, see Managing User Access with Teams.

Event Collection Settings

In the **Event Collection** section of the **Create Group** or **Edit Group** panel, you can define which types of events to record for the sensors in this group by selecting/deselecting event types.

Disabling event collection impacts visibility, but can improve sensor and server performance. Disabling Process Events or Windows Events can cause an "Event Loss" message in the console.

The Event Collection options are explained here:

- **Process information** – Collects process metadata such as starts, stops, and process id (PID).
- **Process user context** – Enables the sensor to record the user name that is associated with each running process. This associates endpoint activity with the operating system user account.
- **File modifications (Filemods)** – Carbon Black EDR captures four types of file system activity:
  - File creation – the creation of a new file.
  - File Write – the first time a file is written to after being opened or created.
  - File Write Complete – the closing of a file that was written to. This event includes both the file path and also the MD5/SHA256 of the written file. The event is only captured for binaries (Windows PE such as EXE, DLL, and drivers), Adobe Docs (PDF), OfficeXML docs (docx, doc, xlsx, xls, pptx, ppt) and zip archives (zip) that are smaller than 10MB in size. This option can be enabled or disabled independently of filemod collection by deselecting **Non-binary file writes**. This option is not available with macOS or Linux sensors.
  - File deletion – the deletion of an existing file.
- **Binary module loads (Modloads)**
  - Reported as a result of `LoadImageNotify` kernel callback and triggered when a binary is mapped into memory.
  - This is the only place in the binary load/execution chain where Windows provides a supported interface to register for notifications.
  - There are conditions where a binary is mapped, but is not subsequently executed. For example: `LoadLibaryEx()` where the `dwFlags` parameter includes LOAD_LIBRARY_AS_IMAGE_RESOURCE or DONT_RESOLVE_DLL_REFERENCES. Windows does not distinguish between binaries that are loaded for execution and binaries that are loaded as a resource.
- **Network connections (Netconns)** – Carbon Black EDR captures network connections that have the following characteristics:
  - TCP over IPv4 or UDP over IPv4 connections.
  - Inbound and outbound connections:

Network connections record TCP or UDP protocol, the remote IPv4 address, port and the domain name associated with the remote IPvAddress.

Inbound connections capture the local port. If the sensor is installed on a typically configured web server, the reported port is 80.

Outbound connections capture the remote port.

For outbound connections that are made after DNS resolution, the name that resolves to the captured IPV4 address is also reported.

The sensor utilizes a passive sensing approach to capturing the domain name, so no additional network traffic is generated.

For DNS/DHCP servers, high CPU and/or memory can be seen due to the high number of netconn events. Instead of disabling all netconn events, disable DNS capture on that machine.

- **Fileless script loads** – Collection of `fileless_scriptload` events through an integration with Microsoft Antimalware Scan Interface (AMSI). Forwarding of these events through the Event Forwarder was introduced in Carbon Black EDR Server 7.2.0 and Carbon Black EDR Windows Sensor 7.1. However, full support of this event type, including storage, console display, and API support, is available in Carbon Black EDR Server 7.6.0. The `fileless_scriptload` event represents each occasion when the sensor detects PowerShell script content that was executed by any process on a supported endpoint. For more information about AMSI, see the *Carbon Black EDR Integration Guide*.
- **Cross process events** – Enables the sensor to record instances when a process crosses the security boundary of another process. Although some of these events are benign, others might indicate an attempt to change the behavior of the target process by a malicious process.

  Certain limitations exist on the cross process events that are reported by the sensor:

  - Parent processes that create cross process events to their children are not reported.
  - Cross process events that are part of the normal OS behaviors are ignored. For example, no cross process events are recorded for the Windows process `csrss.exe` .
  - Cross process events are not reported for macOS or Linux sensors.
  - Cross process, open process, and open thread events are not supported on Windows XP and Windows 2003.
- **Registry modifications** – Carbon Black EDR captures four types of registry activity from both the machine (HKEY_LOCAL_MACHINE or HKLM) and user (HKEY_USERS or HKU) registry hives:
  - Registry key creation
  - Registry key deletion
  - Registry value modification – the creation or modification of a registry value of any type

  - Registry value deletion
- **EMET events**
  - Used in conjunction with the EMET Protection feed to report EMET events on the endpoint.
  - EMET must be installed on the Windows endpoint. Recent operating systems replace EMET with Defender Exploit Protection; this is not currently supported.
- **Binaries (physical storefiles)** – Carbon Black EDR collects binary files, such as Windows PE files (EXEs, DLLs, SYS), OSX binaries, and Linux ELF binaries. For binaries that are larger than 25MB, the first 25MB of the binary is captured. Binaries are compressed before they are transmitted to the Carbon Black EDR Server. The server stores one copy of each unique binary.
- **Binary info (binary metadata)** – Carbon Black EDR captures metadata about binaries that are executed on the endpoint. This metadata includes:
  - Size in bytes
  - Internal version information (file version, product version, etc.)
  - Digital signature information (signature status, digital signer, revocation status, etc.)
  - Icon

Turn off Event Collection of Non-binary File Writes

Some endpoints produce large amounts of non-binary files types, and can therefore produce a massive inbound queue of mostly uninteresting files. This can lead to decreased data retention and system resource usage to ingest this data on the server. If the large amount of non-binary file writes is determined to be an issue, perform the following procedure to turn off this type of event collection.

For the most part, Carbon Black EDR does not record information regarding non-binary files types. However, Carbon Black EDR does record file writes of certain non-binary file types. The following is a list of non-binary files types that the Carbon Black EDR sensor records when they are written to disk:

PE	Elf	UniversalBin
EICAR	OfficeLegacy	OfficeOpenXml
Pdf	ArchivePkzip	ArchiveLzh
ArchiveLzw	ArchiveRar	ArchiveTar
Archive7zip		

Procedure

1 You can create a new sensor group to contain the sensors that are generating the non-binary file write events, or you can edit an existing sensor group. See Create or Edit a Sensor Group (macOS or Windows Sensors) for step-by-step instructions.

2 Click the **Event Collection** tab.

3 Deselect the **Non-Binary File Writes** check box.

4 Save the sensor group.

5 Add sensors to the sensor group as needed. See Move Sensors to Another Group.

Exclusion Settings

Through an addition to the `cb.conf` file, an **Exclusions** section can be added to the **Create Group** or **Edit Group** panel on the Sensors page. This **Exclusions** section lets you define paths to executables to customize event collection from those executables to improve performance or eliminate unnecessary data.

For example, you can specify that execution of one set of applications do not collect network connections or non-binary file writes. You can create another exclusion for a different set of applications that collects everything except cross-process events.

For Windows, be careful when adding multiple paths per exclusion. Syntax errors in one path can cause others that follow that path to not be recognized.

macOS Example:

The Xcode application (which is known to generate a lot of events) can be excluded by adding the path `/Applications/Xcode.app/Contents/MacOS/Xcode`.

Note For more information about `cb.conf`, see the *Carbon Black EDR Server Configuration Guide*.

Add Exclusion Settings to the Sensor Group Panel

Follow this procedure to add **Exclusion** settings to the **Sensor Group** panel.

Procedure

1 On the Carbon Black EDR server, open `/etc/cb/cb.conf` for editing.

2 Add the following setting and value to the `cb.conf` file; consider including a comment to remind you of the purpose of the setting:

```
EventExclusionsEnabled=True
```

3 Save the `cb.conf` file.

4. You must stop and restart the server or cluster to make the new setting effective:

   - For a standalone server:

     ```
     sudo service cb-enterprise restart
     ```

   - For clusters:

     ```
     sudo cbcluster stop
     ```

     (...wait for all the nodes to shut down, and then...)

     ```
     sudo cbcluster start
     ```

Create Exclusions

You can specify exclusions when you create a sensor group, or add them to an existing sensor group. The following procedure assumes that the sensor group already exists.

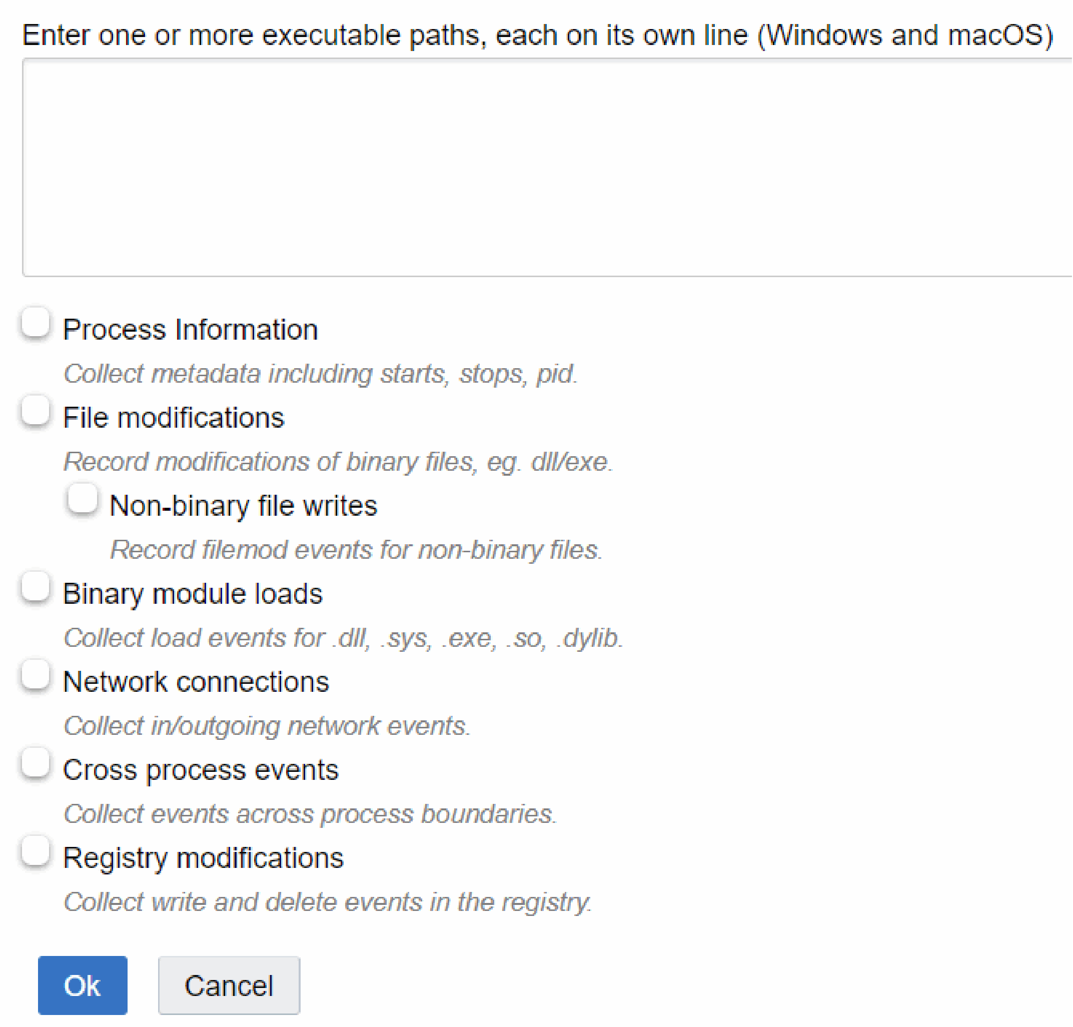
Prerequisites

Before you can perform this procedure, you must add **Exclusion Settings** to the **Sensor Group** panel. See Add Exclusion Settings to the Sensor Group Panel.

Procedure

1. On the left navigation bar, click **Sensors**.
2. In the **Groups** panel of the Sensors page, click the gear icon (⚙) next to the sensor group for which to create exclusions.
3. Click the **Exclusions** bar and click the **Add Exclusion** button.

   The **Exclusion** configuration fields are exposed.



Enter one or more executable paths, each on its own line (Windows and macOS)

□ Process Information

Collect metadata including starts, stops, pid.

□ File modifications

Record modifications of binary files, eg. dll/exe.

□ Non-binary file writes

Record filemod events for non-binary files.

□ Binary module loads

Collect load events for .dll, .sys, .exe, .so, .dylib.

□ Network connections

Collect in/outgoing network events.

□ Cross process events

Collect events across process boundaries.

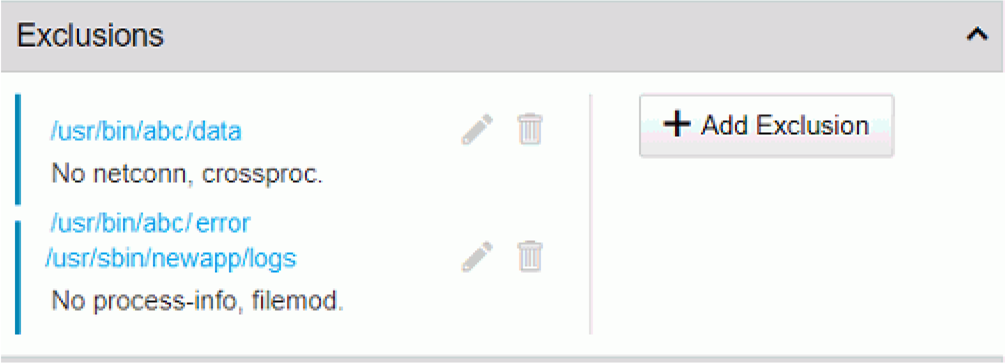
□ Registry modifications

Collect write and delete events in the registry.

4 Enter the path(s) to affect with this exclusion in the textbox in the upper right corner of the panel. Put each path on a new line.

5 Check the box next to each type of information to not collect for the specified paths. Click **Ok** .

The exclusions are saved and displayed in the panel. You can edit or delete any exclusion.



Exclusions

/usr/bin/abc/data

No netconn, crossproc.

/usr/bin/abc/ error
/usr/sbin/newapp/logs

No process-info, filemod.

6 When you have finished creating exclusions, click the **Save Group** button.

Note

- You can use the wildcard * in exclusion paths. The exclusions will apply to all executables under the wildcard path. Be careful when using wildcards: having too many wildcards can affect performance.
- Child processes inherit the event exclusion settings of the parent process.

Upgrade Policy Settings

The **Upgrade Policy** section of the **Create Group** or **Edit Group** panel lets you set a policy to upgrade installed sensors in the sensor group.

Upgrade policy options are as follows:

- **No automatic updates** – Manually decide when to upgrade sensors.
- **Automatically install the latest version** – Automatically upgrades the sensors to the latest version.
- **Automatically install a specific version** – Installs a specific version for all sensors in a group. This maintains all sensors at the selected version. Select a version number using the drop-down list. Selecting the upgrade policy of a specific version is useful when sensor versions must be tested.

Move Sensors to Another Group

After you create sensor groups, you can add sensors to them. By default, sensors are installed into the **Default Group**

On the Sensors page, you can select the group that contains the sensors to add, and then move those sensors from their original group to the new group.

Procedure

1 On the navigation bar, click **Sensors**.

2 In the **Groups** panel, click the sensor group that contains the sensors to move.

3 In the **Sensors** panel, select the sensors to move.

4 Click **Actions > Move to group**.

5 From the drop-down list, select the sensor group to which to move the selected sensors and click **Okay**.

The selected sensors are removed from the former sensor group list and appear in the new sensor group list.

Note If you have set up custom server certificates and strict certificate validation, and you have assigned different certificates to different sensor groups, moving a sensor to another group could affect connectivity. See Chapter 7 Managing Certificates.

Delete Sensor Groups

You can delete sensor groups on the Sensors page. When you delete a sensor group, the teams for which you defined permissions no longer have access to sensors that belong to the deleted sensor group.

Procedure

1 On the navigation bar, click **Sensors**.

2 In the **Groups** panel, click the sensor group to delete.

3 At the top of the Sensors page, click **Delete Group**.

A confirmation message appears indicating that any sensors remaining in this sensor group will be moved to the **Default Group**.

4 Click **OK**.

Managing Certificates

7

This section describes how Carbon Black EDR uses HTTPS and TLS to secure communication and two-way authorization between endpoints and the server.

It also details certificate management features, including the ability to add your own server certificates, assign different certificates to different sensor groups, and opt for stricter certificate validation.

Read the following topics next:

TLS Server Certificate Management Overview

Carbon Black EDR uses the HTTPS and TLS (formerly SSL) protocols to secure communication and two-way authorization between endpoints and the server so that the endpoint communicates only with the Carbon Black EDR server that it trusts, and the server only communicates with trusted endpoints.

Prior to server version 6.4.0, Carbon Black EDR established the trust between endpoints and the server by using “certificate pinning,” which is an out-of-band, reliable and secure trust mechanism. The server built the endpoint installer packages, and those came pre-initialized with the server identity (the public portion of server’s TLS certificate). The Carbon Black EDR server acted as its own root certificate authority (CA), which allowed it to issue client-side certificates that the endpoints could use. This feature is still available and is the default option for securing server to sensor communications.

If you are satisfied with the security that is provided by the certificate generated by your Carbon Black EDR server and do not have any special compliance requirements, you can continue to use the standard certificate and validation method, which relies on certificate pinning only. Past and current sensors continue to support this method.

Beginning with Carbon Black EDR Server 6.4.0, you can choose to provide certificates signed by your organization. In addition, you can use different server certificates to authenticate the connections between the Carbon Black EDR server and different sensor groups, thereby reducing the exposure to a compromised server certificate. You can also add stricter validation methods to certificate pinning so that if a server certificate used by a sensor has expired or fails to meet other operating-system-specific criteria, server-sensor communication is disabled.

See Sensor Support for Certificate Management for information about the sensor versions that support certificate management on each operating system.

In a cluster environment, primary and minion servers use the same certificates. If you add your own certificates to the primary, they are automatically propagated to the minions within a few seconds (unless there are connection issues). No server restart is required. The required format for user-provided certificates allows them to be seamlessly used in a clustered environment.

In addition, Carbon Black EDR provides new certificate visibility features that can be useful for user-provided and Carbon Black EDR “legacy” certificates.

Note

- Currently, you can use certificates signed by your own certificate authority, but use of a certificate that requires validation by a third-party CA is not supported.
- The certificate management features described here apply only to server-sensor communications. They are not used for managing other Carbon Black EDR interactions, such as the connection between the console user interface and the server.

Certificate Management Feature Summary

This topic summarizes Carbon Black EDRcertificate management.

- **Add and delete certificates** – You can add new certificates and delete certificates from your server.
- **View certificate inventory** – A table lists all server certificates that are available on the current server, how many sensors are using each one, and additional certificate information.
- **Choose validation method** – You can use standard certificate “pinning” validation, which only requires that sensors have a certificate matching the server, or you can add stricter validation methods. A certificate that uses standard validation continues to allow sensor and server to communicate even after it expires. but strict validation disables communication after expiration.
- **Be notified of expiring certificates** – When a certificate is close to its expiration date, an alert banner can be displayed at the top of each console page. You can set the number of days in advance you want to be warned, or turn off warnings. Deleting the expired certificate eliminates the notification.

- **Assign and change certificates by sensor group or apply one to all sensor groups** – If you have more than one certificate available, you can choose the certificate that is assigned to secure server communications for each sensor group. You can also apply one certificate to all sensor groups. This can be done for both the initial certificate assignment and to assign a new certificates — for example if a certificate is ready to expire.
- **View the certificate for a sensor** – The Sensors page shows the server certificate that was used for the last successful check-in for each sensor.
- **Control access to certificate features** – Because of their security implications, certificate management features require Global Administrator privileges on the server.

Server-Sensor Certificate Requirements

Whether added during server installation or later through the console, server certificates that are used for sensor communications must meet the following requirements.

- The files you provide must be valid certificate and key files (that is, they must be recognized as a certificate/key pair by the OpenSSL library).
- Certificate files must be in unencrypted ASCII PEM format – this includes both the certificate file and the key file.
- The certificate must have valid dates when uploaded – that is, its "not valid before" date should be in the past and its "not valid after" date should be in the future.
- Certificates must have two distinct SAN DNS entries to address the Carbon Black EDR cluster scenario where sensors must resolve primary and minion virtual addresses to different IP addresses or FQDNs. This is required for every server certificate, even in standalone configurations, so that certificates remain valid if a standalone instance is upgraded to a cluster. The second SAN field is a single virtual address used for all minions, but it is mapped to a different IP address or FQDN hostname as needed by the sensor itself.
- SAN DNS entries must meet the standards for hostname formatting, but should not match any of the existing accessible DNS addresses. It should contain a unique element (for example, virtual_a prefix) that allows the server to support multiple different certificates behind the same DNS hostname. Allowed characters include the hyphen and alphanumeric characters (a to z and 0 to 9). Invalid SAN DNS entries can silently fail and might cause connectivity loss to the server.
- The CN field is not used for validation of new certificates because it has been deprecated. Sensors perform their own local resolution of virtual names to real server addresses, so no additional DNS entries are required.
- When generating replacement custom certificates, the SANs must once again be unique and non-routable. They must be unique not only amongst themselves, but also unique to the certificates they are replacing. As the Carbon Black EDR server compares old and new certificates for veracity, it ensures old SANs are not honored when the new certificate is deployed. To do this, the old SANs cannot be reused.

- Applicable for EL 8 environments only: FIPS-compliant OpenSSL on EL 8 enforces stricter certificate validations.
  - Purpose of the Certificates: Certificates should state usage via Key Usage x509 extensions.
  - Certificate Signing Algorithms: Must be one of the following: SHA-256 (Secure Hash Algorithm 256-bit), SHA-384 (Secure Hash Algorithm 384-bit), or SHA-512 (Secure Hash Algorithm 512-bit).
  - Certificate Asymmetric Encryption Algorithm: RSA with a minimum key size of 2048 bits.

The following example shows how to set up the SAN portion of the certificate if you want to upload two certificates. The first SAN.DNS entry is used for the primary cluster node and the second is used for the minions.

Certificate A

```
CN=<something>
SAN.DNS.1=virtual-a.primary
SAN.DNS.2=virtual-a.minion
```

Certificate B

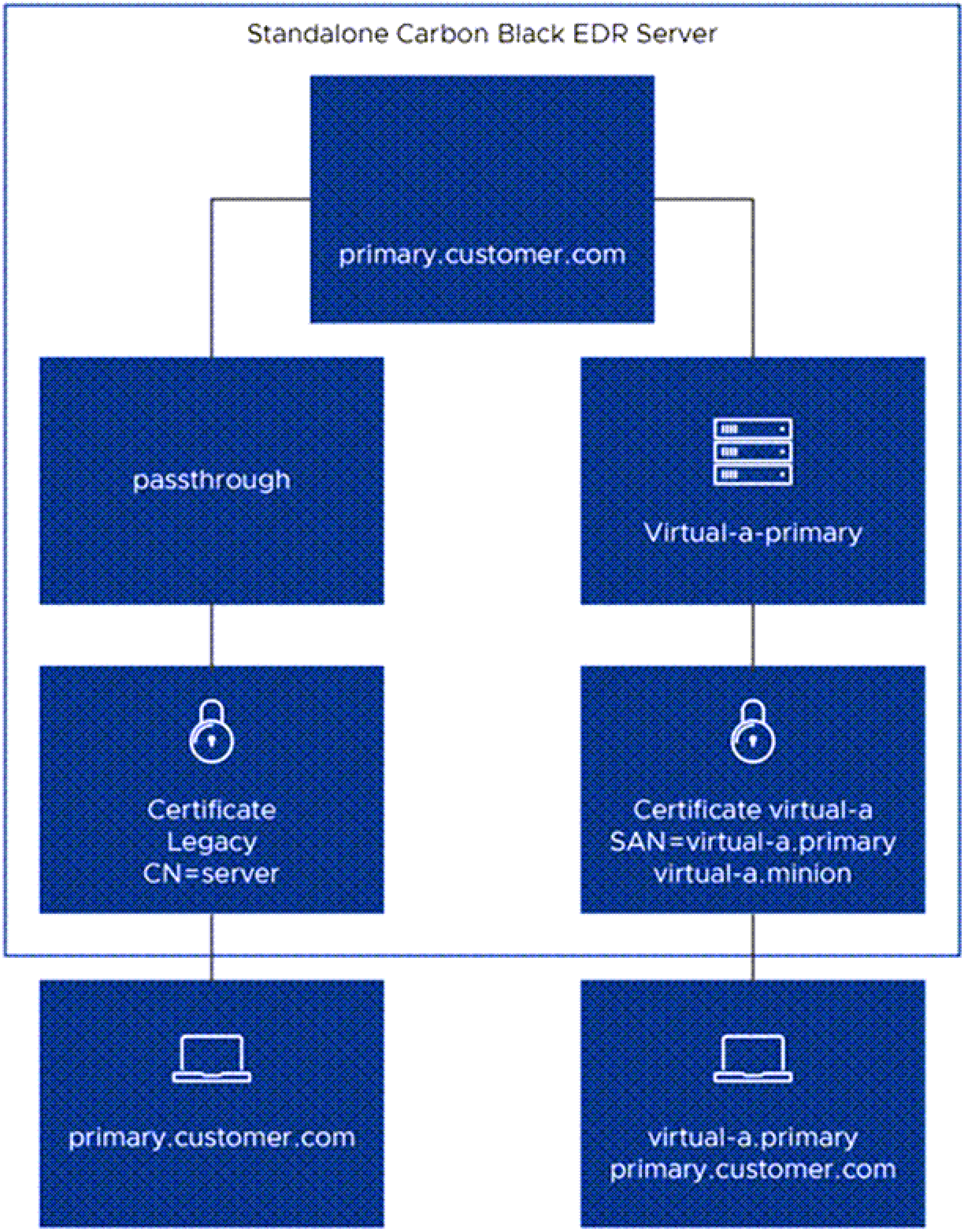
```
CN=<something>
SAN.DNS.1=virtual-b.primary
SAN.DNS.2=virtual-b.minion
```

Multiple Certificate Support

The Carbon Black EDR server uses virtual server names to allow multiple active routes to the server using the same real address and port.

Each virtual name and route uses a different certificate, as depicted in the following schematic. This implementation is done via runtime server blocks in NGINX configuration files.

Virtual server names are parsed from a SAN.DNS entry in the certificate so that each certificate can be validated from the sensor’s perspective.


Standalone Carbon Black EDR Server
primary.customer.com
passthrough
Virtual-a-primary
Certificate
Legacy
CN=server
Certificate virtual-a
SAN=virtual-a.primary
virtual-a.minion
primary.customer.com
virtual-a.primary
primary.customer.com

Sensors use the Server Name Indication (SNI) extension to the TLS protocol handshake to access a specific route and certificate. The Legacy certificate remains available without any SNI indications so that older sensor versions are able to use it to access the server.

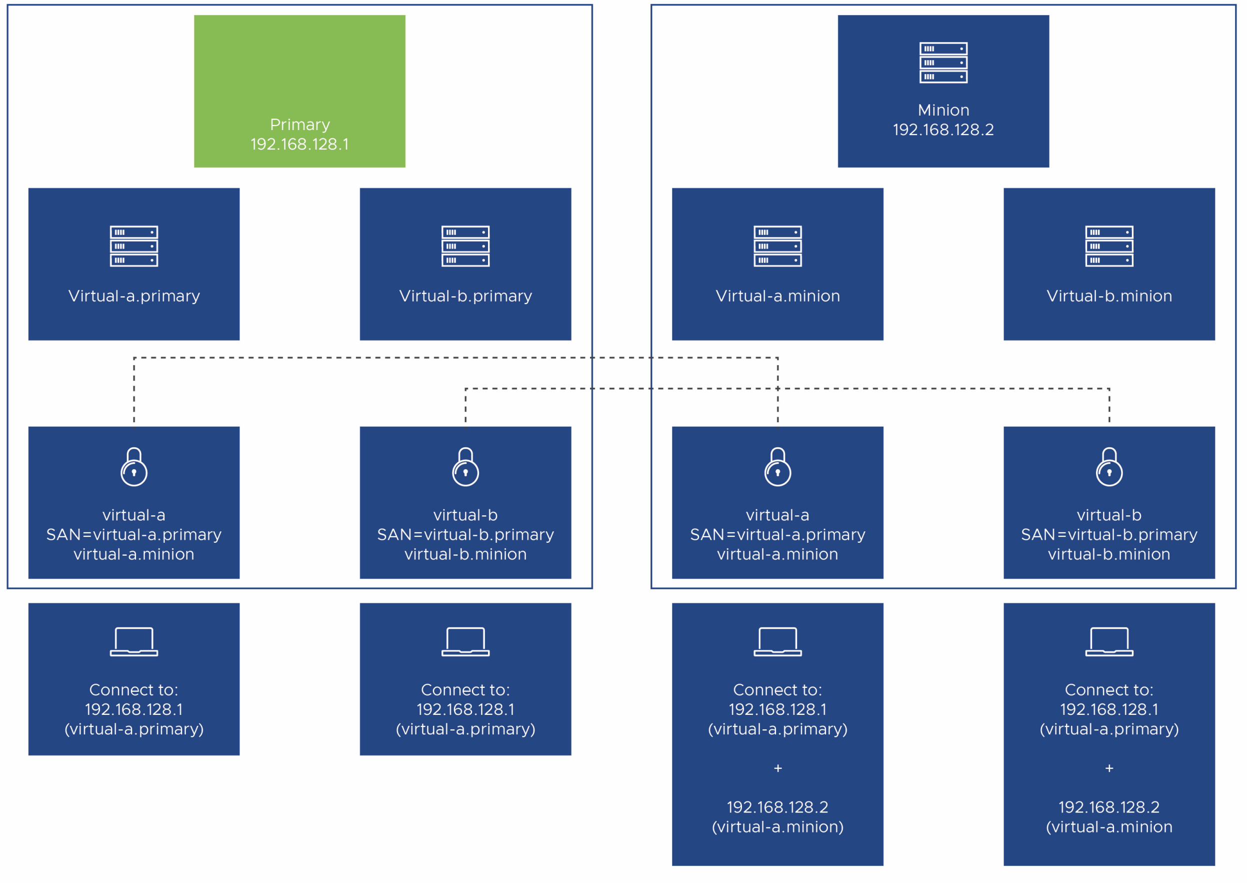
Sensors confirm the resolution of virtual names to addresses internally; for example, resolution from "virtual-a.primary" in the preceding certificate example to the actual "primary.customer.com" server address. That means that virtual server addresses do not need to be added to external DNS servers.

On an upgrade to Carbon Black EDR 6.4.0, the previously used self-signed certificate is named "Legacy" and would be served via the default server access route (using no virtual servers), which is supported for all sensor versions and configurations. New server installations also include a "Legacy" certificate.

Note If you are using a Reverse Proxy, you must manually configure your Reverse Proxy to match the SNI configuration in your server environment. Contact Carbon Black Support for additional details.

Using Multiple Active Certificates in a Cluster

The following schematic describes some details of the clustered server set up with TLS certificate management. Although the legacy route is not depicted in this schematic for readability, it does exist and works in a cluster just as it does in a standalone scenario.



When TLS certificate management is used in a cluster environment:

- Added TLS certificates are automatically copied to minions.
- Virtual server names are replicated on minions.
- Sensors expect the same server certificate to be present on primary and the minion.
- Server certificates support the switch from standalone to cluster or adding new nodes without having to re-issue certificates. This is the reason why the certificates must have two distinct SAN DNS entries.
- Sensors need to distinguish only two different virtual addresses to communicate to the servers (primary and minion). Because resolution of those virtual names happens internally, and virtual minion addresses can get mapped to different real addresses, there is no need for SAN DNS entries for each individual minion in a cluster.

Managing Certificates on the Server

This section describes the tasks you perform to use Carbon Black EDR certificate management features.

You have two opportunities to use server certificates other than the default legacy certificate:

- During server initial installation and configuration, you can substitute your own certificate for the one that would be created by default. See Substituting a Legacy Certificate during Server Installation.
- After the server is installed and configured, you can add certificates through the console. You can do this whether or not you supplied a new legacy certificate during installation. See Add Certificates through the Console.

When you have the certificates you intend to use in place, you can:

- Choose the validation methods that sensors use for certificates. See Choosing a Validation Option.
- Specify the certificate to use for each sensor group or specify the certificate to use for all sensor groups. See Assigning Certificates to Sensor Groups.

You can add certificates, change validation method, and change certificates assigned to sensor groups later, but implementing an initial certificate configuration as soon as possible may be more efficient and prevent disruptions in server-sensor communication.

Viewing Certificate Information in the Console

Certificate information appears in several places in the Carbon Black EDR console.

- The Sensors page includes a column showing the certificate for each sensor.
- The Edit Group page for a sensor group shows the certificate assigned to that group.
- The Server Certificates page shows all of the sensor-server certificates that are available on the current server. It also shows the validation method that is being used for these certificates. See Choosing a Validation Option.

View Certificate Information in the Console

Perform the following procedure to view the available certificates on a server.

Procedure

1 Click ***Username*** > **Settings**.

2 Click **Server Certificates**.

Substituting a Legacy Certificate during Server Installation

When you install a new Carbon Black EDR server, the `cbinit` configuration program you run after installation installs a legacy certificate for use with the standard pinning validation method. By default, this is a certificate that the server produces.

As an alternative to the default legacy certificate, you can substitute your own certificate during the server installation process. In either case, the certificate will be named “Legacy” where certificates appear in the console, and it will be protected from deletion.

Important Certificates and key files added in this way must meet the requirements described in Server-Sensor Certificate Requirements.

When you substitute your own certificate using `cbinit` , Carbon Black EDR runs tests to confirm that the certificate is valid for this use. If the certificate passes the test, it is used for this server. If it is not valid, the default legacy certificate is used, an error message will appear, and the certificate import failure will be logged to `/var/log/cb/cli` . The `cbinit` process still continues if the substitution fails by using the default certificate instead of the one you tried to substitute.

Substitute a Legacy Certificate during Server Installation

Perform the following procedure to upload a custom “legacy” certificate during server installation.

This procedure is for substituting your certificate for the single, legacy certificate only. If you intend to use more than just the legacy certificate, use the console for any additional certificates you need. See Add Certificates through the Console.

Procedure

1. Prepare the certificate you want to use and place it and its key file in an accessible location on the system hosting the Carbon Black EDR server (the primary in a clustered environment).
2. Enter the yum install command for installing the correct server version and wait for that process to complete. See the *Carbon Black EDR Server Cluster Management Guide* for additional installation instructions.
3. When the installation completes, run the following command, providing the arguments and file paths to the certificate file and the key file as shown here:

   ```
   cd /usr/share/cb
   sudo cbinit --server-cert-file=<certpath> --server-cert-key=<keypath>
   ```

4. If the certificate and key files pass all tests, they become the default server certificate and key, and are copied into the server as `/etc/cb/certs/cb-server.crt` and `/etc/cb/certs/cb-server.key`.

Add Certificates through the Console

You can add certificates to the Carbon Black EDR server through the console to secure server-sensor communications.

Prerequisites

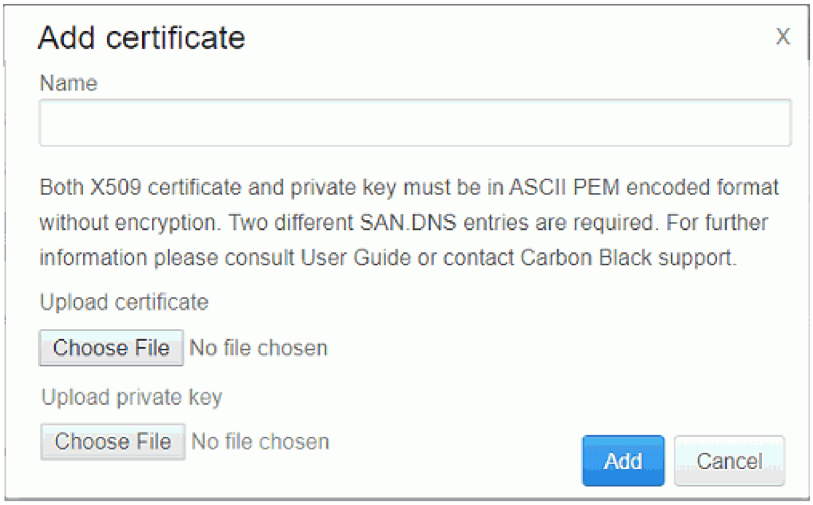
Important Certificates and key files added in this way must meet the requirements described in Server-Sensor Certificate Requirements.

Procedure

1 Click *Username*> **Settings** .

2 Click **Server Certificates** and click the **Add certificate** button.

3 In the **Add certificate** dialog, provide a unique name for the certificate to identify its purpose (use 50 or fewer alphanumeric characters without spaces).



Add certificate

Name

Both X509 certificate and private key must be in ASCII PEM encoded format without encryption. Two different SAN.DNS entries are required. For further information please consult User Guide or contact Carbon Black support.

Upload certificate

Choose File No file chosen

Upload private key

Choose File No file chosen

4 Under **Upload certificate**, click **Choose File** and provide the path to a certificate file that meets the requirements described in Server-Sensor Certificate Requirements.

5 Under **Upload private key**, click **Choose File** and provide the path to the ASCII PEM-encoded, unencrypted key file for this certificate.

6 When you have entered all required information, click the **Add** button in the dialog.

If it passes all tests, the new certificate is listed in the table on the Server Certificates page and is available for use by sensors.

Choosing a Validation Option

You can choose one of two validation methods that sensors use for the server certificates that are used to secure server-sensor communication.

The validation method can be set through the following console method or by providing a value in the `cb.conf` file for `CbServerSSLCertStrictCheck`, in which case it cannot be changed in the console. For more information about `cb.conf`, see the *Carbon Black EDR Server Configuration Guide*.

If the standard validation method (certificate pinning only) is used, certificate expiration does not interrupt server-sensor communication, although an expiration warning will appear if this is configured. The only requirement is that the server and sensor certificates match.

If strict certificate validation is used, the requirements of standard validation must still be met, but additional checks are done on the sensor side. A certificate that has expired or fails any other validation requirements causes server-sensor communication to be disabled. See Sensor Support for Certificate Management for the validation requirements on different sensor platforms.

Caution Do not enable strict validation if you are using the legacy certificate created during Carbon Black EDR server installation. Using strict validation for this or any other certificate that cannot pass validation will disable communication between the sensor and server on some sensors that support the certificate management features, and can require uninstalling and reinstalling sensors.

Change the Validation Method for Server Certificates

Perform this procedure to change the validation method for server certificates.

Procedure

1 Click ***Username*** > **Settings**.

2 Click **Server Certificates**.

3 Two radio buttons/options appear under **Server certificate validation mode**:

   - **Standard validation** – Sensors will only require that their certificate matches the server certificate when connecting.
   - **Strict certificate validation** – Sensors will require that a matching certificate is valid on the host machine when connecting. This includes checking whether the certificate has expired.

   If the button for the method you want to use is not selected, click it.

4 Click the **Save changes** button and click **Confirm**.

   The change will be propagated to all sensors that support TLS server certificate management during their next checkin.

   Caution As the confirmation dialog states, changing validation method can disable communication between sensor and server. Make sure that you have configured certificates properly before changing this setting, especially if you are changing to strict validation.

Change the Expiration Notification Period

You can configure the Carbon Black EDR server to display a warning banner when any of its server certificates is ready to expire. The values available for this are: 0, 15, 30, 60 or 90 days. If the value provided is 0 (zero), no warning appears.

This value can also be set through the console as described here, or by providing a value in `CbServerCertWarnBeforeExpirationDays` in the `cb.conf` file, in which case it cannot be changed in the console. For more information about `cb.conf`, see the *Carbon Black EDR Server Configuration Guide*

If enabled, the warning displays for any expiring certificate listed on the Server Certificates page, even one not used by any sensor groups. If you see warnings for a certificate you are not using and will not use later, delete the certificate to prevent unneeded warnings. See Delete a Certificate from a Server.

Procedure

1 Click ***Username*** > **Settings**.

2 Click **Server Certificates**.

3 In the **Notify me** menu above the table, select the number of days in advance to be warned about expiration of any certificates (whether or not they are being used by any sensors).

Delete a Certificate from a Server

You might want to remove a certificate so that it cannot be used (for example, if it has expired or has been compromised).

You can remove any certificate except the following:

- You cannot delete a certificate that is currently in use by a sensor group.
- You cannot delete the legacy certificate that is created during server installation.

Procedure

1 Click ***Username*** > **Settings**.

2 Click **Server Certificates**.

3 Check that the certificate to delete does not have any sensors using it. If the certificate is not in use, click **Actions > Delete** for that certificate.

4 In the confirmation dialog, click **Delete**.

Caution After you confirm the deletion of a server certificate, any sensors that were using the certificate can no longer communicate with the server. There is no Undo for this action. Although you cannot delete a certificate that is being used by a Sensor Group, it is possible that an offline sensor could miss a change of certificate for its group, and come back online configured to use a certificate that has been deleted.

Upgrades from Previous Server Releases

When you upgrade to Carbon Black EDR Server version 6.4.0, the previously used certificate appears in the server certificates table – that is, the certificate called “Legacy”.

Unless you change it, standard validation (certificate pinning) remains in effect. This allows the server and sensors to communicate as before the upgrade.

After the upgrade is complete, you can implement a different certificate management strategy if you choose. Subsequent server upgrades maintain whatever certificates you have in place at the time of the upgrade.

Assigning Certificates to Sensor Groups

If new or different certificates are assigned to any sensor group, the change of certificates is made for each sensor the next time it checks in with the server.

In addition to using the newly assigned certificate on all subsequent communications with the server, the sensor also stores certificate details locally for use on sensor restarts.

During a change of certificates, the server accepts connections from the sensors utilizing either of two server certificates: the certificate being replaced or the new certificate. Sensor-server communication is not interrupted by certificate replacement. After the connection is successfully established using the new certificate, the old certificate is overwritten and is no longer available for use by the sensor.

If the sensor cannot connect with the new certificate, it reverts to the previous sensor certificate.

For older sensor versions that do not support certificate swaps, the legacy certificate remains in place, regardless of a global or per-sensor-group certificate change. Consider reviewing which sensors support certificate management features before assigning certificates to a group. See Sensor Support for Certificate Management.

In clustered environments, certificate changes are automatically propagated to all servers within a matter of seconds, without requiring a restart.

Assign Different Certificates to Different Sensor Groups

Perform the following procedure to assign a certificate to one sensor group.

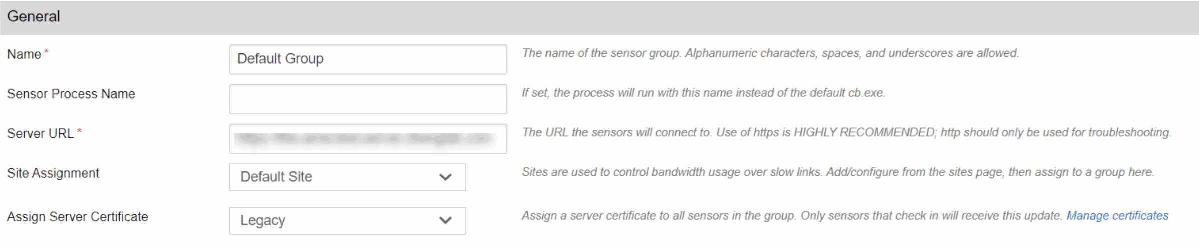
Your organization might consist of multiple sites or groups whose endpoints are mapped to different sensor groups. You can use different server certificates to authenticate the connections between the Carbon Black EDR server and different sensor groups, thereby reducing the exposure to a compromised server certificate. This lets you manage certificate expiration on a per sensor group basis.

You can also use the per-sensor-group assignment of certificates to gradually change a certificate, even if you want to use the same certificate for all sensors. After you see successful server-sensor communications for one group, you can assign the certificate to all sensors or continue assigning the new certificate on a per sensor group basis.

Procedure

1. In the navigation bar, click **Sensors**.
2. In the left panel, click the name of the sensor group whose certificate you want to change. Click the **Edit** button.

3 In the **General** panel of the Edit Group page, click the **Assign Server Certificate** drop-down menu to choose the certificate to use for this group.



General

Name *	Default Group	The name of the sensor group. Alphanumeric characters, spaces, and underscores are allowed.
Sensor Process Name		If set, the process will run with this name instead of the default cb.exe.
Server URL *	[redacted]	The URL the sensors will connect to. Use of https is HIGHLY RECOMMENDED; http should only be used for troubleshooting.
Site Assignment	Default Site	Sites are used to control bandwidth usage over slow links. Add/configure from the sites page, then assign to a group here.
Assign Server Certificate	Legacy	Assign a server certificate to all sensors in the group. Only sensors that check in will receive this update. Manage certificates

4 Click the **Save Group** button at the bottom of the page.

Assign a New Certificate to all Sensor Groups

You might need to use your own custom certificate, capable of strict validation, for communication between the Carbon Black EDR server and all sensors. If you do not need different certificates for different sensor groups, Carbon Black EDR provides a single-click method to assign one new certificate to all sensor groups.

Note Before assigning to all sensor groups, the recommended best practice is to validate certificate connectivity on at least one active sensor group first.

Procedure

1 Click ***Username*** > **Settings**.

2 Click **Server Certificates**.

3 Click the **Actions** drop-down menu and click **Assign to all sensor groups**.

4 Click **Confirm**.

Sensor Support for Certificate Management

Certificate management features are available on Carbon Black EDR Server versions 6.4.0 and later. How those features affect sensors depends on the sensor version and the OS platform of the sensor.

Other than expiration warnings, sensors that do not support TLS certificate management are unaffected the new certificate management settings.

Sensors that do not support certificate swaps continue using the legacy certificate provided by the server, regardless of the certificate assigned to their sensor group.

If you select **Standard validation**, the only requirement for a valid connection is that there is an exact hash match between the certificate on the sensor and the certificate on the server. If you select **Strict validation**, the exact hash match is still required, plus additional validation criteria that varies by platform.

The following list shows the sensors that are included with Carbon Black EDR Server 6.4.0 and their support for certificate management:

- **Windows sensor 6.2.3** – This and later sensors support certificate management and handles strict validation. See Special Requirement for Windows Sensors.

  Windows XP and Windows Server 2003 do not support TLS certificate swap, regardless of the Carbon Black EDR sensor version.
- **macOS sensor 6.2.5** – This and later sensors support the new certificate management features and handles strict validation as shown in the following table.
- **Linux sensor** – As of the version 7.0.0 server release, Linux sensors do not support certificate management but continue to use the default “Legacy” certificate.

The following table shows the different validation criteria that are available for the sensor versions on each platform.

Strict validation mode requirements by sensor platform

Requirement	macOS Sensor 6.2.5+	Windows Sensor 6.2.3+
Exact certificate match (certificate pinning)	Yes	Yes
Expiration date	Yes	Yes
Certificate validation chain	-	Yes
Hostname matches (SAN=)	-	Yes
Writable host file	-	Yes
Revocation check	-	-
Key Usage is Server Auth (1.3.6.1.5.5.7.3.1)	-	Yes

Special Requirement for Windows Sensors

Certificate swapping on an endpoint running the Windows sensor requires that the sensor is able to update the system `hosts` file.

The `hosts` file is a text file that maps IP addresses to hostnames. The file is located at: `C:\Windows\System32\drivers\etc\hosts.`

To confirm that the `hosts` file can be updated successfully:

- **Check AV Exceptions** -- The Carbon Black EDR sensor service must be allowed to open and edit the `hosts` file. By default, it has that permission since it is running as administrator. However, other security products (typically anti-virus products or other monitoring tools) must not block the Carbon Black EDR sensor from accessing the file. If necessary, add exclusions to other security products to allow the Carbon Black EDR Windows sensor to access the `hosts` file. Failure to do so can result in loss of communications between sensors and server.

- **Save the File in ASCII (Windows Sensor 6.2.3 and 6.2.4)** -- For Windows sensor releases through version 6.2.4, the `hosts` file is assumed to be in ASCII encoding. If the sensor modifies an instance of the file that was saved with non-standard encoding, the file can become unreadable.

  If it has been saved in a different format, resave the file in ASCII. For example, in the Windows Notepad application, click **Save As…** and then select **ANSI** as the encoding.

  See the post-6.2.4 sensor release notes to determine whether this requirement still applies.

Upgrading to Sensors that Allow Certificate Management

To use the certificate management features of Carbon Black EDR and upgrade your sensors to a version that is compatible with certificate management, the best practice is to upgrade the sensors first and let the upgrades complete before applying a custom certificate to them.

This best practice reduces the possibility of communication issues due to a mismatch between the server certificate and the sensor during the upgrade. After the sensors are updated, you can apply the custom certificate.

Important If a sensor group is assigned a custom certificate, sensors in that group that support custom certificates cannot be downgraded to sensor versions that do not support custom certificates. Attempts at such a downgrade fail and log an error in the sensorservices debug log.

Responding to Endpoint Incidents

8

This section describes how to respond to endpoint incidents by isolating endpoints by using Live Response, and by banning process hashes.

When you discover a malicious file or process on your endpoint(s) using Carbon Black EDR, you can address the issue in a variety of ways. Carbon Black EDR provides the following methods for responding to threats directly from the console:

- **Endpoint Isolation** – You can isolate an endpoint from the rest of the network, leaving only the connections that are needed for access to its sensor by the Carbon Black EDR server.
- **Live Response** – Live Response opens a command interface for direct access to any connected host running the Carbon Black EDR sensor. Responders can perform remote live investigations, intervene in ongoing attacks, and instantly remediate endpoint threats.
- **Process Hash Banning** – You can ban a process hash so that the process cannot be run again on hosts reporting to this Carbon Black EDR server, and any running version of it is terminated.

These features can be used together or separately. For example, you can isolate an endpoint immediately to prevent the spread of the problem and then use Live Response to end the process and perform any other file removal or needed repairs.

On the other hand, if the incident is not ongoing, isolation might not be necessary. In that case, you can use Live Response to remediate or further investigate the issue on affected endpoints, or simply ban the hash for the malicious process.

Carbon Black EDR does not present a message on the affected endpoint when any of these features is used on an affected sensor. With endpoint isolation, a user would likely become aware quickly that they had lost network access, but would not know why. With Live Response, actions you take on a computer might affect a user's access to files or programs, but there would be no indication that Carbon Black EDR tools are responsible, unless you have chosen to make the user aware of that. Also, when there is an attempt to run a process that is banned by hash, the operating system might display a dialog indicating a lack of access, or the process might silently fail to run.

If you also have the Carbon Black App Control agent on your endpoints, you can use Carbon Black App Control control features to investigate incidents and modify rules to prevent future occurrences. See the *Carbon Black EDR Integration Guide* for details.

Note To use the features described in this chapter, a user must be one of the following:

- A user that has the enhanced Analyst permission for the feature and is a member of a team that has the Analyst role for the sensor group for the endpoint being acted upon (or for any sensor group to ban hashes).
- For Carbon Black EDR installations, a Global Administrator.
- For Carbon Black Hosted EDR installations, an Administrator.

See Chapter 3 Managing User Accounts (Carbon Black EDR) or Managing User Accounts (Carbon Black Hosted EDR) for more information about user roles and privileges.

Read the following topics next:

- Isolating an Endpoint
- Using Live Response
- Banning Process Hashes

Isolating an Endpoint

You can isolate one or more endpoints from the rest of your network and the Internet through the Carbon Black EDR console.

When an endpoint is isolated, its connectivity is limited to the following (unless you have created network isolation exclusions as described in Create an Isolation Exclusion):

- The Carbon Black EDR server can communicate with an isolated computer.
- To allow the sensor to communicate with the Carbon Black EDR server, ARP, DNS, and DHCP services remain operational on the sensor’s host. (For Windows operating systems prior to Vista, ICMP (for example, ping) will remain operational.)
- DNS and DHCP are allowed through on all platforms. This is required for proper communications to the Carbon Black EDR server. Protocols are allowed by UDP/53, UDP/67, and UDP/68.
- ICMP is allowed on the following operating systems:
  - Windows (operating systems prior to Vista)
  - macOS
  - Linux
- UDP is blocked on all platforms.

Isolate an Endpoint

Perform the following procedure to isolate an endpoint.

Prerequisites

To isolate an endpoint, you must be a Carbon Black EDR Global Administrator, a Carbon Black Hosted EDR Administrator, or a user on a team that has Analyst privileges for the endpoint to isolate.

Procedure

1. On the navigation bar, click **Sensors**.
2. Check the box next to each endpoint to isolate.
3. From the **Actions** drop-down menu, click **Isolate**.
4. Optional - Note the reason for this action in the **Description** text box.
5. Click **OK** to confirm that you want to isolate these endpoints.

   The endpoint is isolated from all but the Carbon Black EDR server and the network services that are required to connect the two, in addition to any addresses that are allowed due to network isolation exclusions.

   When you designate an endpoint for isolation, its status on the server first moves into an “isolation configured” state waiting for its next check-in. Because of this, several minutes can pass before the endpoint is actually isolated. When it checks in, the server tells the sensor to isolate the endpoint, and when the sensor responds, its state changes to “isolated”.

   After it is isolated, endpoints normally remain isolated until the isolation is ended through the Carbon Black EDR console. However, if an isolated system is rebooted, it is not isolated again until it checks in with the Carbon Black EDR server, which could take several minutes.

   Having isolated endpoints, you can proceed with remediation steps. For example, you might use Live Response to investigate or modify an endpoint. When you are finished, restore connectivity to the endpoints that you isolated. See Restore Connectivity to an Isolated Endpoint.

Restore Connectivity to an Isolated Endpoint

Perform the following procedure to end isolation on an endpoint and restore its connectivity to your network and the Internet.

Procedure

1. On the navigation bar, click **Sensors**.
2. Check the box next the endpoints for which to restore network connectivity.
3. From the **Actions** drop-down menu, click **Remove isolation**.
4. Optional - Note the reason for this action in the **Description** text box.

5 Click **OK** to confirm the restoration.

The endpoints return to the network with the same access they had before they were isolated (unless you made access changes through Live Response).

Network Isolation Audit

Starting with Carbon Black EDR version 7.7.0, you can monitor all network isolations and remove isolation activities.

The audit support feature lets you track isolation and remove isolation activities so you can coordinate with others in your organization for any additional or critical work needed on the endpoint.

Audit information contains the following details:

- Timestamp: Date and time of the activity
- Action: Isolate or remove isolation
- User Details: Username, first and last name of the user who performed the activity
- User IP address: IP address of the user's client machine
- Notes: An optional note describing the reason of the activity

The isolation audit table can be viewed on the Sensor Details page under the **Status History** panel. See Sensor Status History.

Note To isolate an endpoint, you must be a Carbon Black EDR Global Administrator, a Carbon Black Hosted EDR Administrator, or a user on a team that has Analyst privileges for the endpoint to isolate.

Create an Isolation Exclusion

Starting with Carbon Black EDR version 6.5.0, Windows sensors version 6.2.4 and higher, and macOS sensor versions 6.2.7 and higher, support isolation exclusions. You can add one or more IPv4 addresses or domain URLs that isolated sensors can access in isolation mode in addition to the Carbon Black EDR server.

Prerequisites

This setting is applied on a per-sensor-group basis. It is disabled by default; to enable it, you must edit the `cb.conf` file. See the *Carbon Black EDR Server Configuration Guide* for instructions.

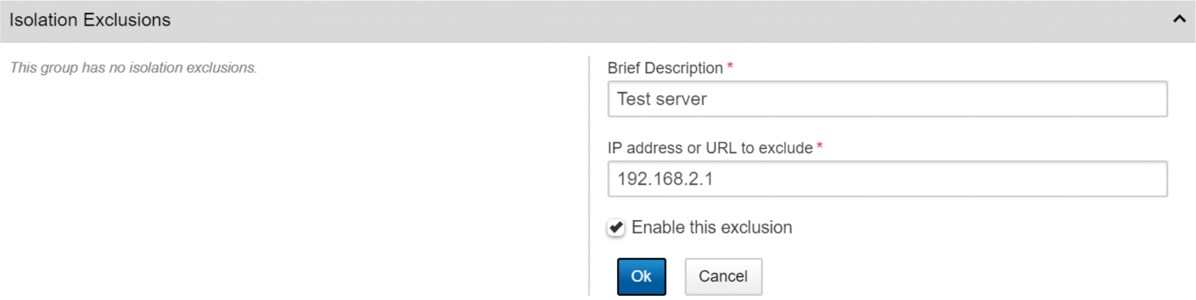
Procedure

1 On the navigation bar, click **Sensors**.

2 Click the gear icon next to the sensor group for which you want to add isolation exclusions.

3 Click **Isolation Exclusions** and then click **Add Exclusion**.

4 Enter a description that identifies the exclusion (50 character maximum), and the IPv4 address or domain URL that specifies the exclusion (253 character maximum).



Isolation Exclusions

This group has no isolation exclusions.

Brief Description *

Test server

IP address or URL to exclude *

192.168.2.1

☑ Enable this exclusion

5 Select **Enable this exclusion** and click **OK**.

6 Click **Save Group**.

After you have created an exclusion, you can edit it by clicking the pencil icon, or you can remove the exclusion by clicking the trash can icon.

Note Duplicate exclusions are not allowed. If you enter the same IP address or URL for more than one exclusion, the last entry that was submitted is retained, but the duplicated entry is removed.

Using Live Response

Live Response opens a command line interface for direct access to any connected host that is running the Carbon Black EDR sensor.

Responders can perform remote live investigations, intervene in ongoing attacks, and instantly remediate endpoint threats. For example, Live Response allows a responder to view directory contents, kill processes, modify the registry, and get files from sensor-managed computers.

Live Response is disabled by default on newly installed Carbon Black EDR systems. Enable or Disable Live Response through the Console and Tune Live Response Network Usage describe ways to enable the feature and adjust data settings.

Important Live Response feature should be used in full compliance with your organization's policy on accessing other user's computers and files. Consider the capabilities described here before giving users access to the feature and choosing the sensor group in which you will place endpoints.

If you do not want console administrators for Carbon Black EDR installations to activate Live Response, make sure `CbLREnabled=False` is set in your `cb.conf` file and is not commented out. For more information about `cb.conf`, see the *Carbon Black EDR Server Configuration Guide*.

There are two Live Response modes:

- **Attached Mode** – When you activate Live Response for a specific endpoint, you create and attach to a *session*. The interface for a session includes information about the endpoint and a command window for interacting with the endpoint. See Live Response Endpoint Sessions.
- **Detached Mode** – You can enter Live Response without being attached to a particular session through the **Go Live** command on the console menu. This interface includes commands to manage and access existing sessions as well as commands that are useful outside of a session. See Live Response Detached Session Management Mode.

Enable or Disable Live Response through cb.conf

For Carbon Black EDR servers, you can edit the `cb.conf` file to fix the state of Live Response so that it cannot be enabled or disabled through the console. This option is not available in Carbon Black Hosted EDR.

Procedure

1. On the Carbon Black EDR server, open `/etc/cb/cb.conf` for editing.
2. Add or uncomment the following setting in the `cb.conf` file and set its value:

   ```
   CbLREnabled=True
   -or-
   CbLREnabled=False
   ```

3. Save the `cb.conf` file.
4. You must stop and restart the standalone server or cluster to make the new setting effective:

   - For a standalone server:

     ```
     sudo service cb-enterprise restart
     ```

   - For clusters:

     ```
     sudo cbcluster stop
     ```

     (...wait for all the nodes to shut down, and then...)

     ```
     sudo cbcluster start
     ```

   For more information about `cb.conf`, see the *Carbon Black EDR Server Configuration Guide*.

Enable or Disable Live Response through the Console

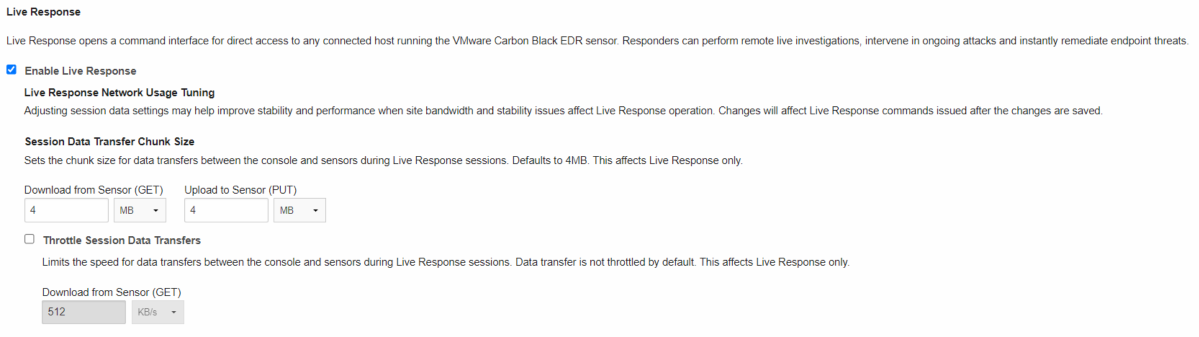
There are two ways to enable and disable Live Response: through the `cb.conf` file or (if not set in `cb.conf`), through the Carbon Black EDR console. Perform the following procedure to enable or disable Live Response through the console.

Prerequisites

If `CbLREnabled` has no value (or is commented out) in the `cb.conf` file, an administrator can enable or disable Live Response in the console using a switch on the Advanced Settings page. This is the default setting in Carbon Black EDR version 6.3.0 and later.

Procedure

1. Log in as a Global Administrator (Carbon Black EDR) or Administrator (Carbon Black Hosted EDR).
2. Click ***Username*** > **Settings**.
3. Click **Advanced Settings** and scroll to the **Live Response** section.
4. Check or uncheck **Enable Live Response** and click the **Save changes** button.



Live Response

Live Response opens a command interface for direct access to any connected host running the VMware Carbon Black EDR sensor. Responders can perform remote live investigations, intervene in ongoing attacks and instantly remediate endpoint threats.

☑ Enable Live Response

Live Response Network Usage Tuning

Adjusting session data settings may help improve stability and performance when site bandwidth and stability issues affect Live Response operation. Changes will affect Live Response commands issued after the changes are saved.

Session Data Transfer Chunk Size

Sets the chunk size for data transfers between the console and sensors during Live Response sessions. Defaults to 4MB. This affects Live Response only.

Download from Sensor (GET): 4 MB

Upload to Sensor (PUT): 4 MB

☐ Throttle Session Data Transfers

Limits the speed for data transfers between the console and sensors during Live Response sessions. Data transfer is not throttled by default. This affects Live Response only.

Download from Sensor (GET): 512 KB/s

Note If the **Enable Live Response** box is grayed out and unresponsive, the value is set in `cb.conf` and cannot be changed through the console. See Enable or Disable Live Response through cb.conf.

Tune Live Response Network Usage

Sites that have bandwidth or stability issues might experience performance problems or failures with Live Response. To help mitigate these issues, you can adjust the data transfer chunk size and also enable throttling of data transfers between the Carbon Black EDR console and sensors during a Live Response session.

Changes to these settings affect only Live Response, and are effective only on commands that you issue after the changes are saved. The settings apply to all users.

Prerequisites

Live Response must be enabled before you can modify its settings. See Enable or Disable Live Response through cb.conf or Enable or Disable Live Response through the Console.

Procedure

1. Log in as a Global Administrator (Carbon Black EDR) or Administrator (Carbon Black Hosted EDR).

2 Click ***Username*** > **Settings**.

3 Click **Advanced Settings** and scroll to the **Live Response** section.

4 In the **Live Response Network Usage Tuning** section, under **Data Transfer Chunk Size**, set new **Download from Sensor (GET)** and/or **Upload to Sensor (PUT)** values. The default value for both settings is 4MB.

5 In the same section, check the box for **Throttle Session Data Transfers** if you want to activate throttling. Throttling is turned off by default, and it must be turned on before you can edit its value.

6 Set the throttling speed in **Download from Sensor (GET)** to a new value. The default setting when throttling is first activated is 512KB.

7 Click the **Save changes** button.

Live Response Endpoint Sessions

To access an endpoint using Live Response, a user must either have Carbon Black EDR Global Administrator or Carbon Black Hosted EDR Administrator privileges, or be on a team with the Analyst role for that endpoint. A session must first be created with the sensor. A session indicates that the sensor is connected to the Carbon Black EDR server to receive real-time commands.

Sessions are created and attached automatically when you click the **Go Live** button on the Sensor Details or Process Analysis pages. If you enter the Live Response console using the **Go Live** command from the console menu, access to an endpoint requires that you first create and attach a session:

```
session new [sensor_id]
```

```
attach [provided_session_id]
```

You can have sessions that have multiple sensors active at the same time. Use the `detach` command to detach from a session but leave it active.

Use the `session close` command to end a session with the sensor. Sessions will timeout when they are not attached and active for five minutes.

Each session has a unique numeric ID. Up to 10 sessions can be running at one time, and multiple users can be attached to the same session.

Note More than one Carbon Black EDR console user can attach to the same session with an endpoint at the same time. If more than one user submits a command through the session at approximately the same time, each command must finish executing before the next one can begin. Also, one user can undo or otherwise modify what another user is doing. Consider this if more than one user has Live Response access to an endpoint.

The following table shows the complete set of Live Response commands. In the descriptions, **remote host** refers to the host that is being accessed through Live Response, and **local host** refers to the host on which the user is running the Carbon Black EDR console. These commands are all run in the SYSTEM context.

Command	Description
archive	Obtain an archive (gzip tarball) of all the session data for this session, including commands run and files downloaded. The archive is downloaded to the computer on which you are running the Carbon Black EDR console by using the browser's download method.
argparse	Test how Live Response parses CLI arguments. This command helps determine if there are any interpretation issues. For example, it can reveal whether spaces or other special characters are properly escaped.
cd [dir]	Change the current working directory. Options include absolute, relative, drive-specific, and network share paths.
clear	Clear the console screen; the `cls` command can also be used for this purpose.
delete [path]	Delete the file specified in the path argument. The file is permanently deleted, not sent to the Recycle Bin.
detach	Detach from the current Live Response session. If a session has no attachments, it remains live until it times out (five minutes by default).
dir	Return a list of files in the current directory or the specified directory if it is added to the command, (for example, `dir c:\temp` or `dir /tmp`)
drives	List the drives on the remote host. This is for Windows only.
exec[processpath]	Execute a background process specified in the processpath argument on the current remote host. By default, process execution returns immediately and output is to stdout and stderr. Options may be combined: ■ **exec**`-o outputfile processpath` – Redirect the process output to the specified remote file, which you can download. ■ **exec**`-w processpath` – Wait for the process to exit before returning. You could combine the options as shown in the example below to execute and capture the output from a script: **exec**`-o c:\output.txt -w c:\scripts\some_script.cmd` You must provide the full path to the process for the processpath argument. For example: `c:\windows\system32\notepad.exe`
execfg [processpath]	Execute a process on the remote host and return stdout/stderr. For example, this command prints the output of ipconfig to the screen: `execfg c:\windows\system32\ipconfig /all`

Command	Description
files [-s session] [action] [option]	Perform actions over cache-stored session files. All files transferred to/from an endpoint with every Live Response session are cached on the server for a period of time after a session is closed. If there is an interruption in the connection between a user's browser and the Carbon Black EDR server, files can be retrieved directly from the cache instead of connecting to the sensor again. This command is valid in both the global and session scopes when attached to a sensor. In the global scope, the session ID must be defined with `-s` . A list of sessions is available through the `sessions` command. If attached to a sensor, the current session is assumed unless otherwise specified. There are three available actions: ■ `list` – List all the cached files that are available in the specified session by file ID. ■ `get [id]` – Get the file [id] from the cache. ■ `delete [id]` – Remove the file [id] from the cache.
get [path]	Obtain the file specified in the path argument from remote host and download it to the host running the Carbon Black EDR console for this session. Progress of the download is indicated in the Live Response window as described in Live Response Status, Error, and Progress Messages.
help	Show the Live Response session commands with a brief description of each. If a command name is added, show the description of the specified command, with additional details (such as options) if available. For example: `help dir`
hexdump	Output the first 50 bytes of the file in a hexdump format.
kill	Terminate the specified process.
memdump [filepath]	Take a full system memory dump and store it to the given file path, which must include a file name. When the memory dump is completed, it is automatically compressed and uploaded to the server. If you name the file with a .zip extension, it will be uploaded using the file name you provided. Otherwise, Live Response will append .zip to the name you provide. Once uploaded, the .zip file can be downloaded through the Carbon Black EDR console. Memory dumps can take several minutes. Progress is indicated in the Live Response window as described in Live Response Status, Error, and Progress Messages. The memdump command is for Windows hosts only.
mkdir	Make a directory on the remote host.
ps	Obtain a list of processes from the remote host. In the output from this command, the listing for each process includes an **Analyze** link. Clicking the link opens the Process Analysis page for the process. Note that analysis information for a newly discovered process might not yet be fully committed to the Carbon Black EDR database and therefore not viewable. Clicking the link navigates away from the Live Response console and loses whatever context you had there.
put[remotepath]	Put a file from the host on which the console is being run onto the remote host at the specified path. You specify the file in the **Open** dialog of the browser, after the command is entered in Live Response. Progress of the upload is indicated in the CBLR console as described in Live Response Status, Error, and Progress Messages.

Command	Description
pwd	Print the current working directory.
reg	View or modify Windows registry settings. The syntax of this command is: **`reg`***[action] [key] [options]* See Registry Access in Live Response or use `help reg` in the Live Response command window for details. This command is for Windows only.

As shown in the preceding table, some commands provide information and others allow you to modify an endpoint.

Note Be sure to use the commands and options as documented here. Although some of the Live Response commands are the same as commands in the DOS command interface, the available options are specific to Live Response.

Create and Attach to a Live Response Sensor Session

Perform the following procedure to create and attach to a Live Response sensor session.

Procedure

1. On the navigation bar, click **Sensors** and then click the name of the endpoint.
2. Click **Go Live**.

   The Live Response page appears with a command window on the left and an information panel on the right. The command window prompt shows the name of the host and the current directory in which Live Response is active. The information panel includes the following:

   - **Host Details**
   - **Alerts** related to the host
   - **Running Processes** on the host

   A status indicator (dot) and a message appear immediately above the command window. The dot has the following color code:

   - **Green** – The sensor is connected and a session has been established. The host name is shown.
   - **Orange** – The Carbon Black EDR server is waiting for the sensor to check in, or no host is connected because no session is attached.
   - **Gray** – A session cannot be established with the sensor because the host is offline, the sensor is disabled, or the sensor is not a version that supports Live Response.

3. To view a list of the available commands, click in the command window area and enter the `help` command. You can get information about a specific command by entering:

   ```
   help commandname
   ```

For a complete list of Live Response commands, see Live Response Endpoint Sessions.

Live Response Status, Error, and Progress Messages

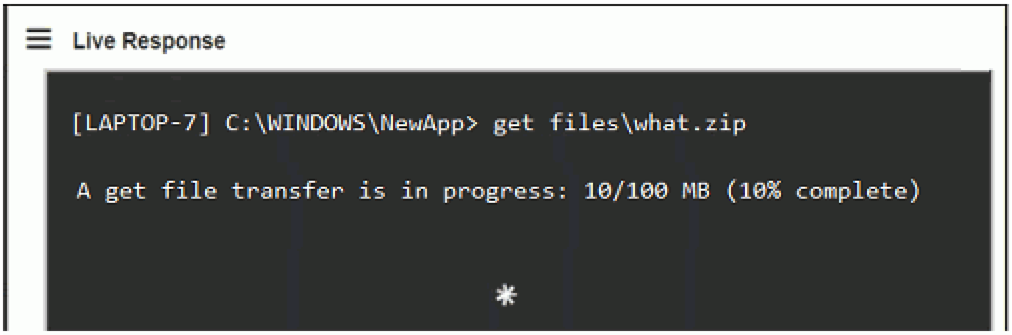
Live Response status and error messages should inform you of any connection or command error issues, but you can also use the `dir` or `pwd` commands to confirm your connection.

For commands that involve file transfers (`get`, `put`, and `memdump`), Live Response reports on the progress of the transfer. As soon as you begin a session, Live Response monitors for file transfer activity.

When one of the file transfer commands is executed, a rotating asterisk character appears in the Live Response window indicating that the transfer is happening. A series of progress messages appear.

The messages depend on the size of the transfer and whether you initiated the transfer in the current session or have attached to an existing session:

- For file transfers that take more than a few seconds, the Live Response window shows the number of bytes and the percentage of the total already transferred, updated every five seconds. This information is displayed to both the user who initiated the command and to any user who is attached to the same session.

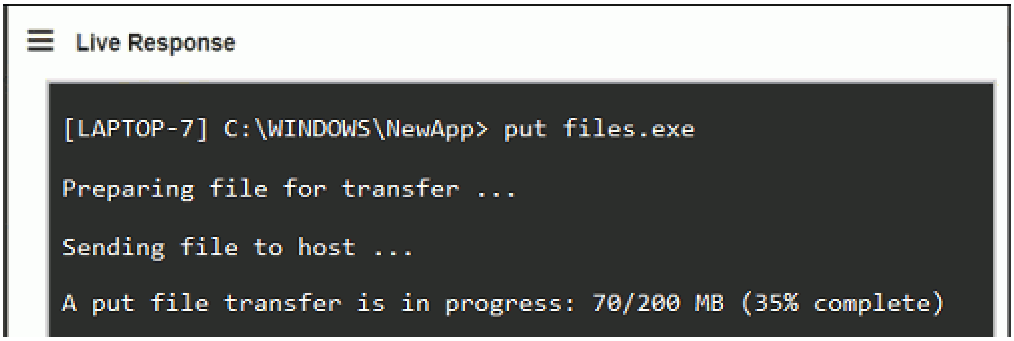


```
[LAPTOP-7] C:\WINDOWS\NewApp> get files\what.zip

 A get file transfer is in progress: 10/100 MB (10% complete)

                         *
```

- There are other status messages, such as “Preparing file for transfer.” If a transfer is small and completed almost immediately, this might be all you see, without byte or percentage numbers.



```
[LAPTOP-7] C:\WINDOWS\NewApp> put files.exe

Preparing file for transfer ...

Sending file to host ...

A put file transfer is in progress: 70/200 MB (35% complete)
```

- During the time that the transfer is in progress, users attaching to a session that has a transfer in progress see only numerical progress indicators.

- A successful transfer results in a “File transfer complete” message for all users who are attached to the session, and a command prompt returns.

Note While a file transfer is in progress, no other commands can be entered in the Live Response window.

End a Live Response Session

Perform this procedure to end a Live Response session.

Procedure

- In the Live Response command window, enter the `detach` command.

  The session with that computer ends and the general **[Live Response]#** prompt replaces the computer-specific prompt.

  Sessions also timeout after a lack of activity. The default timeout value is five minutes. You can change this value in the `CbLRSessionTimeout` setting in the `cb.conf` file. See the *Carbon Black EDR Server Configuration Guide*.

Registry Access in Live Response

In a Live Response session for a Windows sensor, the `reg` command provides direct access to the remote computer’s Windows Registry.

The syntax of the Live Response `reg` command is:

```
reg [action] [key or value] [options]
```

The following table shows the `reg` command actions and their options. These options are intended to mirror the Windows default `reg.exe` command syntax. For all `reg` command actions, key paths can take hive references in either short or long form: `HKLM` or `HKEY_LOCAL_MACHINE` .

Action	Description
query	Format: `reg query`*[key or value] [options]* Options: *(none)* – If no option switch is specified, query for the specified key **`-v`** – Query for the specified value For example: `reg query HKLM\Software\Microsoft\Windows\CurrentVersion\Run`
add	Format: `reg add`*[key] [options]* Options: **`-v`** – Value for the key to be added **`-d`** – Data for the key to be added **`-t`** – Type of the key to be added; accepted types are: ■ `REG_NONE` ■ `REG_BINARY` ■ `REG_SZ` ■ `REG_EXPAND_SZ` ■ `REG_MULTI_SZ` ■ `REG_DWORD` ■ `REG_DWORD_BIG_ENDIAN` ■ `REG_QWORD` For example: `reg add HKLM\Software\Microsoft\Windows\CurrentVersion\Run -v calc -t REG_SZ -d c:\windows\system32\calc.exe`
delete	Format: `reg delete`*[key or value] [options]* Options: *(none)* – If no option switch is specified, delete the specified key **`-v`** – Delete the specified value For example: `reg delete HKLM\Software\Microsoft\Windows\CurrentVersion\Run -v calc`

Live Response Detached Session Management Mode

You can enter Live Response without a specific session. In this mode, you can take certain actions that do not require access to an endpoint.

Actions include viewing active sessions or examining files that have been uploaded to the server as a result of a session. You can attach to (join) an existing session or create a new one.

Some commands in detached mode are accessible by users who do not have Global Administrator privileges, but most are not, and attempting to use them returns an error message in the command window.

To open a Live Response command window without a session, click **Go Live** on the navigation bar. The Live Response page appears. In this mode, the prompt in the command window shows **[Live Response]#** without the name of an endpoint.

The following table shows the available commands in Live Response Management Mode.

Command	Description
archive [id]	Obtain an archive (gzip tarball) of all the session data for the session whose ID is provided.
argparse	Test how Live Response parses CLI arguments. This command helps determine whether there are any interpretation issues.
attach [id]	Attach to the session whose ID is provided. The `session` command can be used to find the ID of an existing session or create a new one. A session must be in active or pending state to be attached.
clear	Clear the console screen. You can also use the `cls` command for this purpose.
files -s [id]	Perform actions over cache-stored files for the session whose ID is provided.
help	Show the commands available in this mode with a brief description of each.
help command	Show the description of the specified command with additional details (such as options) if available. For example: `help dir`
sensor [options]	List sensors that this Carbon Black EDR server manages. Options: `-i` *[1.2.3.4]* – Return all sensors with specified IP address. `-n` *[host_str]* – Return all sensors with matching host name. `-a` – Return all sensors. Searches are case-sensitive substring searches for both host name and IP address. You must use an option with this command. If both **`-n`** and **`-i`** are specified, only **`-i`** is used.
session	Manage Live Response sessions. With no argument, lists all open sessions and their ID numbers, which can be used with the `attach` command. Options: ■ `session new` *[id]* – Create a new session for the sensor whose ID number is provided. You must provide a sensor ID, not a session ID. ■ `session list`**`[-v]`** – List existing sessions. If the **`-v`** option is included, closed sessions are included. This option (without **`-v`**) is the default when no additional arguments are used. ■ `close` *[id]* – Close the session whose ID is provided.

Extending Live Response

Because the built-in commands in Live Response include `put` to put a file on the remote system and `exec` and `execfg` to execute processes on the system, responders can arbitrarily extend the capabilities of Live Response beyond the built-in commands.

For example, an investigator could take the following series of actions:

- Upload `yara.exe` and search memory for your custom yara signatures.
- Upload `winpmem.exe` and dump a memory image.
- Upload `sbag.exe` and parse the registry for Shellbags artifacts.
- Upload a custom PowerShell script and execute it with `powershell.exe` .

Although the library of built-in commands in Live Response will grow, it will never include every command for every situation. The ability to use `put`, `file`, and `create process` together assures that you have the freedom to add utilities you need for forensics and incident response. Additional capabilities are provided by a Live Response API. See Live Response API Reference.

Live Response Activity Logging and Downloads

Live Response activity is logged on both the Carbon Black EDR server running Live Response and the sensors that it accesses.

For any sensor that is accessed by Live Response, commands executed during the session are logged in the `sensor.log` file, which is located in the Carbon Black EDR sensor installation folder on the endpoint.

On the Carbon Black EDR server, Live Response activity can be reviewed in the following files:

- `/var/log/cb/liveresponse/debug.log` – Begin troubleshooting a Live Response issue here. This log contains debug information that is related to the functional operation of the Live Response components and communication between sensor and server.
- `/etc/cb/liveresponse-logger.conf` – You can change the level of information in the `debug.log` .
- `/var/log/cb/audit/live-response.log` – This file audits Live Response activity. It keeps a log of all commands that are executed on an endpoint, the sensor ID, IP address, and hostname of the endpoint, and the username and account of the user who executed each command.
- `/var/cb/data/liveresponse` – This directory stores “get” and “put” files. It also contains the output of all executed commands. For example, if you perform a process listing, the list goes into this directory in JSON format. If you download a file (for example, using the archive command), it appears in this directory (under `/tmp`) and on the host that is running the Carbon Black EDR browser.

You can change the length of time that Live Response data is retained by editing the `CbLrDefaultSessionTTLDays` parameter in the `cb.conf` file. By default, this setting is 7 days. See the *Carbon Black EDR Server Configuration Guide*.

Banning Process Hashes

This section describes hash banning in Carbon Black EDR.

A Carbon Black EDR investigation might reveal that known malware has been allowed to run on endpoints without being blocked. This could be because of a gap in updating your endpoint protection software or a more general gap in protection capabilities. Another possibility is that you receive notification of a threat not yet encountered on your endpoints, and you are not certain that you are fully protected against it.

While not intended to replace endpoint protection products, Carbon Black EDR provides a hash banning feature to prevent malware processes from running in the future. This feature will also terminate the process for a newly banned hash if it is running when the ban is created. You can use this feature to prevent further actions from a threat until your endpoint protection is able to do so.

Note The Carbon Black EDR banning feature identifies and bans processes based on their MD5 hash.

- Hash banning does not ban shared libraries, such as DLLs, SYSs, CPLs, and OCXs. You can follow the steps to ban these files, but it will have no effect.
- Banning does not use SHA-256 hashes, even if they are available.
- If an endpoint is restarted, any banned process that runs on restart will terminate as soon as the Carbon Black EDR sensor begins to run.

Creating Process Hash Bans

You can ban a process MD5 hash from several locations in the Carbon Black EDR console.

- The Binary Details page has a **Ban this hash** button if the binary is an EXE.
- The Process Analysis page has a **Ban this hash** command on the **Actions** menu.
- The Manage Banned Hashes page includes the **Ban More Hashes** button. You can click this button to specify one or more MD5 hashes to be banned. The Manage Banned Hashes page also has checkboxes that allow previously configured bans to be disabled and re-enabled.
- **Note** Banning a process hash without knowing the purpose of the process can have serious consequences. While Carbon Black EDR sensors prevent you from banning most critical processes, a user can ban a process that is required for proper operation of your computers or your business applications. Make sure that all Carbon Black EDR console users understand this before they use the banning feature.

Ban a Process MD5 Hash from the Process Analysis Page

Perform the following procedure to ban an MD5 hash.

Procedure

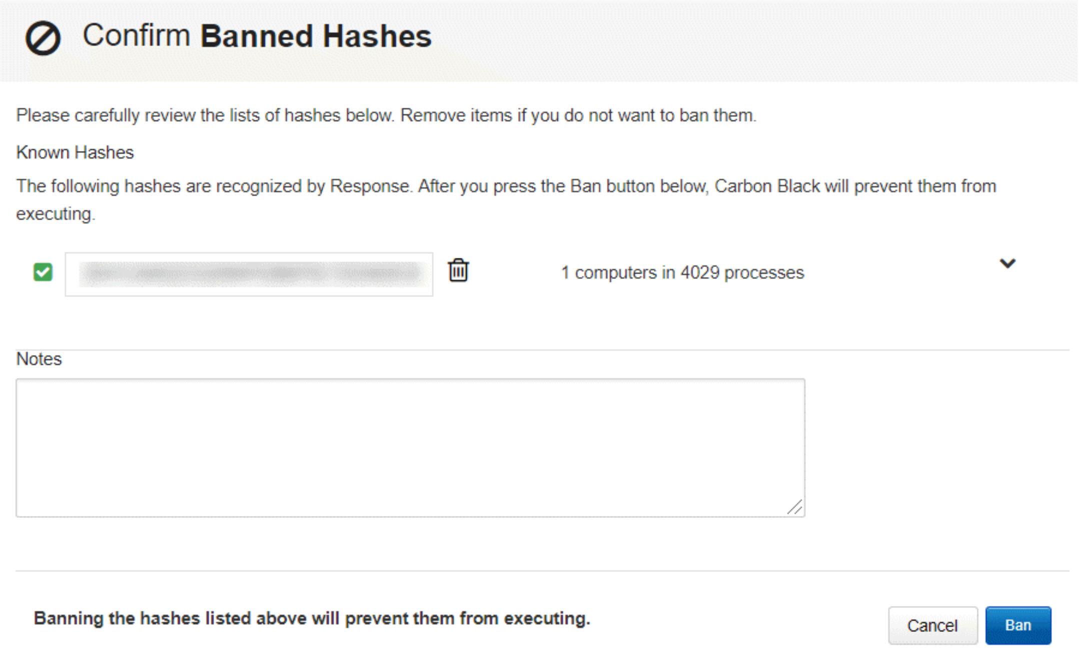
1. On the navigation bar, click **Process Search** and search for the process.

   See Chapter 10 Process Search and Analysis for instructions on searching.

2. Click the name of the process to ban.
3. On the Process Analysis page, click **Ban this hash**.

   Note This button only appears if the binary is an EXE. DLLs cannot be banned.

4 The Confirm Banned Hashes page appears and lists this hash, indicates whether it is known, and the number of endpoints at this site on which the hash for this process has been seen.



Confirm **Banned Hashes**

Please carefully review the lists of hashes below. Remove items if you do not want to ban them.

Known Hashes

The following hashes are recognized by Response. After you press the Ban button below, Carbon Black will prevent them from executing.

☑ 1 computers in 4029 processes

Notes

Banning the hashes listed above will prevent them from executing.

5 Add information in the **Notes** textbox to explain why you banned the hash. This can include a file name, threat report identification, or anything else that is helpful to examine the ban.

6 Click **Ban** to ban the hash. The ban is added to the list on the Manage Banned Hashes page, and is enabled. By default, the list is arranged in alphanumeric order by MD5 hash. See Manage Banned Hashes.

Note To terminate a hash ban action, click the **Trashcan** icon to delete the hash from the list. For single-hash-ban operations, click **Cancel** at the bottom of the page.

Ban a List of Hashes

Perform the following procedure to ban a list of process hashes.

You might have a list of process hashes from Carbon Black EDR or another source that you want to ban. For example, a warning from a threat intelligence source might provide a list of malware hashes. You can ban these processes in bulk on the Manage Banned Hashes page, including processes that are not yet observed by sensors reporting to your Carbon Black EDR server.

Procedure

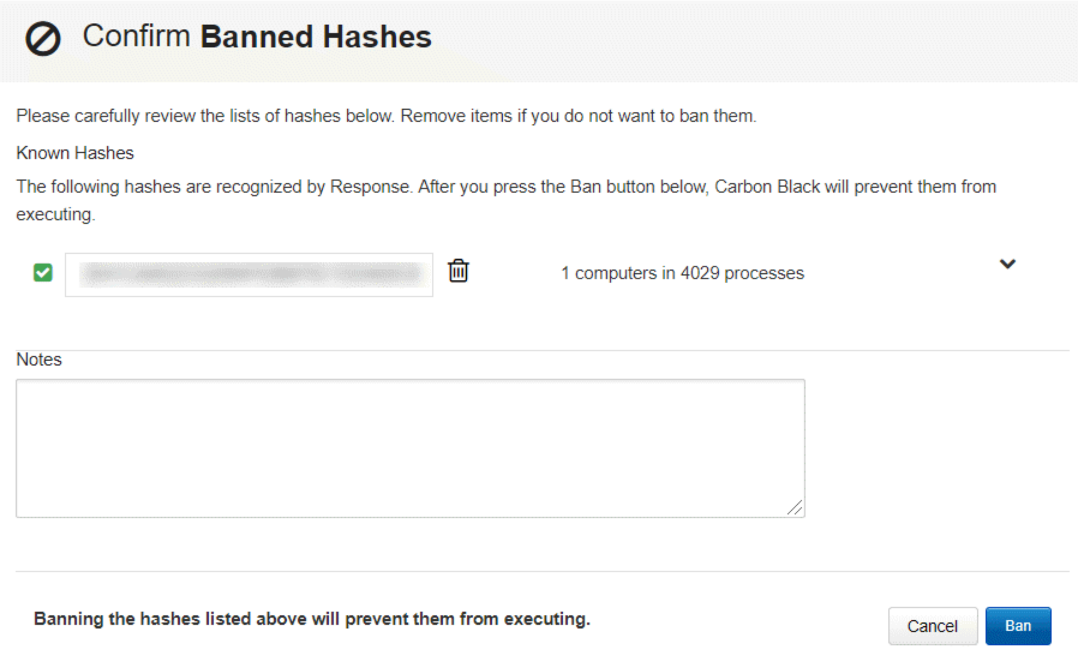
1 On the navigation bar, click **Banned Hashes**.

2 Click the **Ban More Hashes** button.

3 In the **MD5 hashes to ban** field, enter the MD5 hashes for the processes to ban. Each hash must be on its own line.

4 In the **Notes** field, provide information about why these hashes are being banned. You might also want to add names for each of the hashes, if available.

5 After you have entered the hashes and notes, click **Ban Hashes** to display the Confirm Banned Hashes page.



Confirm **Banned Hashes**

Please carefully review the lists of hashes below. Remove items if you do not want to ban them.

Known Hashes

The following hashes are recognized by Response. After you press the Ban button below, Carbon Black will prevent them from executing.

☑ 1 computers in 4029 processes

Notes

Banning the hashes listed above will prevent them from executing.

Note The page indicates whether the hash is already known to this Carbon Black EDR server, and if so, how many instances of the process have been seen and on how many endpoints. This page also allows you to modify the notes before finalizing the ban.

6 For more information about a known hash, click the down-arrow to the right of it.

7 If you decide not to ban a hash, click the **Trash** can icon next to it.

8 Click **Ban** to ban all listed hashes.

The bans are added to the list on the Manage Banned Hashes page and are enabled.

The **Notes** you entered appear next to each hash you included in this ban. By default, the list is arranged in alphanumeric order by MD5 hash.

See Manage Banned Hashes.

Managing and Monitoring Hash Bans

This section describes how to manage and monitor hash bans.

After you begin banning process hashes, several options are available for managing and making use of bans. You can:

- View data about bans on the Manage Banned Hashes page. See Manage Banned Hashes.

- View block events on the **Process Analysis** page. See Chapter 8 Responding to Endpoint Incidents.
- Enable alerts and syslog event recording for process blocks caused by bans, using the Banning Events feed on the Threat Intelligence Feeds page. See Enabling Alerts and Syslog Output for Banning Events.
- Monitor banning alerts on the Triage Alerts page. See Chapter 8 Responding to Endpoint Incidents.
- Enable and disable bans on the Manage Banned Hashes page. See Disable a Hash Ban.

Manage Banned Hashes

The Manage Banned Hashes page lets you add, manage, and get information about process hash bans created on your Carbon Black EDR server.

- **Table of Bans** – Any hash bans that have been created on your Carbon Black EDR server are listed in a table, including bans that are enabled and bans that are not currently enabled. An indicator at the top-right corner of the page shows the total number of bans (both enabled and disabled) that have been created.
- **Access to Additional Ban Information** – Some information about each ban is shown in the table rows, and additional information is available through drill-down features for each ban.
- **Toggling of Ban Status** – The status of each ban is displayed in the **Banned** column. You can enable or disable any ban.
- **Ban More Hashes** – This button opens the **Add Hashes to Ban List** dialog, where you can enter one or more hash values to create new bans.

The table of hashes lists each hash that has been created on this server. You can also search for hash bans by the MD5 hash of the process, and you can control the display of the entire table using the following controls:

- **View** – You can click different buttons in the **View** field to display **All** bans (the default), currently **Banned** hashes, and **Previously Banned** hashes (ban disabled).
- **Sort By** – You can sort the table by **MD5** hash (default), **Date Added** , or **User**. Radio buttons change the sort order from ascending to descending.

The following table describes fields on this page. The table data that reports on blocks caused by bans requires that the Banning Events feed on the Threat Intelligence Feeds page is enabled. (See Chapter 14 Threat Intelligence Feeds.)

Column	Description
Hash	The MD5 hash of the process that is or was banned. Clicking on the hash opens the Binary Details page for the hash.
Notes	Any user-created notes about the ban or hash.
Latest Block	The length of time since the process identified by the MD5 hash was blocked on a system reporting to the Carbon Black EDR server.

Column	Description
Total Blocks	The total number of times this process has been blocked by the ban.
Hosts w/ Blocks	The number of systems on which this process was blocked at least once. If a host name appears, clicking on it opens the Sensor Details page for that host.
Banned	This checkbox controls the status of the ban. When the box is checked, the ban is enabled. When the checkbox is not checked, the ban is disabled.
⌄ (more details)	Click the blue down arrow icon to expand the row of a hash ban to provide additional details.

When you expand the row for a ban using the blue down arrow, information about the ban and its process appears in the panel. You can also use navigation links to go to other pages for more information.

The following table describes the process hash ban details:

Column	Description
Hosts / Processes	Shows how many hosts have run the process identified by this MD5 hash and how many times the process ran before it was banned.
Meta data	The name of the Carbon Black EDR console user who created the ban, when the ban was added, and the date and time of the most recent block caused by the ban. Clicking the user name navigates to the table of users on the User Management page.
Hosts	The endpoints on which the process controlled by this ban has been blocked.
Notes	Any user-created notes about the ban or the hash. Notes can be edited.
View ban history	Opens a separate **Ban History** window that shows status changes (enabled, disabled) for the ban, who made them, and when they were made.
🔍(process search)	Click the blue magnifying glass icon to go to the Process Search page with the search results for this process.

Monitoring Banning Events

When a process is blocked because of a Carbon Black EDR hash ban, that is an indication that some user or process attempted an unwanted activity. Even though the activity was blocked, you might want to investigate the attempt.

Carbon Black EDR reports an event each time a hash ban attempts to block a process, even if the block fails (for example because of an attempt to block a critical system or Carbon Black EDR process). The event appears on the Process Analysis page of the parent process. If a process was running at the time a ban was created and then terminated by the ban, a banner reports that fact on the Process Analysis page.

Blocking events can also trigger alerts and be included in the syslog output from Carbon Black EDR. See Enabling Alerts and Syslog Output for Banning Events.

To view all block events for a parent process, on the Process Analysis page for the parent process, search for **blocked** in the **Type** filter. See Chapter 10 Process Search and Analysis.

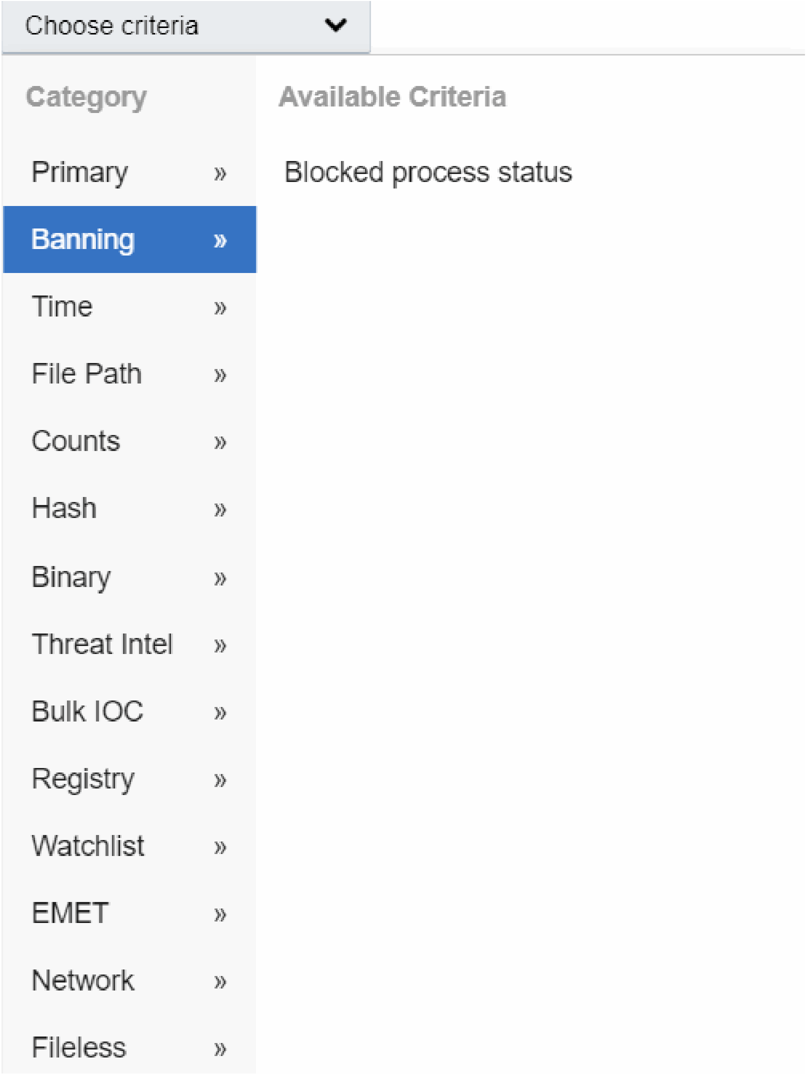
Search for Blocked Processes

The Process Search page lets you search for processes that have been affected by a process hash ban. This includes processes that were successfully blocked and those that could not be blocked for various reasons.

Procedure

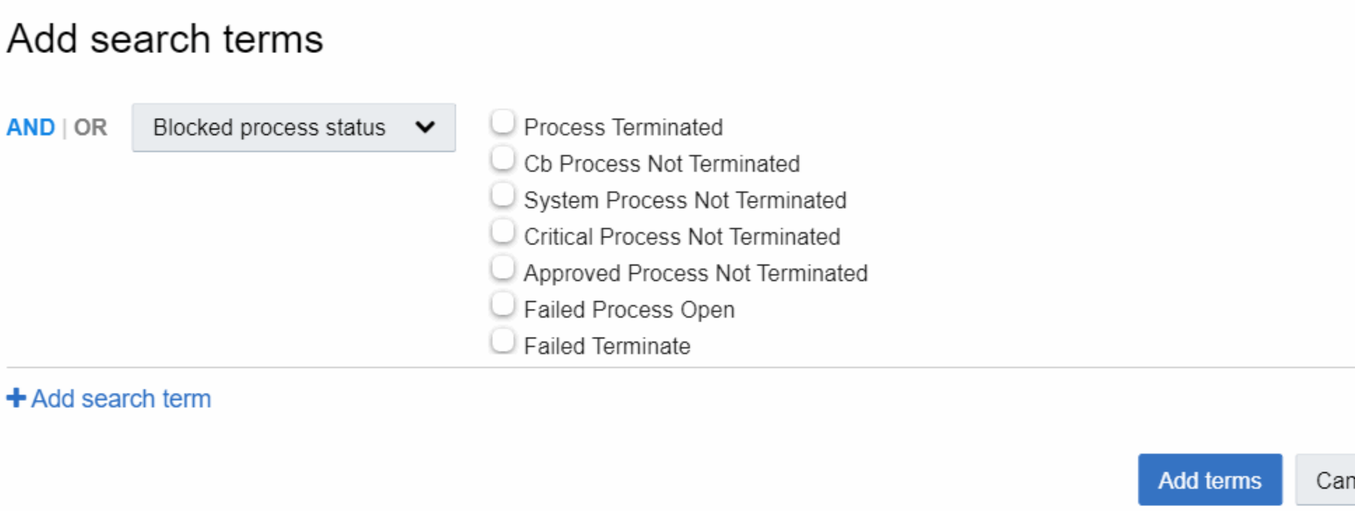
1 On the navigation bar, click **Process Search**.

2 Click **Add Search Terms** and then click **Choose Criteria**.



3 Click **Banning** and then click **Blocked process status**.

4 You can select blocked conditions to search for.



Add search terms

AND | OR Blocked process status

○ Process Terminated
○ Cb Process Not Terminated
○ System Process Not Terminated
○ Critical Process Not Terminated
○ Approved Process Not Terminated
○ Failed Process Open
○ Failed Terminate

To search for successful blocks only, select **Process Terminated** . To search for other block events, check the boxes for those events. When you have selected all relevant boxes, click **Add Terms** and run the search. The search results are updated to match your criteria .

Note You can manually enter one of the following queries for blocked processes:

- `block_status:processterminated`
- `processblock_count:[1 TO *]`

Enabling Alerts and Syslog Output for Banning Events

This section describes how to enable alerts and syslog output for banning events.

Unless banning is disabled entirely, process block events are sent to the Carbon Black EDR server and viewable on the Process Analysis page. (See Chapter 10 Process Search and Analysis.)

To configure alerts and syslog output for process blocks, a special **Banning Events** panel is available on the Threat Intelligence Feeds page. (See Chapter 14 Threat Intelligence Feeds.) This is not a feed in the normal sense because the events for blocks are sent to the server regardless of whether the feed is enabled. However, the feed must be enabled if you want to configure notifications for banning events.

The **Banning Events** feed is available by default and does not require enabling communication with Carbon Black Threat Intel.

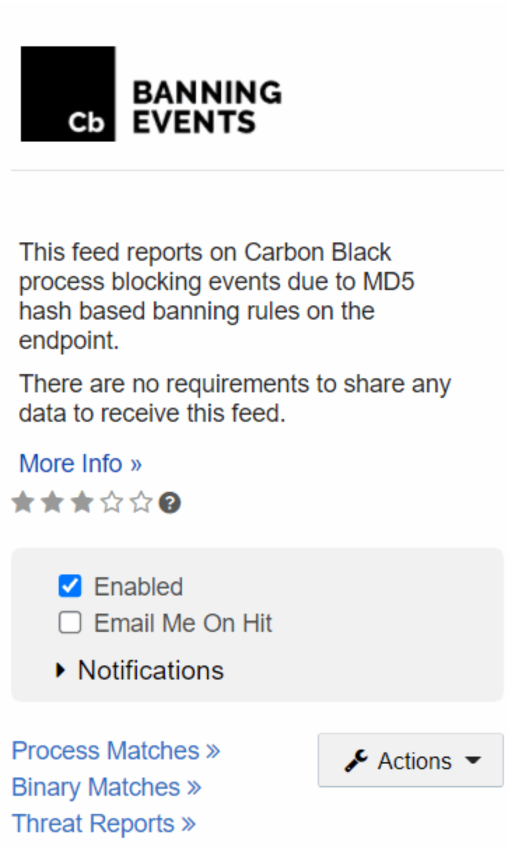
Enable Alerts and Syslog Output for Banning Events

Perform the following procedure to enable alerts and syslog output for banning events.

Procedure

1 On the navigation bar, click **Threat Intelligence**.

2 Locate the **Banning Events** feed.



This feed reports on Carbon Black process blocking events due to MD5 hash based banning rules on the endpoint.

There are no requirements to share any data to receive this feed.

☑ Enabled
□ Email Me On Hit

3 Click **Notifications** and select the notification types to create: **Create Alert** and/or **Log to Syslog**.

4 To receive email when a block event occurs, select **Email Me On Hit**.

View Banned Hash Alerts

Perform the following procedure to view banned hash alerts.

Procedure

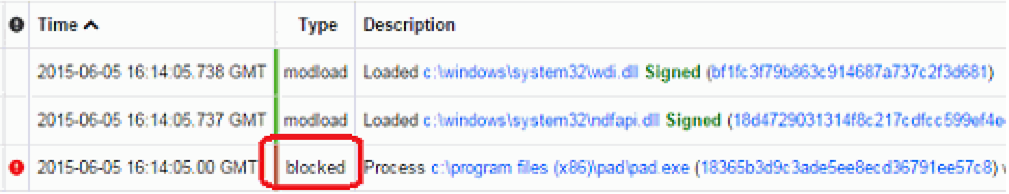
1 On the navigation bar, click **Triage Alerts**.

See Managing Alerts on the Triage Alerts Page.

2 In the search box for the **Feed** filter, enter `cbbanning` and press `Enter`. If it already appears on the list, click **cbbanning**.

In addition to triggering alerts (if enabled), processes that are blocked due to a hash ban generate events that appear on the Process Analysis page.

For example, if you receive a Process Blocking alert for a process, the Process Analysis page for the parent process appears and includes a **blocked** event.



●	Time ^	Type	Description
	2015-06-05 16:14:05.738 GMT	modload	Loaded c:\windows\system32\wdi.dll Signed (bf1fc3f79b863c914687a737c2f3d681)
	2015-06-05 16:14:05.737 GMT	modload	Loaded c:\windows\system32\ndfapi.dll Signed (18d4729031314f8c217cdfcc599ef4e
●	2015-06-05 16:14:05.00 GMT	blocked	Process c:\program files (x86)\pad\pad.exe (18365b3d9c3ade5ee8ecd36791ee57c8)

Disable a Hash Ban

After a hash ban is created, it always appears on the Manage Banned Hashes page. You can turn bans on and off.

Procedure

1. On the navigation bar, click **Banned Hashes**.
2. Locate the row for the hash ban to remove, and deselect the check box in the **Banned** column.

Disable Hash Bans in a Sensor Group

You can disable banning on all hosts in a specified sensor group. In this case, any process hash bans that are configured on the server are ignored by sensors in that group, and no processes are blocked by Carbon Black EDR on those sensors.

Procedure

1. On the navigation bar, click **Sensors**.
2. In the **Groups** panel, click the sensor group to exempt from hash bans.
3. At the top of the **<`name`> Group** panel, click **Edit** and then click **Advanced** to expand the advanced sensor group settings.
4. Deselect **Process Banning**.
5. Click **Save Group**.

Live Query

9

This section describes Carbon Black EDR Live Query, and how to create and run queries against your endpoints.

Live Query can expose an operating system as a high-performance relational database — you can write SQL-based queries that explore operating system data to analyze security vulnerabilities. Live Query is based on osquery, which is an open source project that uses a SQLite interface. Live Query is released with Carbon Black EDR 7.2, and requires the Carbon Black EDR Windows sensor 7.1.0 or higher.

All users can view queries on the sensors for which they have **View** permissions. To execute a Live Query, an analyst must have the **Execute Live Query** enhanced permission. See Adding Enhanced Permissions for Analysts.

Read the following topics next:

- Enable or Disable Live Query
- Creating and Running a Query
- Query Results

Enable or Disable Live Query

Live Query is disabled by default. Carbon Black EDR Global Administrators and Carbon Black Hosted EDR Administrators can enable or disable Live Query.

Procedure

1 Click ***Username*** **> Settings > Advanced Settings**.

2 Check the **Enable Live Query** checkbox to enable Live Query, or uncheck the checkbox to disable it.

3 Confirm your selection.

4 Click **Save Changes**.

Creating and Running a Query

This section describes how to create and run a query.

On the navigation bar, click **Live Query**. The Live Query page shows any currently running query, a completed query, or a blank page depending on the status of the most recently run query.

You can run only one query at a time. If you run a new query, previous query results are discarded.

The maximum number of sensors you can target for a single query is 200. If you select more than 200 sensors, only the first 200 sensors receive the query, based on the 200 sensors that have most recently checked in.

There are two ways to run a query – you can use a preformed recommended query that Carbon Black EDR provides, or you can write your own SQL query.

Recommended queries are organized into the following categories:

- Compliance – verify that hosts are in compliance with common security-related requirements
- IT hygiene – check the status of credentials, certificates, and accounts on your hosts
- Threat hunting – check for commonly used threat techniques on your hosts
- Vulnerability management – discover which patches, drivers, chrome extensions, etc. are active on your hosts

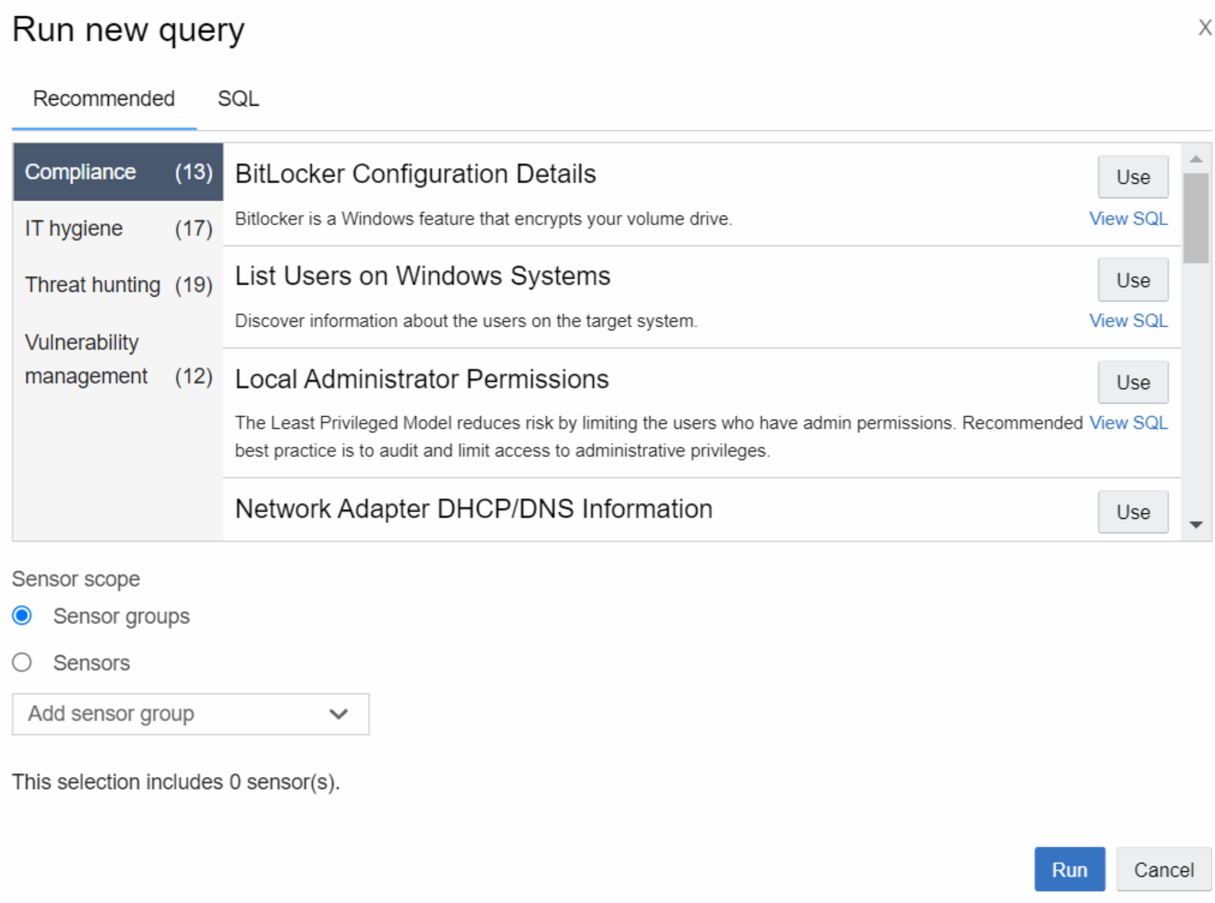
Run a Recommended Query

Perform the following procedure to run a recommended query.

Procedure

1. On the navigation bar, click **Live Query**.
2. On the Live Query page, click **Run New Query**.

3 Click the **Recommended** tab if it is not already selected.



Run new query

Compliance (13)

IT hygiene (17)

Threat hunting (19)

Vulnerability management (12)

BitLocker Configuration Details

Bitlocker is a Windows feature that encrypts your volume drive.

List Users on Windows Systems

Discover information about the users on the target system.

Local Administrator Permissions

The Least Privileged Model reduces risk by limiting the users who have admin permissions. Recommended best practice is to audit and limit access to administrative privileges.

Network Adapter DHCP/DNS Information

Sensor scope

● Sensor groups

○ Sensors

Add sensor group

This selection includes 0 sensor(s).

4 To optionally view the query SQL code, click **View SQL**.

5 Click **Use** next to the recommended query name. The query appears in the **SQL** tab so that you can modify it, or run it as is.

6 Identify the endpoints to receive the query.

You can select endpoints by sensor group, or you can select individual sensors by host name. A message shows the number of sensors that you have selected. Note that the number of sensors that are shown includes all sensors, not just the sensors that are Live Query-compatible.

7 Click **Run**. The selected sensors pick up the query the next time the sensors check in with the server.

Run your own SQL Query

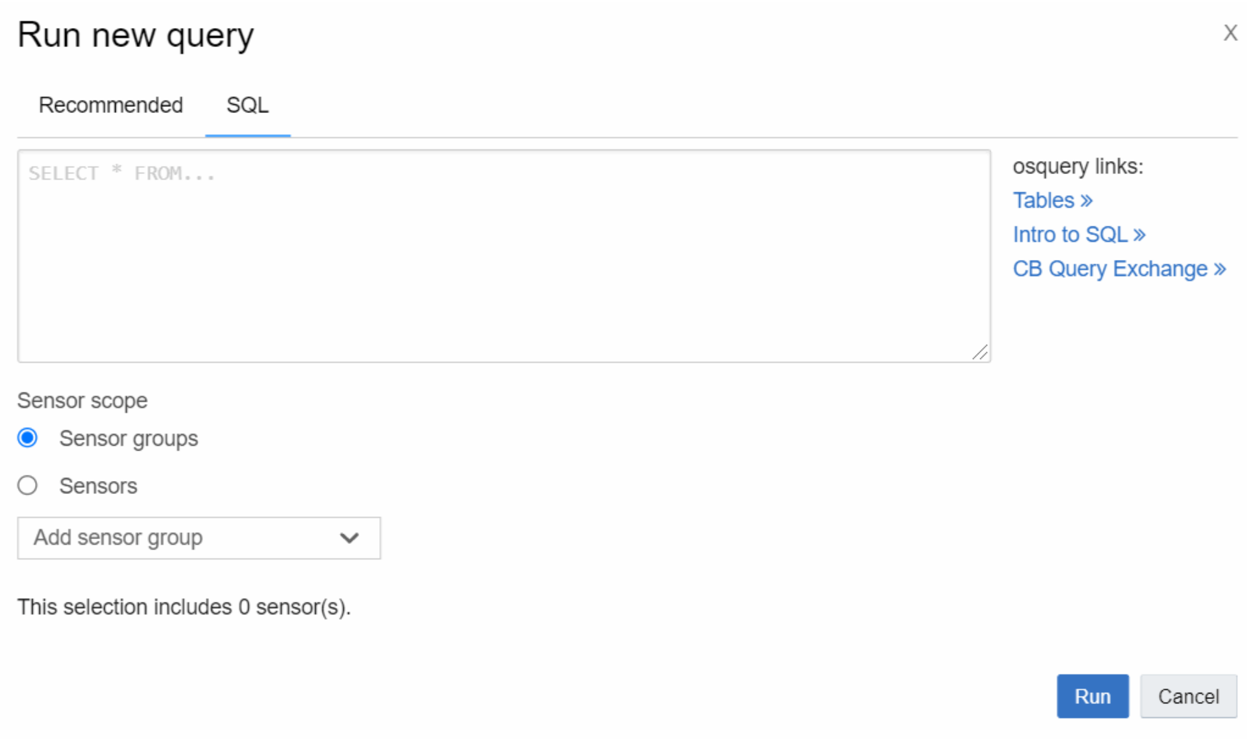
Perform the following procedure to run your own SQL query.

Procedure

1 On the navigation bar, click **Live Query**.

2 On the Live Query page, click **Run New Query**.

3 Click the **SQL** tab.


Run new query

Sensor scope

● Sensor groups

○ Sensors

This selection includes 0 sensor(s).

4 In the text box, type your SQL query. For help writing a query, click the provided links:

- Tables
- Intro to SQL
- CB Query Exchange

5 Identify the endpoints to receive the query. You can select endpoints by sensor group, or you can select individual sensors by host name. A message displays the number of sensors that you have selected. Note that the number of sensors that are shown includes all sensors, not just the sensors that are Live Query-compatible.

6 Click **Run**. The selected sensors pick up the query the next time the sensors check in with the server.

Note Double quotation marks produce errors in your SQL query. Use single quotation marks instead. Chained queries (separated by ;) are not supported.

Query Results

Query results automatically fill the Results table on the Live Query page.

Because the request is asynchronous, you do not have to stay on the page to see the results. You can leave the Live Query page and come back later to see the results.

If the current query is too long to be displayed on a single line, click the diagonal arrow next to the query to see the entire query.

Query results are returned in three states:

- Completed – the query completed successfully
- Truncated – returned data exceeds the acceptable length
- Error – incorrect SQL syntax, unavailable osquery table, etc.

You can only see results for sensors that you have permissions to view. If you run a query and the results contain sensors to which you have no access, you cannot see their results. However, the count of sensors that responded to the query (on the top right of the page) includes them.

You can filter the Results table by computer name. The Results table always displays the following two columns:

Column	Description
Computer Name	Name and query status of the endpoint on which the query ran.
Time Received	The time (day) that the query ran on the endpoint.

The remaining displayed columns depend on the query itself (see Tables). Query results reside in memory and are retained until a new query is run or services are restarted.

Export Live Query Results

You can export Live Query results into a CSV file.

Procedure

1. On the navigation bar, click **Live Query**.
2. On the Live Query page, click **Export** and then click **Export all**.

   A CSV file is downloaded into the `C:\Users\`*`username`*`\Downloads` folder.

Process Search and Analysis

10

This section describes how to perform detailed process searches and in-depth analyses of processes in search results.

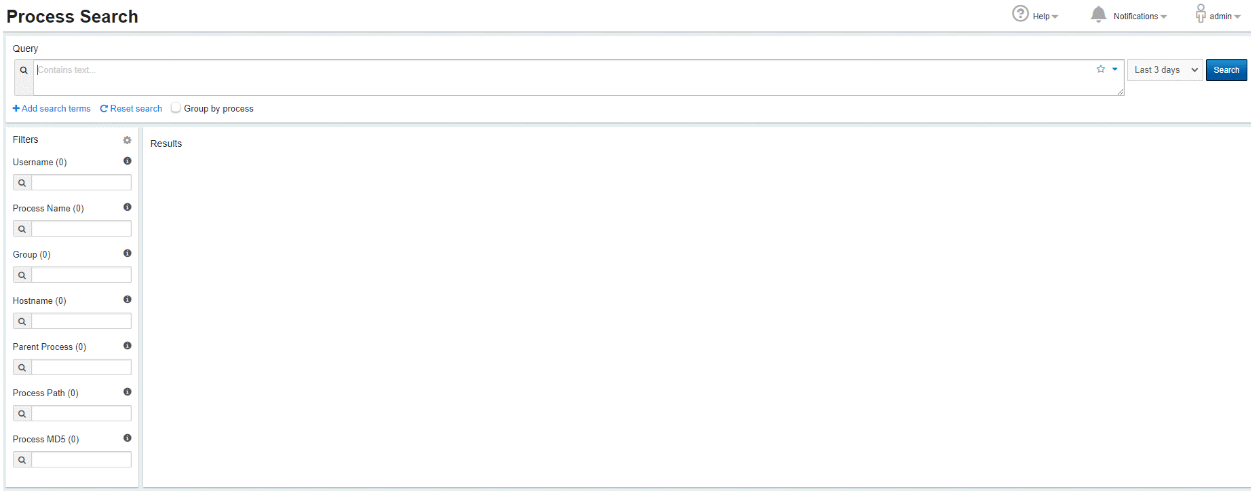
Read the following topics next:

Overview of Process Search

This section describes how to perform basic process searches using search strings and predefined search criteria.

When you become aware of an incident that could be a threat, you can search all your systems and endpoints for processes that have Indicators of Compromise (IOCs). For example, you might receive a call reporting unusual software behavior or an alert from a threat intelligence report or watchlist. Carbon Black EDR sensors automatically collect data so that you can immediately start analyzing issues and finding solutions.

Use the Process Search page to begin investigating potential threats. On the navigation bar, click **Process Search**.



Process Search Time Filter

You can specify that the results show only processes that appear in events that occurred within a specified time period.

Select a time filter from the dropdown menu to the right of the **Search** field. If you select **Custom**, you are presented with a calendar by which you can set a custom time period.

Process Search Filters

Search filters provide ways to specify and narrow a search. Each filter represents terms that exist in various fields, such as **Process Name** or **Hostname**.

The percentage next to each term shows the relative frequency with which the term appears in the field.

No content appears in the search filters until after you have initiated a search. The search filters populate according to their match to the search results.

Enable or Disable Filters

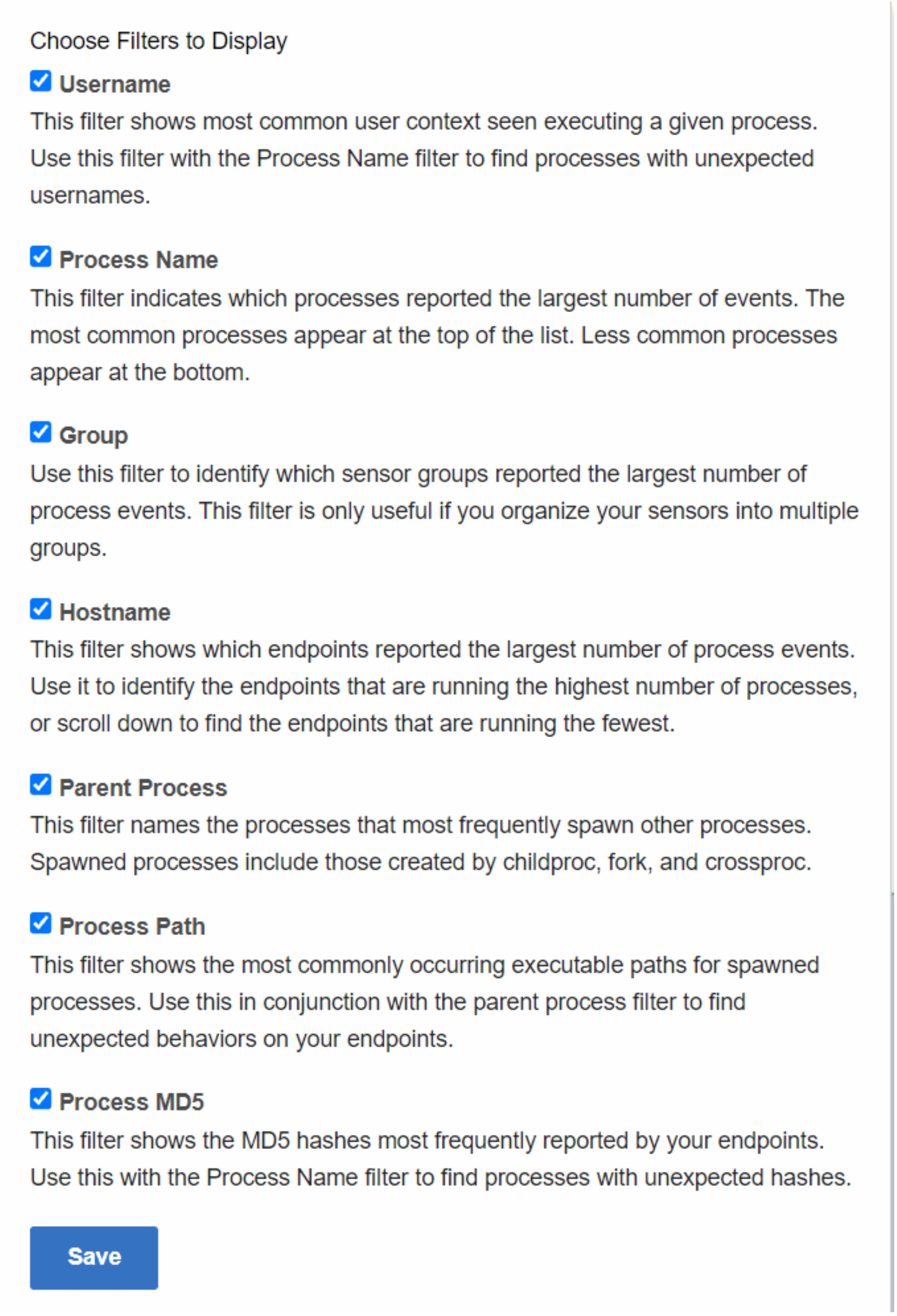
Perform the following procedure to display only certain search filters on the Process Search page.

Procedure

1 On the navigation bar, click **Process Search**.

2 Click the **Gear** icon to the right of **Filters**.

3 Select checkboxes to enable or disable the filters to display.



Choose Filters to Display

☑ **Username**

This filter shows most common user context seen executing a given process. Use this filter with the Process Name filter to find processes with unexpected usernames.

☑ **Process Name**

This filter indicates which processes reported the largest number of events. The most common processes appear at the top of the list. Less common processes appear at the bottom.

☑ **Group**

Use this filter to identify which sensor groups reported the largest number of process events. This filter is only useful if you organize your sensors into multiple groups.

☑ **Hostname**

This filter shows which endpoints reported the largest number of process events. Use it to identify the endpoints that are running the highest number of processes, or scroll down to find the endpoints that are running the fewest.

☑ **Parent Process**

This filter names the processes that most frequently spawn other processes. Spawned processes include those created by childproc, fork, and crossproc.

☑ **Process Path**

This filter shows the most commonly occurring executable paths for spawned processes. Use this in conjunction with the parent process filter to find unexpected behaviors on your endpoints.

☑ **Process MD5**

This filter shows the MD5 hashes most frequently reported by your endpoints. Use this with the Process Name filter to find processes with unexpected hashes.

Disabling a filter removes it from view, and if it is part of the search query, those pieces of the query are removed. Enabling a filter places it back into view.

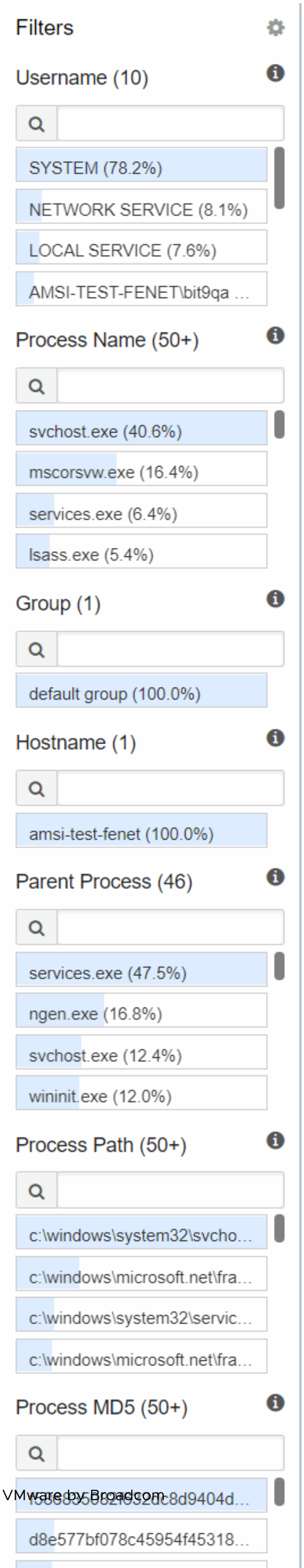
4 Click **Save**.

Select Multiple Filter Rows

You can select specific filter rows within a filter table by 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 in yellow. You can click a filter row to deselect it.


Filters
Username (10)
SYSTEM (78.2%)
NETWORK SERVICE (8.1%)
LOCAL SERVICE (7.6%)
AMSI-TEST-FENET\bit9qa …
Process Name (50+)
svchost.exe (40.6%)
mscorsvw.exe (16.4%)
services.exe (6.4%)
lsass.exe (5.4%)
Group (1)
default group (100.0%)
Hostname (1)
amsi-test-fenet (100.0%)
Parent Process (46)
services.exe (47.5%)
ngen.exe (16.8%)
svchost.exe (12.4%)
wininit.exe (12.0%)
Process Path (50+)
c:\windows\system32\svcho…
c:\windows\microsoft.net\fra…
c:\windows\system32\servic…
c:\windows\microsoft.net\fra…
Process MD5 (50+)
f368395682f832dc8d9404d…
d8e577bf078c45954f45318…

Filter Row Percentages

Filter row percentages indicate the percentage of processes that have occurred in a particular filter. This is always equivalent to 100% when you add up all filter rows in a filter.

The top row in a filter has occurred more than any other process within that filter.

Filter Search Fields

Each filter contains a **Search** field into which you can enter search parameters to refine search results.

Process Search Field

You can manually enter keyword searches or predefined search criteria in the **Search** field.

As you enter search criteria, the correct syntax appears. If you do not enter any search criteria, the system runs a search using *.* This displays every process that has executed. The processes are ranked according to the process start time — the most frequently executed processes are at the top. You can sort the results according to the count of events or last update time.

Note The **Search** field is expandable: click and drag the bottom right corner to display long queries.

The **Search** field and criteria fields can be used independently or in combination. When used in combination, the system combines them using an `AND` operator.

Clicking the blue **Search** button executes a search using selected parameters. When viewing events on the Process Search page, the time displayed next to the process is the time that the sensor recorded the event — not the time that the server received the event.

Saving and Executing Saved Process Searches

This topic describes how to save, execute, edit, and clear Process searches.

Save a Process Search

You can save frequently executed searches. Perform the following procedure to save a search.

Procedure

1 On the navigation bar, click **Process Search**.

2 Click 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.

Execute a Saved Search

Perform the following procedure to execute a saved Process Search.

Procedure

1. On the navigation bar, click **Process Search**.
2. Click the down arrow to the right of the **Favorite (star)** icon and select the saved search.

Edit a Saved Search

Perform the following procedure to edit a saved Process Search.

Procedure

1. On the navigation bar, click **Process Search**.
2. Click the down arrow to the right of the **Favorite (star)** icon and select the saved search.
3. Click the pencil icon.

Clear a Saved Search

Perform the following procedure to clear a saved Process Search.

Procedure

1. On the navigation bar, click **Process Search**.
2. Click the down arrow to the right of the **Favorite (star)** icon and select the saved search.
3. Click the trashcan icon.

Clear all Saved Searches

Perform the following procedure to clear all saved Process searches.

Procedure

1. Click ***Username*** > **My Profile**.
2. Click **Clear Preferences**.
3. Click **Save changes**.

Add Process Search Terms

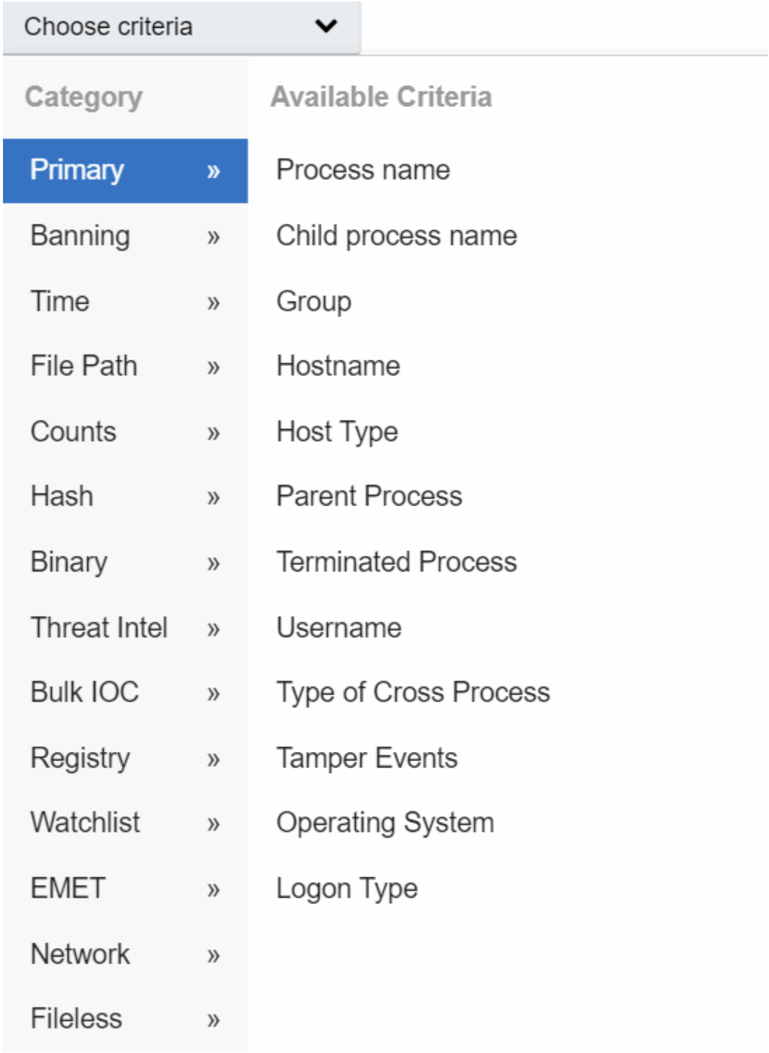
Perform the following procedure to add Process Search terms.

Process searches explicitly support `AND/OR` operators. You can select from an array of filters to form your search using these `AND/OR` operators.

Procedure

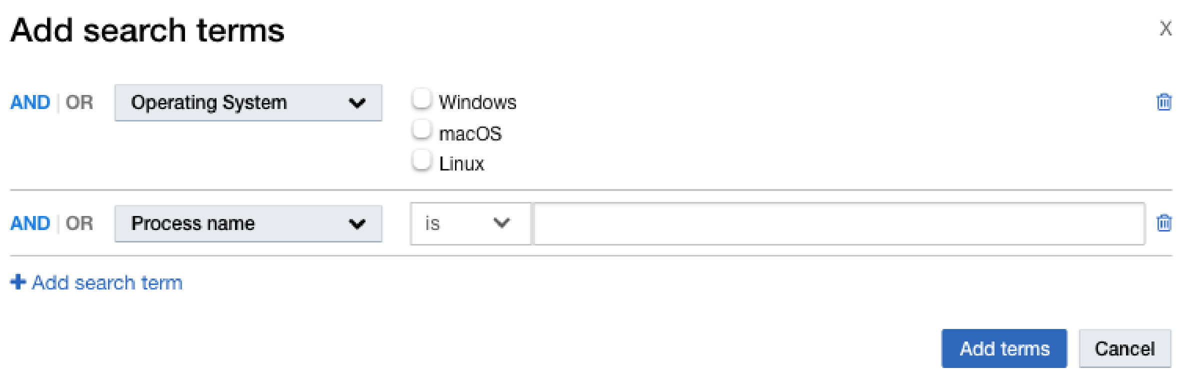
1. On the navigation bar, click **Process Search**.
2. Click **Add Search Terms** to add search terms (in the form of `AND/OR` operators).

3 Select a search term type from the **Choose Criteria** drop-down menu.



4 Click **Add Search Term** to add terms. When you are finished, click **Add Terms**. The search terms are validated, and you are notified if there is an error.

For example, the following search terms display processes on Windows endpoints that have been received in the last 30 minutes.



Add search terms

AND | OR Operating System

○ Windows
○ macOS
○ Linux

AND | OR Process name is

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

To reset and remove all search terms, click **Reset Search**.

Search for Events

You can search for all events in the **Events List** on the Process Analysis page.

You can search for any event by typing the search text query into the **Search** text box and pressing **`Enter`**. The search API returns matching events.

Standard search terms apply. Be aware of the following criteria:

- Click the **Later Events** and **Earlier Events** buttons to return the respective elements. If it takes longer than the time specified in the `ProcessAnalysisEventSearchTimeout` parameter in the `cb.conf` file to return the events, the API returns a timeout and displays a timeout error message. The respective button for that search is disabled.
- If you apply a filter from the **Filter** panel on the left, the events that satisfy the filter and search query are returned. If you click **Later Events** or **Earlier Events** and it takes longer than the time specified in the `ProcessAnalysisEventSearchTimeout` parameter to return the events, the API returns a timeout, a timeout message displays, and the **Later Events** or **Earlier Events** button for which the timeout occurred is disabled. The default timeout is 30 seconds.
- In the case of a timeout, you can reset the search criteria and filter to perform a new search. You can modify or reset the query by altering or deleting the text in the **Search** text box and pressing **`Enter`**.

Note Any records that were fetched until the timeout occurred are returned and displayed.

See `ProcessAnalysisEventSearchTimeout` in the *Carbon Black EDR Server Configuration Guide*.

Process Search Results

This topic discusses Process Search results.

- To combine results with the same process into groups, select **Group by process** on the Process Search page.
- If no search results match your search criteria, the Process Search page displays a message that recommends you widen your search by deselecting filters.
- If search results are too large, you can use filters to narrow your search, or you can view all results.
- A **Get Comprehensive Results** button appears on the Process Search page if a search query spans both current data and older data that was collected prior to Carbon Black EDR version 6.1, and if the query has complex search terms that require special processing on the server.
  - If you do not request comprehensive results, the server returns correct search results for old data, but results might be incomplete for data that was collected prior to Carbon Black EDR version 6.1.
  - If you request comprehensive results, the server returns full search results for current data, but excludes data that was collected prior to Carbon Black EDR version 6.1.

Example Process Search

This topic describes how to perform a Process Search.

Procedure

1. On the navigation bar, click **Process Search**.
2. Enter search criteria by performing one (or combining all) of these tasks:

   a. Enter keyword searches or predefined search criteria in the **Search** field.

   b. Click **Choose Criteria** to display a list of searchable criteria. Select the search criteria, such as **Banning > Blocked Process Status > Process Terminated**. Then, click **Add Search Term** to add an additional set of search criteria, such as **Time > Process Start > In the last 90 minutes**. Repeat this process to add search criteria.
3. Click **Search**.

   The search results appear in the **Results** table. See Process Search Results Table.

   Note To perform advanced queries, see Chapter 12 Advanced Search Queries.

Managing High-Impact Queries

Certain process searches can cause significant performance issues in Carbon Black EDR.

Two types of searches that can have a negative impact are:

- Searches with leading wildcards

- Searches with binary terms (which require a join between the process and module databases) if you have very large modules cores; see Searching with Binary Joins.

Beginning with Carbon Black EDR version 6.2.3, these searches are blocked by default when executed through the console. However, there are options in both the console interface and the server configuration file (`cb.conf`) for blocking and unblocking these types of process searches.

The blocking features, both from `cb.conf` and through the console, apply only to interactive searches in the console. Searches executed via the API, existing watchlists or feeds are not impacted by these settings.

For more information about `cb.conf`, see the *Carbon Black EDR Server Configuration Guide*.

Responding to Blocked Searches

This topic describes how to respond to blocked process searches.

Users attempting blocked search types will see a message describing why the search was blocked. If you determine that a setting is preventing searches that you expect to succeed, you can either modify the settings or modify your search. If you unblock one of the search types, monitor the performance impact to determine whether you can operate successfully in that mode.

You can also reconfigure the search to avoid the blocked condition. See Chapter 12 Advanced Search Queries for more information about creating complex process searches.

Block or Allow High Impact Process Searches in the Console

The settings for blocking searches with leading wildcards and searches with joins of large module cores are on the **Advanced Settings** tab on the Settings page. By default, both process search settings are checked, thereby blocking these searches.

Prerequisites

You must have Global Administrator (Carbon Black EDR) or Administrator (Carbon Black Hosted EDR) privileges to change these settings. In addition, if the settings are controlled by a `cb.conf` file configuration, they will be grayed out and unavailable for editing. In this case, see Process Search Settings in cb.conf.

Procedure

1 Log in to as a Global Administrator or Carbon Black Hosted EDR as an Administrator.

2 Click ***Username*>** **Settings** and then click **Advanced Settings**.

3 Under **Process Search Settings**, check (or uncheck) the box for the search type to block or unblock.

4 Click the **Save changes** button.

Process Search Settings in cb.conf

Two settings in the `cb.conf` file affect whether process searches with possibly significant performance impact are blocked, allowed, or configurable through the console.

- `ForceBlockLeadingWildcardsInSearchTo` (interacts with Block Searches with Leading Wildcards in the console).
- `ForceBlockCoreJoinsInSearchTo` (interacts with Block Searches that included Binary Metadata with Large Binary Stores in the console).

By default, neither of these settings are present in the `cb.conf` file.

If a setting is `True`, process searches in the relevant category are **blocked**, and no user, including a Global Administrator, can change this setting through the console.

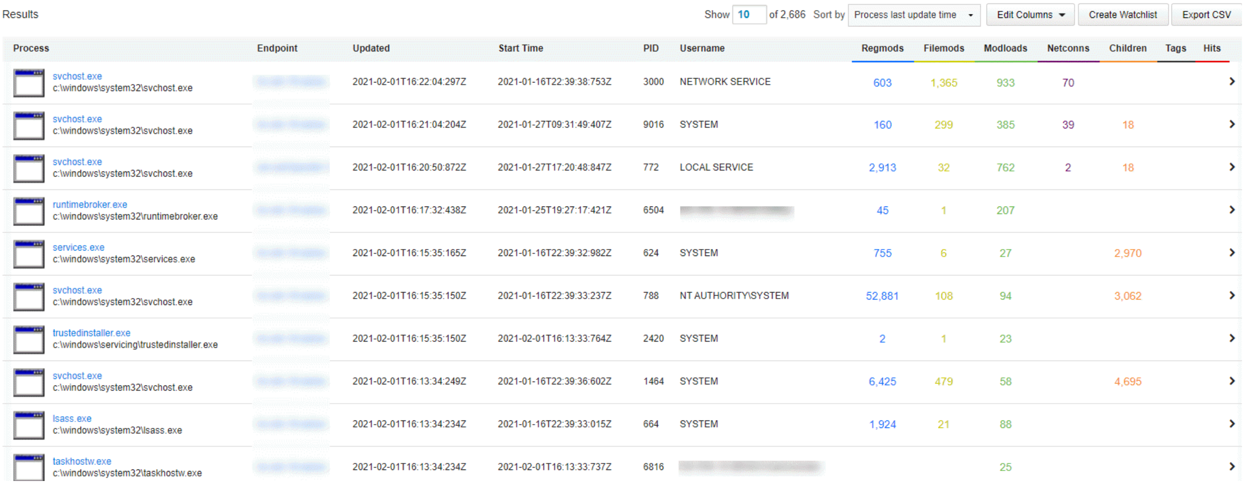
If a setting is `False`, process searches in the relevant category are **allowed**, and no user, including a Global Administrator, can change this setting through the console.

A third setting in `cb.conf` , `ModuleCoreDocumentCountWarningThreshold` , sets the number of module core documents that is considered large enough to be blocked when **Block Searches that included Binary Metadata with Large Binary Stores** is activated. By default, this setting has a value of ten million.

See the *Carbon Black EDR Server Configuration Guide* for information about editing `cb.conf`.

Process Search Results Table

The Process Search Results table describes executed processes that match the Process Search criteria.



Results

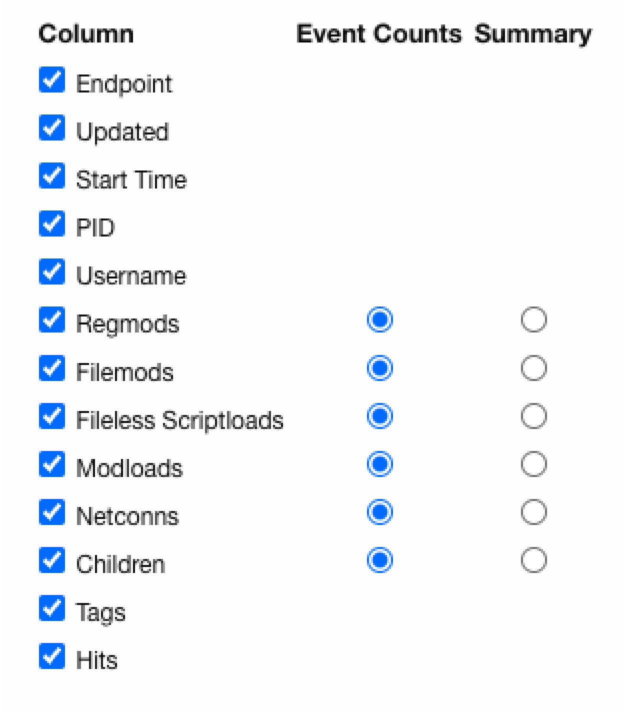
Process	Endpoint	Updated	Start Time	PID	Username	Regmods	Filemods	Modloads	Netconns	Children	Tags	Hits	
svchost.exe c:\windows\system32\svchost.exe	[redacted]	2021-02-01T16:22:04:297Z	2021-01-16T22:39:38:753Z	3000	NETWORK SERVICE	603	1,365	933	70				›
svchost.exe c:\windows\system32\svchost.exe	[redacted]	2021-02-01T16:21:04:204Z	2021-01-27T09:31:49:407Z	9016	SYSTEM	160	299	385	39	18			›
svchost.exe c:\windows\system32\svchost.exe	[redacted]	2021-02-01T16:20:50:872Z	2021-01-27T17:20:48:847Z	772	LOCAL SERVICE	2,913	32	762	2	18			›
runtimebroker.exe c:\windows\system32\runtimebroker.exe	[redacted]	2021-02-01T16:17:32:438Z	2021-01-25T19:27:17:421Z	6504	[redacted]	45	1	207					›
services.exe c:\windows\system32\services.exe	[redacted]	2021-02-01T16:15:35:165Z	2021-01-16T22:39:32:982Z	624	SYSTEM	755	6	27		2,970			›
svchost.exe c:\windows\system32\svchost.exe	[redacted]	2021-02-01T16:15:35:150Z	2021-01-16T22:39:33:237Z	788	NT AUTHORITY\SYSTEM	52,881	108	94		3,062			›
trustedinstaller.exe c:\windows\servicing\trustedinstaller.exe	[redacted]	2021-02-01T16:15:35:150Z	2021-02-01T16:13:33:764Z	2420	SYSTEM	2	1	23					›
svchost.exe c:\windows\system32\svchost.exe	[redacted]	2021-02-01T16:13:34:249Z	2021-01-16T22:39:36:602Z	1464	SYSTEM	6,425	479	58		4,695			›
lsass.exe c:\windows\system32\lsass.exe	[redacted]	2021-02-01T16:13:34:234Z	2021-01-16T22:39:33:015Z	664	SYSTEM	1,924	21	88					›
taskhostw.exe c:\windows\system32\taskhostw.exe	[redacted]	2021-02-01T16:13:34:234Z	2021-02-01T16:13:33:737Z	6816	[redacted]			25					›

Process Search Results Table Options

This topic describes options for the Process Search Results table.

The following options appear above the Process Search Results table:

- **Show** – Adjust the maximum number of search results that display on a page. The default setting is ten results per page.
- **Sort by** – Sort search criteria by the following options:
  - **None**
  - **Process last update time**
  - **Process start time**
  - **Process name**
  - **Network connections**
  - **Registry modifications**
  - **File modifications**
  - **Binary loads**
- **Edit Columns** – Select which columns are visible in the search results. You can also choose whether to show event counts or summary information.



Column	Event Counts	Summary
☑ Endpoint		
☑ Updated		
☑ Start Time		
☑ PID		
☑ Username		
☑ Regmods	●	○
☑ Filemods	●	○
☑ Fileless Scriptloads	●	○
☑ Modloads	●	○
☑ Netconns	●	○
☑ Children	●	○
☑ Tags		
☑ Hits		

- **Create Watchlist** – Create a watchlist that is based on the current query string. A watchlist is a saved search that you can use to track specific IOCs. See Chapter 19 Watchlists.
- **Export CSV** – Export the first 1000 process search results to a CSV file for reporting, retention, or compliance. Each row contains a URL to access the details of each result.

- **Note** To export more than 1000 rows, you must configure API functionality to capture and save the data. See the Carbon Black Developer Network at https://developer.carbonblack.com/reference/enterprise-response/.

Process Search Results Table Row Details

This topic describes rows in the Process Search Results table.

On each row within the Process Search Results table, the following information appears:

Title	Description
Process	The icon of the process or program that was executed and the name of the executable file that was run; for example, `notepad.exe`. The file system path from which the process was executed appears.
Endpoint	The endpoint that is associated with the result.
Updated	The timestamp when the process was last updated.
Start Time	The timestamp when the process started.
PID	The Process ID.
Username	The username that is associated with this process.
Regmods	The number of Windows registry modifications that were made by the execution of this process. Regmods are color-coded in blue.
Filemods	Contains a color-coded dot if the execution of the process resulted in file modifications. Filemods are color-coded in yellow.
Modloads	Contains a color-coded dot if the execution of the process resulted in loaded modules. Modloads are color-coded in green.
Netconns	Contains a color-coded dot if the execution of the process resulted in attempted or established network connections. Netconns are color-coded in purple.
Children	Contains a color-coded dot if the execution of the process resulted in generated child processes. Children are color-coded in orange.
Tags	Contains a color-coded dot if the execution of the process resulted in events that were tagged in a Carbon Black EDR investigation. Tags are color-coded in black.
Hits	Contains a color-coded dot if the execution of the process resulted in watchlist or feed hits. Hits are color-coded in red.
>	Opens the Process Analysis page.

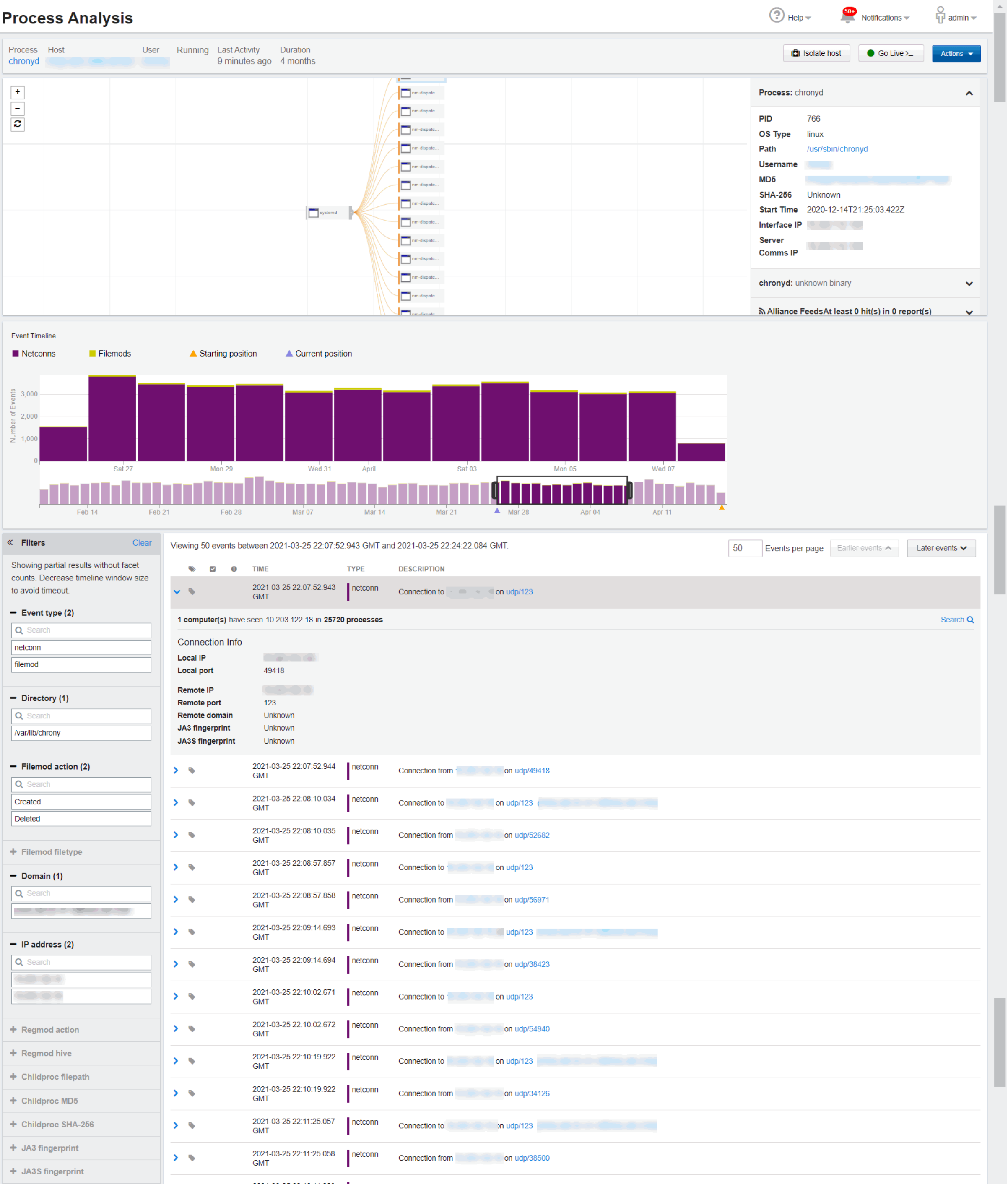
Process Analysis Page

When you find a process that merits investigation, you can open the Process Analysis page.

Procedure

1 Execute a query as discussed in Overview of Process Search.

2 In the Process Search Results table, locate the process to analyze and click the arrow (>).



Process Analysis

Process	Host	User	Running	Last Activity	Duration
chronyd	[redacted]	[redacted]		9 minutes ago	4 months

Process: chronyd

PID	766
OS Type	linux
Path	/usr/sbin/chronyd
Username	[redacted]
MD5	[redacted]
SHA-256	Unknown
Start Time	2020-12-14T21:25:03.422Z
Interface IP	[redacted]
Server Comms IP	[redacted]

chronyd: unknown binary

Alliance FeedsAt least 0 hit(s) in 0 report(s)

Filters Clear

Showing partial results without facet counts. Decrease timeline window size to avoid timeout.

− **Event type (2)**
Search
netconn
filemod

− **Directory (1)**
Search
/var/lib/chrony

− **Filemod action (2)**
Search
Created
Deleted

+ Filemod filetype

− **Domain (1)**
Search
[redacted]

− **IP address (2)**
Search
[redacted]
[redacted]

+ Regmod action
+ Regmod hive
+ Childproc filepath
+ Childproc MD5
+ Childproc SHA-256
+ JA3 fingerprint
+ JA3S fingerprint

Viewing 50 events between 2021-03-25 22:07:52.943 GMT and 2021-03-25 22:24:22.084 GMT.

50 Events per page Earlier events Later events

TIME	TYPE	DESCRIPTION
2021-03-25 22:07:52.943 GMT	netconn	Connection to [redacted] on udp/123

1 computer(s) have seen 10.203.122.18 in **25720 processes** Search

Connection Info

Local IP	[redacted]
Local port	49418
Remote IP	[redacted]
Remote port	123
Remote domain	Unknown
JA3 fingerprint	Unknown
JA3S fingerprint	Unknown

TIME	TYPE	DESCRIPTION
2021-03-25 22:07:52.944 GMT	netconn	Connection from [redacted] on udp/49418
2021-03-25 22:08:10.034 GMT	netconn	Connection to [redacted] on udp/123 [redacted]
2021-03-25 22:08:10.035 GMT	netconn	Connection from [redacted] on udp/52682
2021-03-25 22:08:57.857 GMT	netconn	Connection to [redacted] on udp/123
2021-03-25 22:08:57.858 GMT	netconn	Connection from [redacted] on udp/56971
2021-03-25 22:09:14.693 GMT	netconn	Connection to [redacted] udp/123 [redacted]
2021-03-25 22:09:14.694 GMT	netconn	Connection from [redacted] on udp/38423
2021-03-25 22:10:02.671 GMT	netconn	Connection to [redacted] on udp/123
2021-03-25 22:10:02.672 GMT	netconn	Connection from [redacted] on udp/54940
2021-03-25 22:10:19.922 GMT	netconn	Connection to [redacted] on udp/123 [redacted]
2021-03-25 22:10:19.922 GMT	netconn	Connection from [redacted] on udp/34126
2021-03-25 22:11:25.057 GMT	netconn	Connection to [redacted] on udp/123 [redacted]
2021-03-25 22:11:25.058 GMT	netconn	Connection from [redacted] on udp/38500

Process Summary

The process summary information is located in the top panel of the Process Analysis page.

The process summary displays the following general process execution details:

- **Process**: Identifies the main process for which the analysis is displayed.

- **Host**: Identifies the host upon which the command was initiated.
- **User**: Identifies the user who was logged in at the endpoint when the command was initiated.
- **Running**: Identifies the state of the process. The state can be Running or Terminated.
- **Last Activity**: The last time that the document was updated.
- **Duration**: The number of hours that the process has been running.
- **Isolate Host**: Click the **Isolate host** button to isolate an endpoint. The action presents an optional **Description** text box where you can note the reason for the activity.

  For example, you might discover that suspicious files are executing on a particular endpoint and you want to prevent them from spreading to other endpoints in your network.

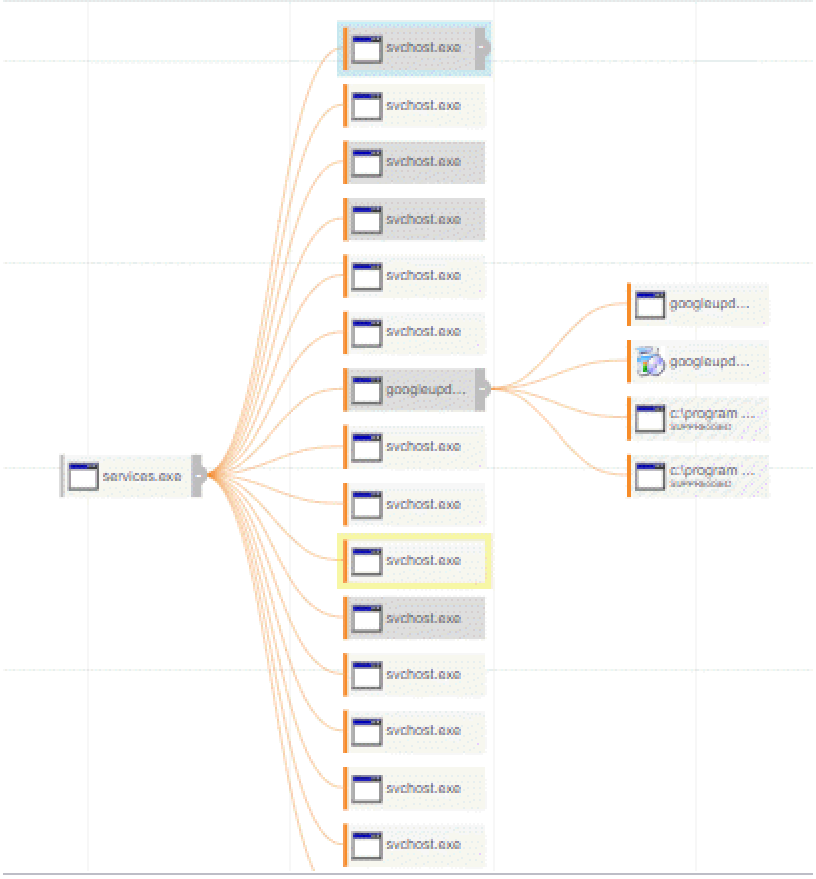
  When an endpoint is isolated, connections to the Carbon Black EDR server (such as DHCP and DNS) are maintained, but all other connections are blocked or terminated. The user is not notified by Carbon Black EDR, but the endpoint will not work as expected.

  Note To isolate an endpoint, you must be a Carbon Black EDR Global Administrator, a Carbon Black Hosted EDR Administrator, or a user on a team that has Analyst privileges for the endpoint to isolate.

  The computer remains isolated until this option is disabled or the computer reboots. See Isolating an Endpoint.
- **Go Live**: The **Go Live** button is useful when you are investigating an IOC. After you have identified an endpoint that has suspicious activity, you can directly access the content on that endpoint. You can open an interactive live session to the endpoint host and execute commands in real time to help isolate or eradicate the threat. See Using Live Response.
- **Actions**: The **Actions** dropdown menu includes the following options:
  - **Ban this hash** – Creates a ban of the process. If process hash banning is enabled for a sensor group, hosts attempting to run this process will find it blocked, and any running instances of the process are terminated. See Banning Process Hashes.
  - **Export events to CSV** – Downloads a `Report.zip` archive to your local computer. The files contain the information in the **Description** fields for each **Type** filter that appears in the Results table at the bottom of the **Process Analysis** window. See Process Event Filters.
  - **Share** – Opens the Carbon Black EDR user's default email client, creates an email, and includes the details from the `summary.txt` file (path, MD5, start timestamp, last updated timestamp, hostname, and full command line), and a URL that accesses the same page in which **Share** was clicked.

Interactive Process Tree

By default, the interactive **Process Tree** displays the parent process of the selected process executable file in a search result together with the relevant child processes.



You can interact with the **Process Tree** by clicking child and parent processes to identify issues. This view shows the selected process event and includes its parent process and child processes. Siblings to the selected process also appear.

To expand or collapse nodes in the **Process Tree**, click a parent or child node.

To view additional nodes, left-click and hold any part of the **Process Tree** while moving your cursor.

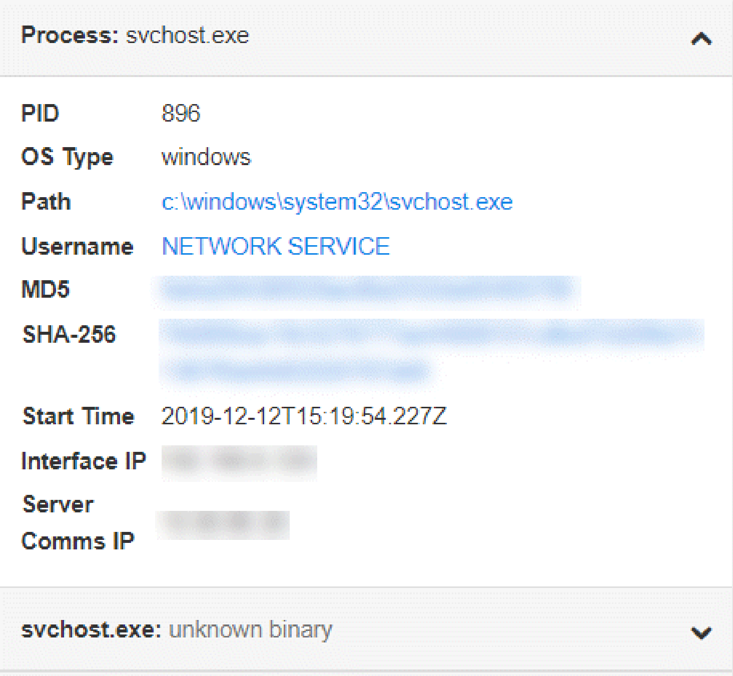
Clicking other child or parent processes updates the Process Analysis page in context to show the newly selected process details, including the summary tables and graphs.

Note The **Process Tree** can display up to 15 child processes: 15 unsuppressed, 15 suppressed, or 15 of both types.

For processes that have more than 15 unsuppressed and 15 suppressed child processes, the **Process Tree** shows unsuppressed processes first, and then suppressed processes, until a total of 15 child processes appear.

Process Execution Details

Process execution details appear in the panel to the right of the **Process Tree** on the Process Analysis page.



Process: svchost.exe

PID	896
OS Type	windows
Path	c:\windows\system32\svchost.exe
Username	NETWORK SERVICE
MD5	[redacted]
SHA-256	[redacted]
Start Time	2019-12-12T15:19:54.227Z
Interface IP	[redacted]
Server Comms IP	[redacted]

svchost.exe: unknown binary

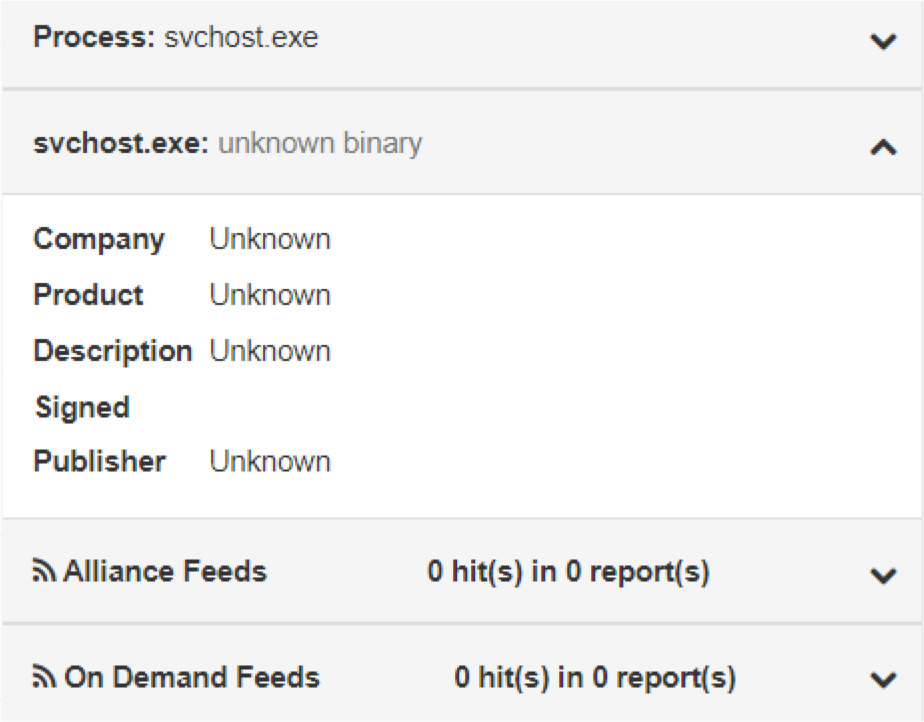
If the process is an executable, the following information is displayed:

Field	Description
Process	The name of the process executable file.
PID	The Process Identification (PID) number of the process.
OS Type	The operating system on which the process was executed.
Path	The physical path from which the process was executed.
Username	The name of the user who executed the process.
MD5	The MD5 hash value of the process.
SHA-256	The SHA-256 hash value of the process. **Note** Availability of SHA-256 hash data is dependent upon sensor capabilities. The macOS sensor version 6.2.4, which is packaged with Carbon Black EDR Server version 6.3, sends SHA-256 hashes to the server. Check Broadcom Carbon Black Support for information about other sensors that can generate SHA-256 hashes. For files that were originally discovered by a sensor that did not provide SHA-256 hashes, process information for new executions show SHA-256 hashes, but binary entries show SHA-256 as **(unknown)** until they appear as new files on a sensor that supports SHA-256.
Start Time	The date and time of the process execution.
Interface IP	The IP address of the network adapter on the sensor. Pre-5.1 sensors do not report an Interface IP.
Server Comms IP	The IP address from which the server recognizes the sensor that is reporting data. If the sensor is communicating through a Proxy or NAT, the address is for the Proxy or NAT.

Binary Metadata

This topic describes binary metadata for the process executable file.

To view digital signature information and metadata about the process executable file, click the down-arrow next to the process executable file name at the bottom of the **Process Execution Details** panel:



Process: svchost.exe

svchost.exe: unknown binary

Company	Unknown
Product	Unknown
Description	Unknown
Signed	
Publisher	Unknown

Alliance Feeds 0 hit(s) in 0 report(s)

On Demand Feeds 0 hit(s) in 0 report(s)

For both **Alliance Feeds** and **On Demand Feeds**, only items that occur within the timeline selection window are displayed. For long-running processes, if the timeline selection window is large, Carbon Black might only report a subset of the Alliance and On Demand Feed hits. In this case, a message displays this status. The same limitations apply to the facets that are displayed.

Alliance Feeds shows if the process event details had any hits from Carbon Black Threat Intel partner feeds.

If there are any hits, the results appear below the Carbon Black Threat Intel feeds in rows that are expanded by default. Each row shows:

- The source of the feed.
- A link to information about the threat that was detected.
- The date and score of the hit.
- The IOC (Indicator of Compromise) value of the process event that caused the hit.

Click the IOC hash value to go directly to the process event row for that event.

On Demand Feeds shows if the process event details had any hits from on-demand feeds.

On-demand feeds provide information from the Carbon Black Threat Intel “on demand” when a process that is part of the Carbon Black Threat Intel database is viewed on the Process Analysis page. This information includes domain classification and threat geolocation. There might not be any on-demand data available for a process that you are analyzing.

Click the **Sharing Settings** link to access the **Sharing** page to set this up. For more information, see On-Demand Feeds from Carbon Black Threat Intel.

EMET Protections Enabled (Windows Only)

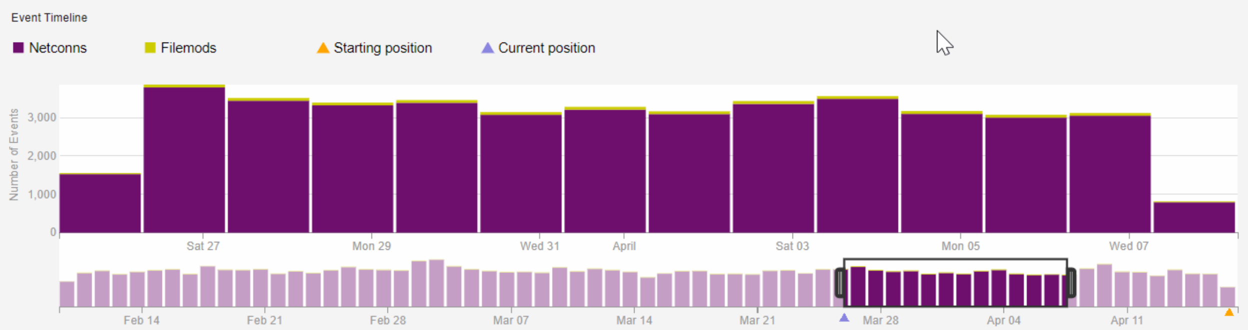
The **EMET Protections Enabled** panel appears if Enhanced Mitigation Experience Toolkit (EMET) is installed on the host that reported the process and EMET Protection is enabled for the process on that host.

Note EMET is at end of life (EOL) with Microsoft as of Windows 10.

Event Timeline

The **Event Timeline** is useful for investigating IOCs for events that occurred at a specific time.

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 displays event counts, which are broken down into event segments. The top graph expands, collapses, and slides back and forth in conjunction with the time range window that you select in the bottom graph.

The bottom graph contains an interactive time range selector window that you can expand or collapse to zoom in, on, and out of the timeline. Place your cursor on the black handle in the graph and slide the range selector back and forth across the timeline to contract or expand it. Or, place your cursor inside the area that is defined by the handles and move the range selector window. As you move the range selector window, the process event list is updated (see Event List). The bottom graph includes two indicators:

- An orange triangle, which represents the starting point of the segment selected from Process Search.
- A purple triangle, which represents the current point of the segment that you are viewing in the events list.

Process Event Search

This section describes how to search for process event results on the Process Analysis page.

A search bar is located above the Process Event Results table and below the Event Timeline to the right of the **Filters** section. You can use the search bar to search for one or more process events that match your criteria. Search and Filters criteria are applied simultaneously.

The process events in the Process Event Results table are populated based on the selected Event Timeline time range. Thus, searches execute against all process events that occurred within the selected time range.

Supported Search Syntax and Content

This topic describes supported search syntax and content for process event searches on the Process Analysis page.

Process Analysis event search is a value-based search feature. Thus, enter the value you are searching for. In the context of a key:value pair, this means you should specify the value, without specifying the key. Search is not case-sensitive.

To clear your search criteria, delete the content in the search bar and press **`Enter`**.

The following sections provide an overview of the searchable values per event type.

Blocked

- Time value (without GMT)
- Values provided in the `Description` field from the API (not the pre-populated text that is presented in the console)
- Values provided in the following fields in the Process Metadata section:
  - `Activity`
  - `Username`
  - `MD5`
  - `SHA-256`
  - `Command line`

  Note Values contained in the **Binary Info** section are not searchable.

Child Process (childproc)

- Time value (without GMT)
- Values provided in the `Description` field from the API (not the pre-populated text that is presented in the console)
- Values provided in the following fields in the Process Metadata section:
  - `Activity`

  - `Username`
  - `MD5`
  - `SHA-256`
  - `Command line`
  - `Suppressed`

  Note Values contained in the **Binary Info** section are not searchable.

Cross Process (crossproc)

- Time value (without GMT)
- Values provided in the `Description` field from the API (not the pre-populated text that is presented in the console)
- Values provided in the following fields in the Process Metadata section:
  - `Activity`
  - `Username`
  - `MD5`
  - `SHA-256`
  - `Command line`

  Note Values contained in the **Binary Info** section are not searchable.

File Modification (filemod)

- Time value (without GMT)
- Values provided in the `Description` field from the API (not the pre-populated text that is presented in the console)

Fileless Scriptload

- Time value (without GMT)
- Values provided in the `Description` field from the API (not the pre-populated text that is presented in the console)
- Values provided in the following fields in the Fileless Script Load Metadata section:
  - `SHA-256`
  - `Command length`
  - `Command line`

Fork

- Time value (without GMT)

- Values provided in the `Description` field from the API (not the pre-populated text that is presented in the console)

Module Load (modload)

- Time value (without GMT)
- Values provided in the `Description` field from the API (not the pre-populated text that is presented in the console)

  Note Values contained in the **Binary Info** section are not searchable.

Network Connection (netconn)

- Time value (without GMT)
- Values provided in the `Description` field from the API (not the pre-populated text that is presented in the console)
- Values provided in the following fields in the Connection Info section:
  - `Local IP`
  - `Local port`
  - `Remote IP`
  - `Remote port`
  - `Remote domain`
  - `JA3 fingerprint`

Posix_exec Process on macOS and Linux (posix exec)

- Time value (without GMT)
- Values provided in the `Description` field from the API (not the pre-populated text that is presented in the console)
- Values provided in the following fields in the Process Metadata section:
  - `Activity`
  - `Username`
  - `MD5`
  - `SHA-256`
  - `Command line`

  Note Values contained in the **Binary Info** and **Alliance Info** sections are not searchable.

Unsupported Search Syntax and Content

This topic describes unsupported search syntax and content for process event searches on the Process Analysis page.

Unlike search on the Process Search page, the use of field names is not supported in search on the Process Analysis page. On the Process Search page, `key:value` pair format is supported, but it is not supported on the Process Analysis page. Simply specify the value without specifying the respective key (field name).

The use of logical Boolean operators, like `AND`, `OR`, `NOT`, `XOR`, is not supported in search on the Process Analysis page.

Search for the Event type, like `filemod`, `regmod`, and so on, is not supported. To specify the Event types you are looking for, use the **Event type Filters** section.

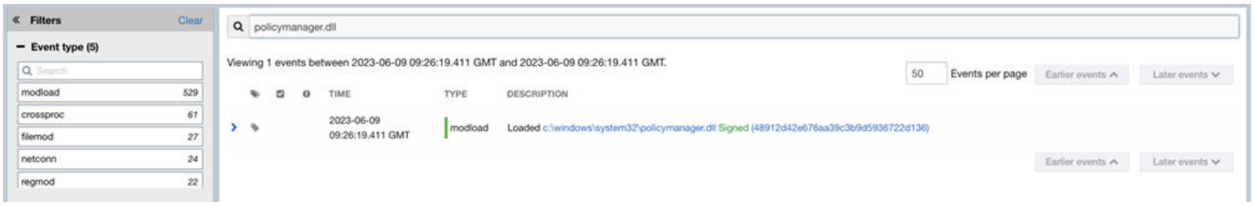
Content contained in the **Binary Metadata** and **Alliance Info** sections is not searchable.

Search Examples

This topic provides search examples for process event searches on the Process Analysis page.

Example 1: Search Criteria Applied

In this example, the user wants to find module load (`modload`) events where `c:\windows\system32\policymanager.dll` was loaded. Even though `modload` is not selected in the **Event type Filters** facet, the user can find only modload events where `c:\windows\system32\policymanager.dll` was loaded by searching for **`policymanager.dll`**.



policymanager.dll

Viewing 1 events between 2023-06-09 09:26:19.411 GMT and 2023-06-09 09:26:19.411 GMT.

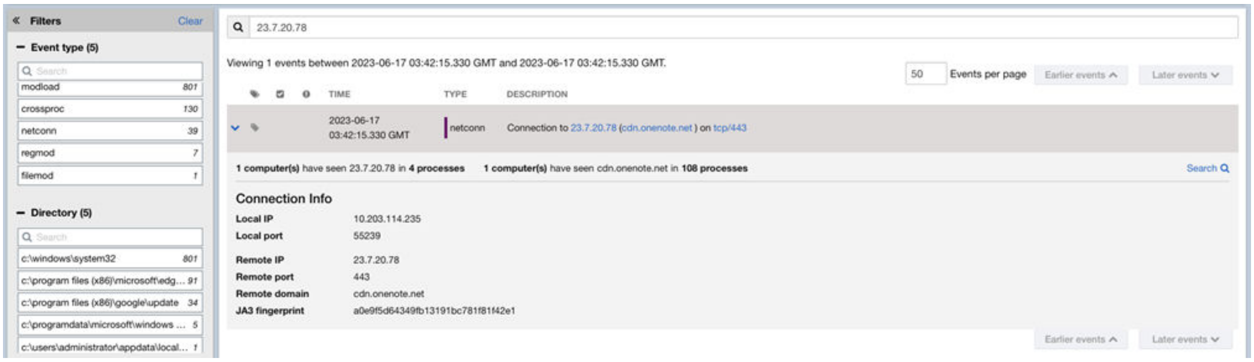
50

Event type (5)	
modload	529
crossproc	61
filemod	27
netconn	24
regmod	22

TIME	TYPE	DESCRIPTION
2023-06-09 09:26:19.411 GMT	modload	Loaded c:\windows\system32\policymanager.dll Signed (48912d42e676aa39c3b9d5936722d136)

Example 2: Search Criteria Applied

In this example, the user wants to find network connection (`netconn`) events, where the `Remote IP` value equals `23.7.20.78`.



23.7.20.78

Viewing 1 events between 2023-06-17 03:42:15.330 GMT and 2023-06-17 03:42:15.330 GMT.

50

Event type (5)	
modload	801
crossproc	130
netconn	39
regmod	7
filemod	1

Directory (5)	
c:\windows\system32	801
c:\program files (x86)\microsoft\edg...	91
c:\program files (x86)\google\update	34
c:\programdata\microsoft\windows ...	5
c:\users\administrator\appdata\local...	1

TIME	TYPE	DESCRIPTION
2023-06-17 03:42:15.330 GMT	netconn	Connection to 23.7.20.78 (cdn.onenote.net) on tcp/443

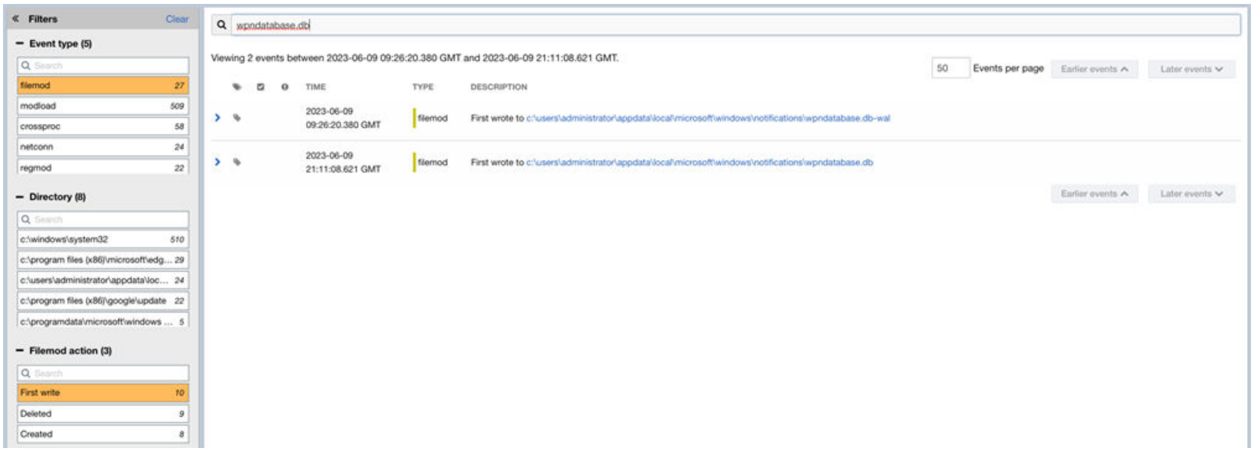
1 computer(s) have seen 23.7.20.78 in **4 processes** **1 computer(s)** have seen cdn.onenote.net in **108 processes**

Connection Info

Local IP	10.203.114.235
Local port	55239
Remote IP	23.7.20.78
Remote port	443
Remote domain	cdn.onenote.net
JA3 fingerprint	a0e9f5d64349fb13191bc781f81f42e1

Example 3: Search and Filters Criteria Applied

In this example, the user wants to find file modification (`filemod`) events, where the `Filemod` action equals `First write` and the process first wrote to a file path that contains `wpndatabase.db`. The user can use Filters to filter by `Event type = filemod` and `Filemod action = First write`, and then the user can search for `wpndatabase.db` to further refine the search results to only those where the process first wrote to a file path that contains `wpndatabase.db`. As shown in the following image, two results are returned: one where the file name is `wpndatabase.db`, and one where the file name is `wpndatabase.db-wal` because both values match the search content.



wpndatabase.db

Viewing 2 events between 2023-06-09 09:26:20.380 GMT and 2023-06-09 21:11:08.621 GMT.

50 Events per page

TIME	TYPE	DESCRIPTION
2023-06-09 09:26:20.380 GMT	filemod	First wrote to c:\users\administrator\appdata\local\microsoft\windows\notifications\wpndatabase.db-wal
2023-06-09 21:11:08.621 GMT	filemod	First wrote to c:\users\administrator\appdata\local\microsoft\windows\notifications\wpndatabase.db

Event type (5)	
filemod	27
modload	509
crossproc	58
netconn	24
regmod	22

Directory (8)	
c:\windows\system32	510
c:\program files (x86)\microsoft\edg...	29
c:\users\administrator\appdata\loc...	24
c:\program files (x86)\google\update	22
c:\programdata\microsoft\windows ...	5

Filemod action (3)	
First write	10
Deleted	9
Created	8

Search Timeout

This topic describes default search timeout behavior, configuration of the search timeout duration, and the error message that appears when a search timeout occurs.

Note For more information about the `/etc/cb/cb.conf` file, see *Carbon Black EDR Server Configuration Guide*.

A search timeout error displays if the search does not complete across all process events that occurred in the selected time range. The search timeout duration is set to 30 seconds by default, but this value is configurable using the `ProcessAnalysisEventSearchTimeout` setting in the `/etc/cb/cb.conf` file.

If the search times out, the following error message displays:

Search timed out. Please try narrowing search criteria.

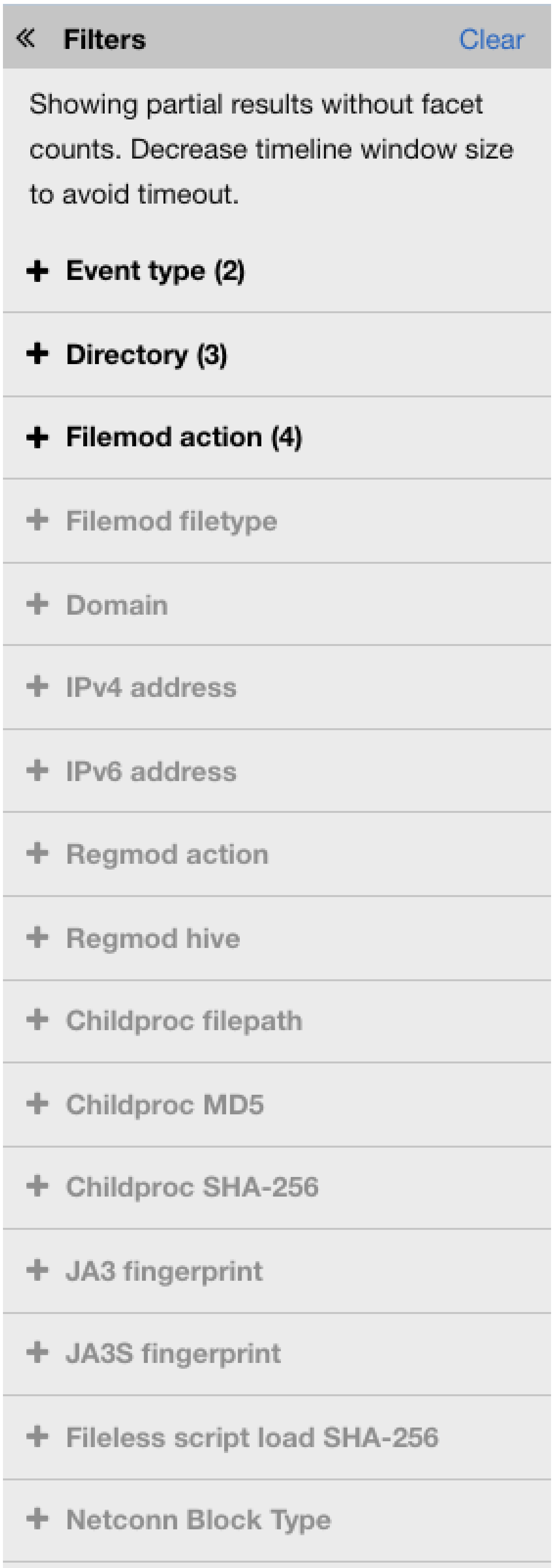
Even if a timeout occurs, process events that matched the search criteria within the configured timeout duration can be returned; that is, a partial set of results can be returned.

To avoid a timeout, narrow the selected time range or narrow the search and filter criteria.

Process Event Filters

This topic describes process event filters on the Process Analysis page.

Filters display in a scrollable list to the left of the **Events List**. To hide or show the filters list, click the expand button to the left of the list.



Showing partial results without facet counts. Decrease timeline window size to avoid timeout.

Process event filters provide a further refinement of the displayed data on the Process Analysis page.

Filter rows can show the number of events that match that value. Click the **Reset** button to reset all filters to their original state.

The following table describes each filter (in alphabetical order).

Filter	Description
Childproc filepath	Paths to child processes that were created by this process.
Childproc md5	MD5 files of child processes that were created by this process.
Childproc sha-256	SHA-256 of child processes that were created by this process.
Directory	The directories that this process uses.
Domain	The domain (DNS) names that are associated with network connections that were made by this process.
Event type	Shows process event types. See Process Event Types for more details. ■ **filemod** – file modifications ■ **modload** – number of modules loaded ■ **regmod** – (Windows only) registry modifications ■ **netconn** – number of network connections enabled ■ **childproc** – child processes ■ **fork** – (macOS and Linux only) fork processes ■ **posix_exec** – (macOS and Linux only) posix_exec processes ■ **crossproc** – (Windows only - not supported on Windows XP/2003) cross processes ■ **blocked** – process blocked due to ban ■ **emet** – (Windows only) EMET mitigation
Fileless script load SHA-256	The fileless_scriptload event represents each occasion when the sensor detected PowerShell script content that was executed by any process on a supported endpoint.
FileMod action	The types of file modifications that occurred during the execution of this process (create, delete, first write, last write), and the number of times those actions occurred.
FileMod file type	The types of files that were modified.
IP address	The IP addresses that are associated with network connections that were made by this process.
JA3	JA3 fingerprint of the client TLS hello packet.
JA3S	JA3S fingerprint of the server TLS hello packet.
Netconn Block Type	The classification of the network connection attempt. This is a sub-field of a netconn event. ■ Blocked due to isolation – The network connection attempt was blocked due to the endpoint being in Isolation (Type 1). ■ Not blocked – The network connection attempt was successful (Type 0).
RegMod action	(Windows only) The type of registry modification (created, deleted key, deleted value, first write, last write).
RegMod hive	The location of the registry that is associated with registry modification events.

Process Event Types

The following table shows different types of details that display for each type of event.

Event Type	Details
blocked	**The path and hash of a process that is blocked by a Carbon Black EDR process hash ban. When expanded, metadata for the process and its binary appear:** ■ **Process metadata – when the process was terminated, username of the user attempting to run the process, process MD5, command line path for the process.** ■ **Binary metadata – SHA-256 hash (if available), company name, product name, product description, signature status, publisher.**
childproc	■ The number of endpoints that have observed the MD5 in the description and the number of processes in which the MD5 was observed. Lists the names of the processes. ■ Process metadata – The length of time for which the process was active, and when the process execution occurred, username of the user who is executing the process, MD5 hash, SHA-256 hash (if available), and the command line of the process executable file. ■ Binary information – SHA-256 hash (if available), company name, product name, product description, signature status, and publisher. ■ If the child process is suppressed due to Retention Maximization, then it also shows the username and command line. You choose maximization levels in the Edit Group Settings and Create Group pages. See Advanced Settings. This image shows suppressed vs. unsuppressed child processes. Suppressed child processes are labeled **Suppressed** in the **process tree**. You can discover whether a childproc in the **Event List** is suppressed by expanding its details. ■ The **process tree** shows a maximum of 15 child processes: either 15 unsuppressed, 15 suppressed, or 15 of both types. ■ For processes that have more than 15 unsuppressed and 15 suppressed child processes, the tree shows unsuppressed processes first, and then suppressed processes, until a total of 15 child processes appear in the tree.
crossproc	Windows only (not supported on Windows XP/2003): Shows occurrences of processes that cross the security boundary of other processes: ■ Description of the OpenProcess API call for the cross process. Carbon Black EDR records all OpenProcess API calls that request `PROCESS_CREATE_PROCESS`, `PROCESS_CREATE_THREAD` , `PROCESS_DUP_HANDLE` , `PROCESS_SUSPEND_RESUME` , `PROCESS_VM_OPERATION` , or `PROCESS_VM_WRITE` access rights. These access rights allow this process to change the behavior of the target process. ■ Process metadata – the length of time the cross process was active, username of the user who executed the process, MD5 hash, SHA-256 hash (if available), and the command line of the process executable file. ■ Binary metadata – SHA-256 hash (if available), the company name, product name, product description, signature status, and publisher.
emet	(Windows only) The EMET mitigation type reported when this process was invoked and the filename used in the attempt to run the process. Additional details include number of endpoints and processes that have seen the event, the time of the EMET mitigation, the EMET ID of the event, and any warnings. Output from EMET might provide additional details.

Event Type	Details
blocked	**The path and hash of a process that is blocked by a Carbon Black EDR process hash ban. When expanded, metadata for the process and its binary appear:** ■ **Process metadata – when the process was terminated, username of the user attempting to run the process, process MD5, command line path for the process.** ■ **Binary metadata – SHA-256 hash (if available), company name, product name, product description, signature status, publisher.**
fileless scriptload	(Windows only) The fileless_scriptload event represents each occasion when the sensor detected AMSI-decoded script content that was executed by any process on the endpoint. This content consists only of fileless script content that was not stored in a file on the file system when that content was executed. When expanded, Fileless Script Load Metadata appears: `Fileless Script Load Metadata - SHA-256 hash (if available), Command length, and Command line.` **Note** The `Command line` field can be expanded to view the full command if it is truncated.
filemod	The number of endpoints that have seen this file modification and the number of processes in which the file modification occurred on those endpoints.
fork	(macOS and Linux only) Indicates that this is a fork process and shows the instance's parent process, forked with a different Process ID (PID). When a process performs a fork() system call, all activity for that process continues to be associated with the parent. A new **fork** event type is displayed on the Process Analysis 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 occurred is recorded
modload	■ The number of endpoints that have seen the MD5 hash for the module that was loaded and the number of processes in which the MD5 appears on those endpoints. ■ Binary information – SHA-256 hash (if available), company name, product name, a description of the binary, signature status, and publisher ■ Carbon Black Threat Intel information – the source of the threat intelligence feed, a link to the report for the MD5 hash, the MD5 score, and the MD5 trust status.
netconn	The number of network connections that the execution of this process either attempted or established.
posix_exec	(macOS and Linux only) Indicates this is a posix_exec process and shows the instance's process that is loaded and the new binary image. If a process performs an exec() system call, a new process document will not be created. This activity will be reported as a new **posix_exec** event type within the process, and the process metadata will be updated to reflect the new image and command line associated with the exec() system call.
regmod	Windows sensors only. The number of endpoints that have seen a modification of a registry key, and the number of processes in which the registry modification occurred on those endpoints.

Event List

The **Event List** describes every recorded event that occurred inside the selected time range from the timeline.

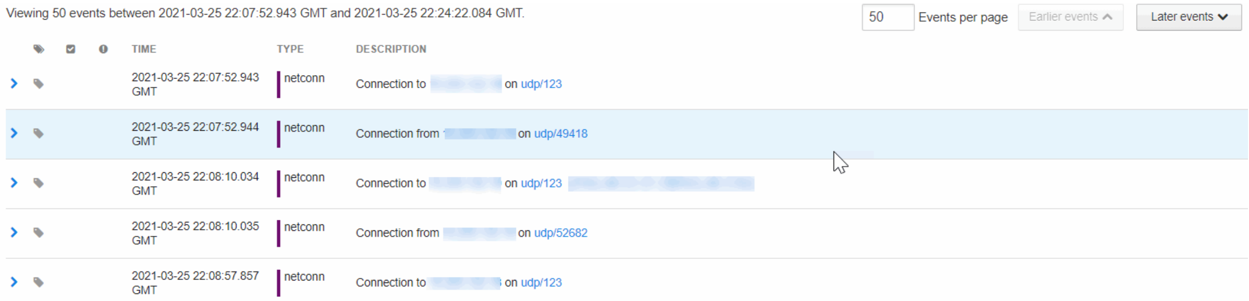
When you first enter the Process Analysis page by choosing a Process Search result row, the selected row corresponds with a particular segment of the process. This selection affects the starting position of the **Event List**.

For example, a process shown in the Process Search result row might have three segments: 8:00, 9:00, and 10:00. If you choose the middle segment, the first row of the **Event List** displays the first event of the 9:00 segment. The timeline displays an orange triangle icon indicating that starting point.

If you expand, shrink, or move the timeline window, the window snaps to the nearest time segment boundary, and the **Event List** automatically scrolls to the first event of that time segment. The timeline displays a purple triangle icon indicating where that event is inside the process.

When you expand the timeline window, you increase the total number of events to display in the **Event List**. A 10-second timeout is in effect; this timeout only applies to the filters and process metadata. The events list will always show the true amount of events. If all filters and metadata for this timeline are not retrieved within 10 seconds, a message displays that the data is incomplete; to rectify this, reduce the size of the timeline window.

When you select a node in the Process Tree, the chosen segment becomes the first segment of the displayed process in the **Event List**.



Viewing 50 events between 2021-03-25 22:07:52.943 GMT and 2021-03-25 22:24:22.084 GMT.

50 Events per page

				TIME	TYPE	DESCRIPTION
›				2021-03-25 22:07:52.943 GMT	netconn	Connection to [redacted] on udp/123
›				2021-03-25 22:07:52.944 GMT	netconn	Connection from [redacted] on udp/49418
›				2021-03-25 22:08:10.034 GMT	netconn	Connection to [redacted] on udp/123 [redacted]
›				2021-03-25 22:08:10.035 GMT	netconn	Connection from [redacted] on udp/52682
›				2021-03-25 22:08:57.857 GMT	netconn	Connection to [redacted] on udp/123

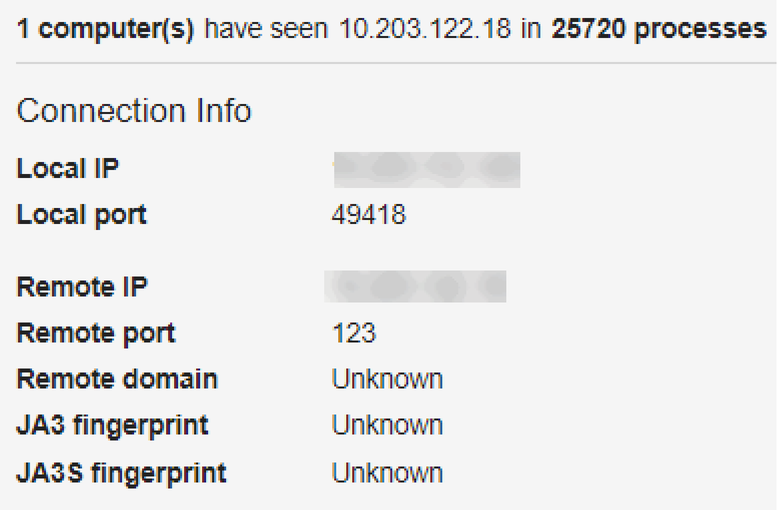
You can set the number of events to view per page, and define whether to view earlier or later events.

The **Event List** shows the following details:

Heading	Description
Expand event >	Allows you to expand the event for additional data.
Tag 🏷	Shows if an event is tagged for an investigation. You can click the tag icon to select this event for a future investigation. After you select the tag icon, it turns blue to indicate that it is now included in an investigation.
Trusted Events ☑	Shows if the event is trusted. When you click on the row, the trust information appears with a link to the source.
Threat Intelligence Feed Hits ❗	Shows if this event has matched a threat intelligence feed.
Time	The time that the event occurred in Greenwich Mean Time (GMT).

Heading	Description
Type	The process event type. For more details, see Process Event Types. ■ **crossproc** (cross process) – appears with a red bar (Windows only - not supported on Windows XP/2003). ■ **child process** (child process) – appears with an orange bar. ■ **fork** (fork process) – appears with a yellow-orange bar (macOS and Linux only). ■ **filemod** (file modification) – appears with a yellow bar. ■ **modload** (number of modules loaded – appears with a green bar. ■ **posix_exec** (posix_exec process) – appears with a blue green bar (macOS and Linux only). ■ **regmod** (registry modification) – (Windows only) appears with a blue bar. ■ **netconn** (number of network connections enabled) – appears with a purple bar. ■ **blocked** (process blocked by hash ban) – appears with a brown bar. ■ **emet** (EMET mitigation) – appears with a gray bar (Windows only).
Description	The operation that the **Type** event performed. See Process Event Types. The **Description** column can contain: ■ **filemod** – “Deleted” or “Created” and then provide the path to the file that was modified. ■ **modload** – The module that was loaded by the process. Modload descriptions can also include the path of the module that was loaded, if the module was signed or unsigned by the publisher, and the unique MD5 hash. ■ **regmod** – The Windows registry key that was created or modified. ■ **netconn** – The connection made, including the IP address (including hostname unless DNS resolution is excluded for the host), port, and protocol. ■ **childproc** – The child process start time, end time, and PID of the selected parent process. ■ **fork** – (macOS and Linux only) The instance’s parent process, forked with a different Process ID (PID). ■ **posix_exec** – (macOS and Linux only) The instance’s loaded process and the new binary image. ■ **crossproc** – (Windows only - not supported on Windows XP/2003) The action it performed; for example, opening a handle or starting or ending processes. ■ **blocked** – Blocked events. These are associated with the banning functionality. ■ **emet** – (Windows only) The EMET mitigation type reported when this process was invoked and the filename that was used in the attempt to run the process.
Search	Lets you reduce the number of events that display and focus the results based on terms entered into the **Search** box. For example, entering “Microsoft” into the **Search** box would display only Microsoft events.

When you expand an event by clicking the **>** on the left side of the row, details about the event appear. This example shows details for an event of the type netconn:



1 computer(s) have seen 10.203.122.18 in **25720 processes**

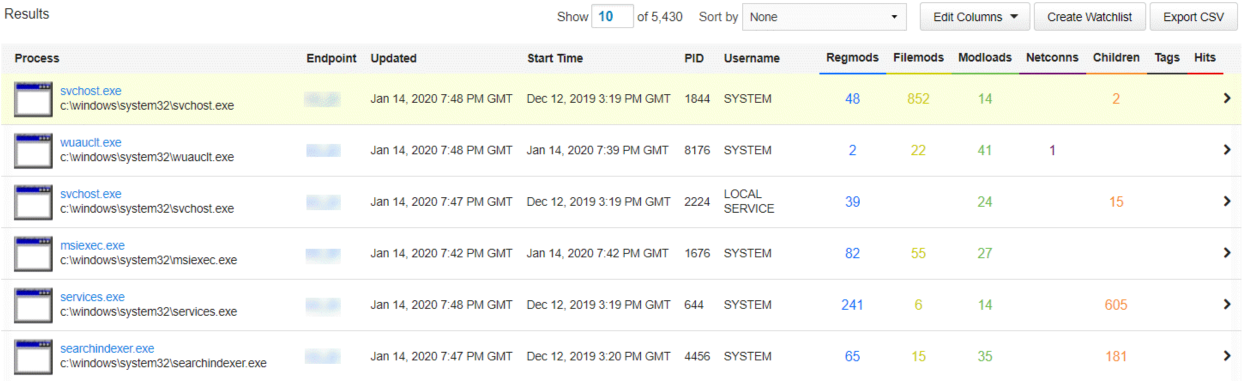
Connection Info

Local IP	[redacted]
Local port	49418
Remote IP	[redacted]
Remote port	123
Remote domain	Unknown
JA3 fingerprint	Unknown
JA3S fingerprint	Unknown

Process Analysis Preview Window

This topic describes the Process Analysis Preview window.

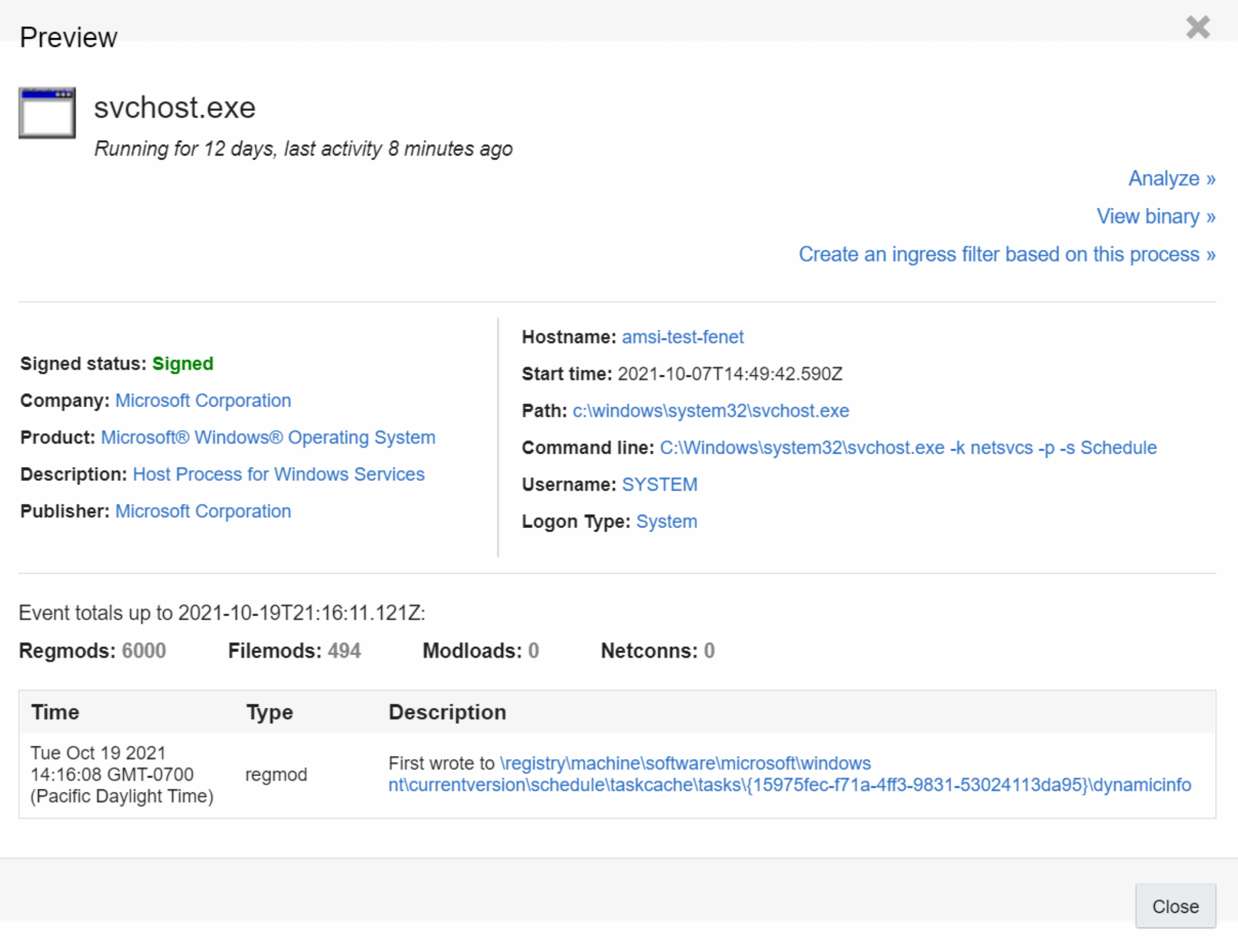
On the Process Search page (see Overview of Process Search), scroll to the **Results** table (see Process Search Results Table). Click anywhere in a query result row (except for a hyperlinked item or the **>** icon).



Results

Process	Endpoint	Updated	Start Time	PID	Username	Regmods	Filemods	Modloads	Netconns	Children	Tags	Hits	
svchost.exe c:\windows\system32\svchost.exe	[redacted]	Jan 14, 2020 7:48 PM GMT	Dec 12, 2019 3:19 PM GMT	1844	SYSTEM	48	852	14		2			>
wuauclt.exe c:\windows\system32\wuauclt.exe	[redacted]	Jan 14, 2020 7:48 PM GMT	Jan 14, 2020 7:39 PM GMT	8176	SYSTEM	2	22	41	1				>
svchost.exe c:\windows\system32\svchost.exe	[redacted]	Jan 14, 2020 7:47 PM GMT	Dec 12, 2019 3:19 PM GMT	2224	LOCAL SERVICE	39		24		15			>
msiexec.exe c:\windows\system32\msiexec.exe	[redacted]	Jan 14, 2020 7:42 PM GMT	Jan 14, 2020 7:42 PM GMT	1676	SYSTEM	82	55	27					>
services.exe c:\windows\system32\services.exe	[redacted]	Jan 14, 2020 7:48 PM GMT	Dec 12, 2019 3:19 PM GMT	644	SYSTEM	241	6	14		605			>
searchindexer.exe c:\windows\system32\searchindexer.exe	[redacted]	Jan 14, 2020 7:47 PM GMT	Dec 12, 2019 3:20 PM GMT	4456	SYSTEM	65	15	35		181			>

The **Process Analysis Preview** window appears and provides a brief overview of the process that you selected, without leaving the page:



Preview

svchost.exe

Running for 12 days, last activity 8 minutes ago

Signed status: Signed

Company: Microsoft Corporation

Product: Microsoft® Windows® Operating System

Description: Host Process for Windows Services

Publisher: Microsoft Corporation

Hostname: amsi-test-fenet

Start time: 2021-10-07T14:49:42.590Z

Path: c:\windows\system32\svchost.exe

Command line: C:\Windows\system32\svchost.exe -k netsvcs -p -s Schedule

Username: SYSTEM

Logon Type: System

Event totals up to 2021-10-19T21:16:11.121Z:

Regmods: 6000 **Filemods:** 494 **Modloads:** 0 **Netconns:** 0

Time	Type	Description
Tue Oct 19 2021 14:16:08 GMT-0700 (Pacific Daylight Time)	regmod	First wrote to \registry\machine\software\microsoft\windows nt\currentversion\schedule\taskcache\tasks\{15975fec-f71a-4ff3-9831-53024113da95}\dynamicinfo

Title	Description
Analyze	Click to open the Process Analysis page for a granular analysis of the process executable file. See Process Analysis Page.
View Binary	Click to view the detailed binary analysis page for the process executable file. See Chapter 11 Binary Search and Analysis.
Create an ingress filter based on this process	Click to open the **Add Ingress Filter** window to create an ingress filter based on the selected process. See Adding an Ingress Filter.
Signed status	Shows if the process executable file is signed by the publisher.
Company	The company name of the process executable file.
Product	The product for which the process executable file was created.
Description	A text description of the process executable file.
Publisher	The official publisher of the process executable file.
Hostname	The name of the host (endpoint) on which the process was run.
Start time	The full timestamp for the time when the process was run.
Path	The physical path from which the process was run.

Title	Description
Command line	The full command line specific to the execution of this process.
Username	The user on the given host who executed the process. The format is `<domain>\<username>` .
Logon Type	The method of logon associated with the process.
Regmods	The number of Windows registry modifications that were made by the process execution.
Filemods	The number of files that were modified by the execution of this process.
Modloads	The status of modules that were loaded by this process execution.
Netconns	The number of network connections that this process execution either attempted or established.

Binary Search and Analysis

11

This section explains how to search for and analyze binary metadata.

Read the following topics next:

Binary Search Criteria

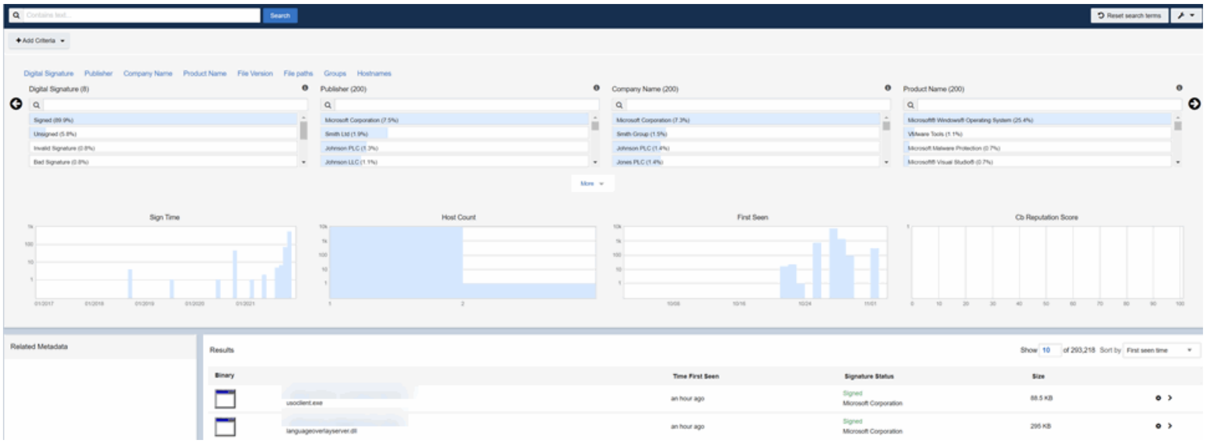
Carbon Black EDR sensors begin tracking binaries when they are executed by a process. You can perform a binary search to explore the metadata of a binary.

You can enter keyword searches or pre-defined search criteria in the **Search** box at the top of the page. While you enter search criteria, the correct syntax is displayed. However, the search not only auto-completes your criteria but estimates results as well.

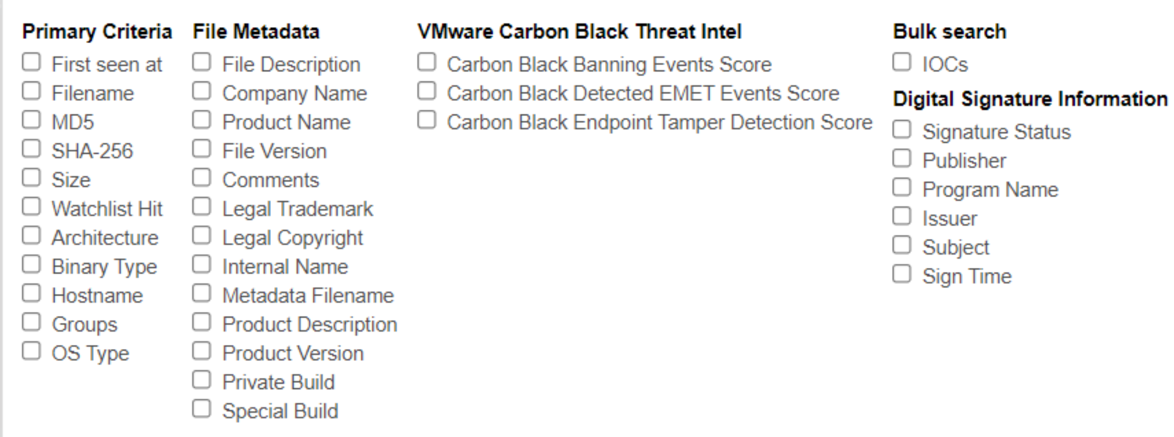
If you do not enter any search criteria, the system runs a search with *.*, which includes every binary that has executed in your environment. The results appear with a single instance of each binary and its metadata. Each binary is identified by its MD5 hash value.

Procedure

1 On the navigation bar, click **Binary Search**.



2 In the **Search** box, enter a search string (formatted with the correct syntax) or click **Add Criteria** to display predefined search criteria options:



Primary Criteria

□ First seen at
□ Filename
□ MD5
□ SHA-256
□ Size
□ Watchlist Hit
□ Architecture
□ Binary Type
□ Hostname
□ Groups
□ OS Type

File Metadata

□ File Description
□ Company Name
□ Product Name
□ File Version
□ Comments
□ Legal Trademark
□ Legal Copyright
□ Internal Name
□ Metadata Filename
□ Product Description
□ Product Version
□ Private Build
□ Special Build

VMware Carbon Black Threat Intel

□ Carbon Black Banning Events Score
□ Carbon Black Detected EMET Events Score
□ Carbon Black Endpoint Tamper Detection Score

Bulk search

□ IOCs

Digital Signature Information

□ Signature Status
□ Publisher
□ Program Name
□ Issuer
□ Subject
□ Sign Time

If you select a search criteria option, you must specify details for that search criteria option. For example, if you select the **OS Type** search criteria option, you must select one or more OS types for this search and then click **Update**.

If you add multiple search criteria fields, they are combined using an `AND` operator.

3 When you finish entering search criteria, click **Search**.

Search results appear in a series of facets and graphs together with a Binary Search Results table.

Note For detailed information about using queries, see Chapter 12 Advanced Search Queries.

Additional Binary Search Page Features

This topic describes Binary Search page features in addition to the **Search** component.

In the top-right corner of the page, an **Actions** menu provides several options:

- **Share** – Share query strings. You can email the URL of the Carbon Black EDR server with a query string to another Carbon Black EDR user. That user can then use the string to view the same results in their own Carbon Black EDR console.
- **Add Watchlist** – Create a watchlist that is based on the current query string. A watchlist is a saved search that you can use to track specific IOCs. See Chapter 19 Watchlists.
- **Export CSV** – Export the first 1000 process search results into a CSV file in a comma-separated value format for reporting, retention, or compliance. Each row contains a URL to access the result details.

Note To export more than 1000 rows of data, you must configure API functionality to capture and save the data. See the Carbon Black Developers Network at https://developer.carbonblack.com/reference/enterprise-response/ .

The **Reset search terms** button at the top right of the Binary Search page removes all search criteria and restores the default view using *.* as the search criteria.

Below the facets and to the left of the binary search results, the **Related Metadata** panel appears. If you hover over an item in **Related Metadata**, rows that correspond with the selected common elements are highlighted to the right.

High-level Binary Search Result Summaries

When you click **Search** , the Binary Search page updates the results data with information that is specific to your search criteria. The results are displayed in a variety of formats that allow you to quickly find suspicious binaries.

A summary of the results appears in facets (small tables and graphs that provide high-level result data). Each process that matches your search criteria appears in a row below the facets.

Facets provide a high-level summary of your current search results. Click the information icons to learn more about each facet.

The top row of facets contains information about the binary search results. Click the right-arrow to see all facets in this row.

- **Digital Signature** – The percentages of signed, unsigned, explicit distrust, and expired binaries.
- **Publisher** – A list of binary publishers and the percentage of binaries that have those publishers.
- **Company Name** – A list of binary publisher companies and the percentage of binaries with those company names.
- **Product Name** – The product name of the binary.
- **File Version** – The file version of the binary.

- **File Paths** – A list of file paths where files matching the current binary search have been seen.
- **Groups** – A list of the sensor groups that have identified binaries.
- **Hostnames** – A list of host names for computers on which binaries have been identified.

The second facet row contains graphs. Clicking on a facet within a graph filters the results to show the items that match that value. By default, these facets are sorted by the highest-to-lowest percentage.

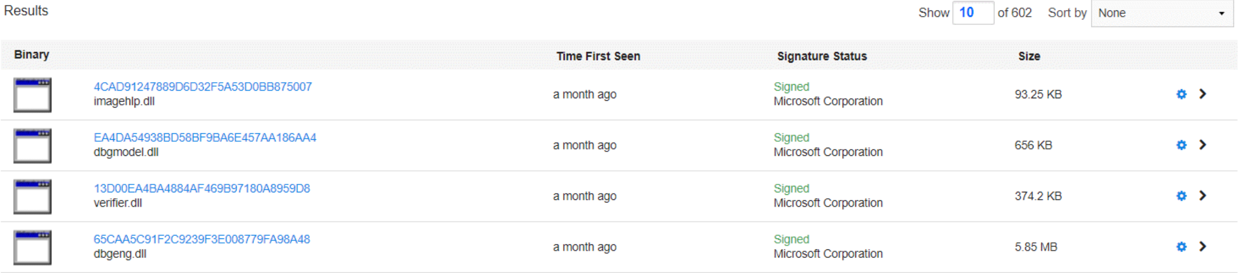
Hovering over a facet within a graph displays binary counts.

The second facet row displays the following information about binaries in the results:

- **Sign Time** – The number of binaries that were signed on a particular date.
- **Host Count** – The number of binaries that were seen by Carbon Black EDR on a host or a number of hosts.
- **First Seen** – The number of binaries that were first detected on a particular date.
- **Carbon Black Reputation Score** – The number of binaries that match the current search listed by Carbon Black Reputation Score.

Binary Search Results Table

At the bottom of the page (to the right of **Related Metadata**), the Binary Search Results table appears. Each row provides details about binary metadata that matches the search criteria.



Results

Show 10 of 602 Sort by None

Binary	Time First Seen	Signature Status	Size
4CAD91247889D6D32F5A53D0BB875007 imagehlp.dll	a month ago	Signed Microsoft Corporation	93.25 KB
EA4DA54938BD58BF9BA6E457AA186AA4 dbgmodel.dll	a month ago	Signed Microsoft Corporation	656 KB
13D00EA4BA4884AF469B97180A8959D8 verifier.dll	a month ago	Signed Microsoft Corporation	374.2 KB
65CAA5C91F2C9239F3E008779FA98A48 dbgeng.dll	a month ago	Signed Microsoft Corporation	5.85 MB

Above the search results, you can see how many binaries match the search criteria and selected filters. You can select sorting options for the list of binaries.

Search results provide the following information about the binaries in the list:

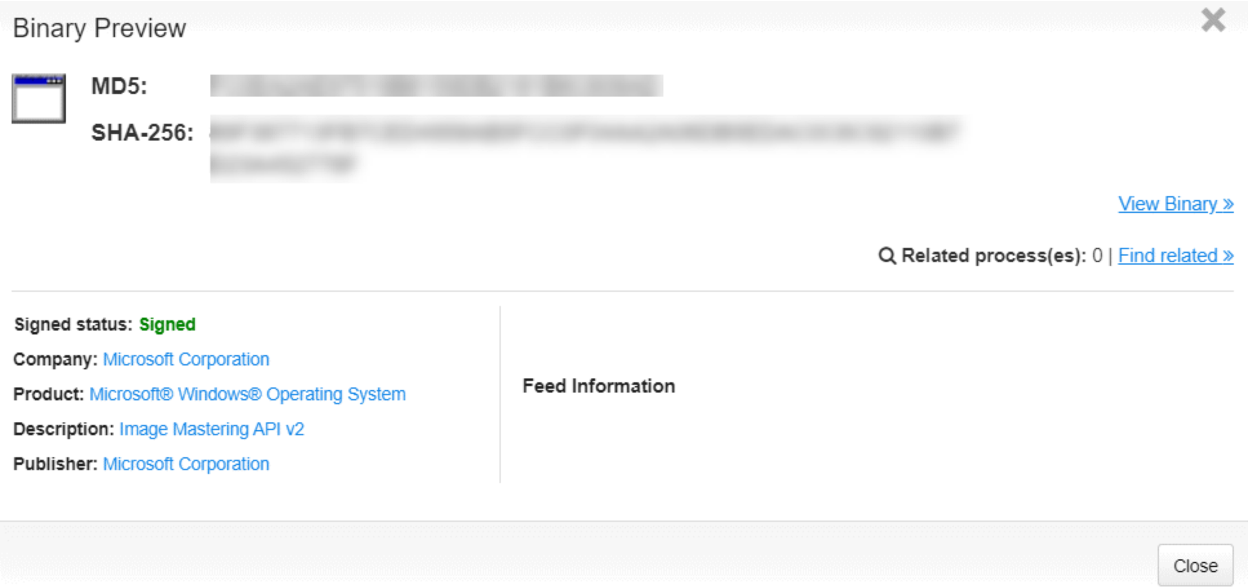
Title	Description
Icon	The icon of the file in which the binary was detected. For example: Click to display the Binary Preview page. See Binary Preview.
Binary MD5 Hash	The MD5 hash value of the binary.

Title	Description
Time First Seen	The first time that the binary was seen.
Signature	Shows whether the binary file is signed or unsigned.
Size	The size of the file that contains the binary.
⚙	Indicates whether an existing watchlist identified the binary. Click the icon to open the watchlist. See Chapter 19 Watchlists
>	Click to display the Binary Analysis page. See Binary Analysis.

Binary Preview

At the top of the Binary Preview page, the hashes of the binary (MD5 and, if available, SHA-256) appear. The file name(s) that the binary has used are listed beneath the hash value (if available).

- Click the icon at the left of a row on the Binary Search Results table to view the Binary Preview page.



Binary Preview

MD5: [redacted]

SHA-256: [redacted]

Related process(es): 0 | Find related »

Signed status: Signed

Company: Microsoft Corporation

Product: Microsoft® Windows® Operating System

Description: Image Mastering API v2

Publisher: Microsoft Corporation

Feed Information

The Binary Preview page provides a quick overview of the following details:

- **Metadata**
- **Signed status** – The status of whether the binary file is signed by the publisher.
- **Company** – The company name identified in the metadata of the binary file.
- **Product** – The product name identified in the metadata of the binary file.
- **Description** – A text description of the binary file.
- **Publisher** – The official publisher of the binary file.

- **Feed Information** – A list of Carbon Black Threat Intel feed scan results. You can click on the blue links to go to the source of the results.

At the top right of the page, the following options appear:

- **View Binary** – Click to view the detailed Binary Analysis page. See Binary Analysis.
- **Find related** – Click to open the Process Search page with a predefined query for the MD5 hash value of this binary. The number of related processes displays to the left of the **Find related** link. See Chapter 10 Process Search and Analysis

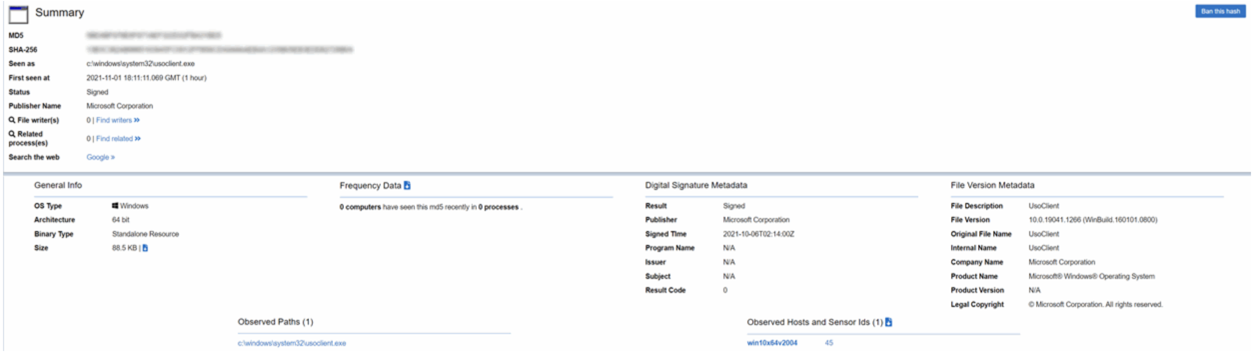
Binary Analysis

Use the Binary Analysis page to thoroughly investigate a binary.

You can access the page in one of two ways:

- Click the **View Binary** link on the Binary Preview page.
- Click the **>** icon on the right end of a binary search results table row from the Binary Search page. See

The Binary Analysis page displays:



Summary

MD5	[redacted]
SHA-256	[redacted]
Seen as	c:\windows\system32\usoclient.exe
First seen at	2021-11-01 18:11:11.069 GMT (1 hour)
Status	Signed
Publisher Name	Microsoft Corporation
File writer(s)	0 | Find writers »
Related process(es)	0 | Find related »
Search the web	Google »

General Info

OS Type	Windows
Architecture	64 bit
Binary Type	Standalone Resource
Size	88.5 KB |

Frequency Data

0 computers have seen this md5 recently in **0 processes** .

Digital Signature Metadata

Result	Signed
Publisher	Microsoft Corporation
Signed Time	2021-10-06T02:14:00Z
Program Name	N/A
Issuer	N/A
Subject	N/A
Result Code	0

File Version Metadata

File Description	UsoClient
File Version	10.0.19041.1266 (WinBuild.160101.0800)
Original File Name	UsoClient
Internal Name	UsoClient
Company Name	Microsoft Corporation
Product Name	Microsoft® Windows® Operating System
Product Version	N/A
Legal Copyright	© Microsoft Corporation. All rights reserved.

Observed Paths (1)

c:\windows\system32\usoclient.exe

Observed Hosts and Sensor Ids (1)

win10x64v2004	45

Binary Overview

The **Binary Overview** section of the Binary Analysis page includes the following information.

Heading	Description
MD5 Hash Value	MD5 hash value for the binary.
SHA-256 Hash Value	The SHA-256 hash value for the binary. **Note** **Note:** Availability of SHA-256 hash data is dependent upon sensor capabilities. The macOS sensor version 6.2.4, which is packaged with Carbon Black EDR server version 6.3, sends SHA-256 hashes to the server. Check Broadcom Carbon Black Support for information about other sensors that can generate SHA-256 hashes. For files that were originally discovered by a sensor that did not provide SHA-256 hashes, process information for new executions show SHA-256 hashes, but binary entries show SHA-256 as “(unknown)” until they appear as new files on a sensor that supports SHA-256.

Heading	Description
Seen as	Filenames that were seen for binaries that match this MD5 hash value.
First seen at	Full time stamp of the time that this binary was last observed by currently installed sensors.
Status	Signature status — either **Signed** or **Unsigned** .
Publisher Name	Name of the binary publisher.
File writer(s)	Number and names of files the binary has written to. Click the **Find Writers** link to view the files on the Process Search page.
Related Process(es)	Number of processes that have used this binary. Click the Find related link to find related process on the Process Search page.
Search the web	Performs a Google search for the MD5 hash value of the binary.
Feed Information	Shows scan results for this binary from Carbon Black Threat Intel feeds. Click the links to see the results.
Ban this hash	Click this button to ban this hash. Banning a hash terminates a process, if running, and prevents it from running in the future. See Banning Process Hashes.

Binary Analysis General Info

The **General Info** section of the Binary Analysis page shows the following details about the binary file.

Heading	Description
OS Type	Binary operating system.
Architecture	Binary architecture — 32-bit or 64-bit.
Binary Type	Binary resource type — Standalone or Shared .
Size	Size of the binary file. Also provides a link to download the physical binary.

Download	Click the **Download** button to download a copy of this binary in a zip file with a name derived from the MD5 hash of the file (for example, `A96E734A0E63B7F9B95317125DDEA2BC.zip`).
	The zip file contains two files: metadata and filedata.
	The metadata file is a text file that contains a timestamp and original filename.
	For example:
	`Timestamp: 01/17/2022 09:50:56`
	`OrigFilename: : \Device\HarddiskVolume1\ProgramData\Microsoft\Windows Defender\Definition Updates\{0B1E4D3A-9612-462F-8067-B0EDCE49CBF2}\mpengine.dll`

Binary Analysis Frequency Data

Frequency Data shows how many hosts have observed the binary that has this MD5 hash value.

Click the down-arrow to download the list of hosts into a CSV file.

Binary Analysis Digital Signature Metadata

This topic describes the **Digital Signature Metadata** section of the Binary Analysis page.

Digital signature metadata about the binary is as follows:

Heading	Description
Result	The status of the binary signature — either Signed or Unsigned.
Publisher	The publisher of the binary.
Signed Time	The time that the binary was signed.
Program Name	The binary program name.
Issuer	The binary issuer.
Subject	The binary subject.
Result Code	The result or exit code that followed the execution of the binary.

Binary File Version Metadata

This topic describes the **File Version Metadata** section of the Process Analysis page.

File version metadata about the binary is as follows:

Heading	Description
File Description	Binary name (from the publisher).
File Version	Binary version
Original Filename	Binary filename.
Internal Name	Internal name of the binary
Company Name	Company name of the binary.
Product Name	Product name of the binary.
Product Version	Product version of the binary file.
Legal Copyright	Copyright details for the file, including its publisher.

Observed Paths, Hosts, and Sensor IDs

This topic describes the **Observed Paths**, **Observed Hosts**, and **Observed Sensor IDs** sections of the Process Analysis page.

- Observed paths are the full physical paths from which the binary was loaded.

- Observed Hosts and Sensor IDs shows the names of hosts on which this binary was observed together with the ID number of the sensor. Click the down-arrow to download the list of hosts into a CSV file.

Advanced Search Queries

12

The Carbon Black EDR console lets you choose criteria for searches of processes, binaries, alerts, and threat reports. This section describes how to construct complex queries.

The fields, field types, and examples in this section focus on queries to search for processes and binaries, but most of the syntax descriptions also apply to alerts and threat reports.

Read the following topics next:

- Query Syntax Details
- Fields in Process and Binary Searches
- Fields in Alert and Threat Report Searches
- Field Types
- Searching with Multiple (Bulk) Criteria
- Searching with Binary Joins
- Example Searches

Query Syntax Details

Carbon Black EDR supports multiple types of operators and syntax that can form complex queries in the Search boxes on the Process Search, Binary Search, Threat Report Search, and Triage Alerts pages

Searches are generally case-insensitive.

Terms, Phrases, and Operators

This topic describes terms, phrases, and operators you can use when constructing a complex query.

A term is a single keyword (without whitespace) that is searched in the Carbon Black EDR process or binary data store, or in the alerts or threat reports on your server. For example, a keyword could be: `svchost.exe.`

Terms can be combined by logical operators and nested to form complex queries; for example:

- and, AND, or whitespace — Boolean `AND` operator: `svchost.exe cmd.exe, svchost.exe and cmd.exe`
- or, OR — Boolean `OR` operator: `svchost.exe or cmd.exe`
- - — Boolean `NOT` operator: `-svchost.exe`
- nesting using parenthesis: `(svchost.exe or cmd.exe) powershell.exe”`
- Wildcard searches with `*` ; for example, `process_name:win*.exe`

Terms can be limited to a single field with *<field>:<term>* syntax; for example:

`process_name:svchost.exe`

Multiple terms are connected with `AND` if not otherwise specified.

Terms that are not preceded by fields are expanded to search *all* default fields.

Because terms are whitespace-delimited, use double quotes, or escape whitespaces with a single backslash, when required.

For example:

`path:"microsoft office\office15\powerpnt.exe"`

or

`path:microsoft\ office\office15\powerpnt.exe`

Terms can be combined to form phrases. A phrase is a set of terms that are separated by whitespace and enclosed in quotes. Whitespace between the terms of a quoted phrase is not treated as a logical `AND` operator. Instead, a phrase is searched as a single term.

For example: `“svchost.exe cmd.exe”`

Phrases can be combined and nested with other phrases and terms using logical operators.

For example: `"svchost.exe cmd.exe" or powershell.exe`

Restrictions on Terms

This topic describes restrictions on terms you can encounter when constructing a complex query.

Whitespace

Whitespace is the default delimiter. A query with whitespace is “tokenized” and parsed as multiple terms.

For example:

This input: `microsoft office\office15\powerpnt.exe`

is interpreted as two terms: `microsoft AND office\office15\powerpnt.exe`

Use quotation marks to avoid automatic parsing into individual terms.

For example:

This input: "`microsoft office\office15\powerpnt.exe`"

Is interpreted as: `microsoft office\office15\powerpnt.exe`

Alternatively, you can escape whitespaces by using the backslash (\).

For example:

This input: `microsoft\ office\office15\powerpnt.exe`

Is interpreted as: `microsoft office\office15\powerpnt.exe`

See Field Type: path for more information about how whitespaces and slashes affect path tokenization.

Parentheses

Parentheses are used as a delimiter for nested queries. A query with parentheses is parsed as a nested query, and if a proper nesting cannot be found, a syntax error is returned.

For example:

This input: `c:\program files (x86)\windows`

is interpreted as: `c:\program AND files AND x86 AND \windows`

Use quotation marks around the whole phrase to avoid automatic nesting. Otherwise, escape the parentheses (and whitespaces) using the backslash (\).

For example:

This input: `c:\program\ files\ \(x86\)\windows`

is interpreted as: `c:\program files (x86)\windows`

Negative Sign

The negative sign is used as logical `NOT` operator. Queries that begin with a negative sign are negated in the submitted query.

For example:

This input: `-system.exe`

is interpreted as: `not system.exe`

This input: `-alliance_score_srstrust:*`

is interpreted as: `Return all results that are not trusted by the alliance.`

You can use a phrase query to avoid automatic negation.

Double Quotes

Double quotes are used as a delimiter for phrase queries. A query in which double quotes should be taken literally must be escaped using backslash (\).

For example, the following query input:

```
cmdline:"\"c:\program files \(x86\)\google\update\googleupdate.exe\" /svc"
```

is interpreted to match the following command line (with the command line including the quotes as shown):

```
"c:\program files (x86)\google\update\googleupdate.exe\" /svc
```

Leading Wildcards

The use of leading wildcards in a query is not recommended unless absolutely necessary, and is blocked by default. Leading wildcards carry a significant performance penalty for the search.

For example, the following query is not recommended:

```
filemod:*/system32/ntdll.dll
```

The same results would be returned by the following query, and the search would be much more efficient:

```
filemod:system32/ntdll.dll
```

Note While process searches with leading wildcards are blocked by default beginning in Carbon Black EDR version 6.2.3, you can change this either through the Advanced Settings page or the `cb.conf` file. See Managing High-Impact Queries and the *Carbon Black EDR Server Configuration Guide*.

Fields in Process and Binary Searches

This topic contains a complete list of fields that are searchable in Carbon Black EDR Process and Binary searches.

Some fields are valid in only one of the two, and some in both. Any binary-related field that the process search uses actually searches the executable file backing the process.

If a query specifies a term without specifying a field, the search is executed on all default fields. Default fields are indicated by `(def)` .

Note Availability of SHA-256 hash data is dependent upon sensor capabilities. The macOS sensor version 6.2.4, which is packaged with Carbon Black EDR Server version 6.3, sends SHA-256 hashes to the server. Check Broadcom Carbon Black Support for information about other sensors that can generate SHA-256 hashes.

For files that were originally discovered by a sensor that did not provide SHA-256 hashes, process information for new executions show SHA-256 hashes, but binary entries show SHA-256 as “(unknown)” until they appear as new files on a sensor that supports SHA-256. This applies to all SHA-256 related fields.

Field	Process Search	Binary Search	Field Type	Description
blocked_md5	x (def)	-	md5	MD5 of a process blocked due to a banning rule.
blocked_status	x	-	status	Status of a block attempt on a running process due to a banning rule, one of the following: a-ProcessTerminated b-NotTerminatedCBProcess c-NotTerminatedSystemProcess d-NotTerminatedCriticialSystemProcess e-NotTerminatedWhiltestedPath f-NotTerminatedOpenProcessError g-NotTerminatedTerminateError
childproc_count	x	-	count	Total count of child processes created by this process.
childproc_md5	x (def)	-	md5	MD5 of the executable backing the created child processes.
childproc_sha256	x (def)	-	sha256	SHA-256 of the executable backing the created child processes (if available).
childproc_name	x (def)	-	keyword	Filename of the child process executables.
cmdline	x (def)	-	cmdline	Full command line for this process.
comments	-	x (def)	text	Comment string from the class FileVersionInfo.
company_name	x	x (def)	text	Company name string from the class FileVersionInfo.
copied_mod_len	x	x	count	Number of bytes collected.
crossproc_count	x		count	Total count of cross process actions by an actor process.
crossproc_md5	x		md5	MD5 of an actor process that performed a cross process action on a target process.
crossproc_sha256	x		sha256	SHA-256 of an actor process that performed a cross process action on a target process (if available).
crossproc_name	x		keyword	Name of an actor process that performed a cross process action on a target process.

Field	Process Search	Binary Search	Field Type	Description
crossproc_type	x (def)		keyword	■ **processopen** (or process_open) finds processes which opened a handle into another process with a set of access rights. Sample results: OpenThread() API call requested THREAD_GET_CONTEXT, THREAD_SET_CONTEXT, THREAD_SUSPEND_RESUME access rights. ■ **remotethread** (or remote_thread) finds processes which injected a thread into another process. Sample results: CreateRemoteThread API used to inject code into target process. ■ **processopentarget** is similar to processopen, but instead of finding the actor, the process returns the targeted process; i.e., the process which the handle is opened into. ■ **remotethreadtarget** is similar to remotethread, but instead of finding the actor process, it returns the targeted process; i.e., the process which the thread was injected into.
digsig_issuer	x	x (def)	text	If digitally signed, the issuer.
digsig_prog_name	x	x (def)	text	If digitally signed, the program name.
digsig_publisher	x	x (def)	text	If digitally signed, the publisher.
digsig_result	x	x (def)	sign	If digitally signed, the result. Values are: ■ “Bad Signature” ■ “Invalid Signature” ■ “Expired” ■ “Invalid Chain” ■ “Untrusted Root” ■ “Signed” ■ “Unsigned” ■ “Explicit Distrust”
digsig_sign_time	x	x	datetime	If digitally signed, the time of signing.
digsig_subject	x	x (def)	text	If digitally signed, the subject.
domain	x (def)	-	domain	Network connection to this domain.
file_desc	x	x (def)	text	File description string from the class FileVersionInfo.
file_version	x	x (def)	text	File version string from the class FileVersionInfo.

Field	Process Search	Binary Search	Field Type	Description
fileless_scriptload_cmdline	x	-	text	Command line contents of a fileless scriptload event.
fileless_scriptload_cmdline_length	x	-	integer	Length of the command line contents of a fileless scriptload event.
filemod	x (def)	-	path	Path of a file modified by this process.
filemod_count	x	-	count	Total count of file modifications by this process.
filewrite_md5	x (def)	-	md5	MD5 of file written by this process.
filewrite_sha256	x (def)	-	md5	SHA-256 of file written by this process (if available).
group	x (def)	x (def)	keyword	Sensor group this sensor was assigned to at the time of process execution.
has_emet_config	x	-	bool	True or False - Indicates whether process has EMET mitigations configured/enabled.
has_emet_event	x	-	bool	True or False - Indicates whether process has EMET mitigation events.
host_count	-	x	count	Count of hosts that have seen a binary.
host_type	x (def)	-	keyword	Type of the computer: workstation, server, or domain controller.
hostname	x (def)	x (def)	keyword	Hostname of the computer on which the process was executed.
internal_name	x	x (def)	text	Internal name string from the class FileVersionInfo.
ipaddr	x	-	ipaddr	Network connection to or from this IP address. Only a remote (destination) IP address is searchable regardless of incoming or outgoing. IPv4-mapped addresses (::FFFF:1.2.3.4) are stored as IPv4 netconns, and can be queried using either ipaddr:1.2.3.4 or ipv4mapped:1.2.3.4. IPv4-mapped addresses can also be queried using the ipv6addr:::FFFF:1.2.3.4 . Such queries are automatically translated to ipv4mapped:1.2.3.4.

Field	Process Search	Binary Search	Field Type	Description
ipv6addr	x	-	ipv6addr	Network connection to or from this IPv6 address. Only a remote (destination) IP address is searchable regardless of incoming or outgoing. IPv4-compatible IPv6 addresses (::1.2.3.4) are stored as IPv6 netconns and can be queried using either ipv6addr:::1.2.3.4 or ipv6addr::0102:0304 (the latter is the native form; the dotted quad form is automatically translated to the native form).
ipport	x	-	integer	Network connection to this destination port.
is_64bit	x	x	bool	True if architecture is x64.
is_executable_image	x	x	bool	True if the binary is an EXE (versus DLL or SYS).
ja3	x	-	md5	JA3 fingerprint of the client TLS hello packet. You can search for the hash value. The term searched for must exactly match the value in the field.
ja3s	x	-	md5	JA3S fingerprint of the server TLS hello packet. You can search for the hash value. The term searched for must exactly match the value in the field.
last_server_update	x	-	datetime	Last activity in this process in the server's local time.
last_update	x	-	datetime	Last activity in this process in the computer's local time.
legal_copyright	x	x (def)	text	Legal copyright string from the class FileVersionInfo.
legal_trademark	x	x (def)	text	Legal trademark string from the class FileVersionInfo.
md5	x (def)	x (def)	md5	MD5 of the process, parent, child process, loaded module, or a written file.
modload	x (def)	-	path	Path of module loaded into this process.
modload_count	x	-	count	Total count of module loads by this process.
netconn_block_type	x	-	integer	The classification of the network connection attempt. This is a sub-field of a netconn event: 0 equals a successful network connection; 1 equals a network connection attempt that was blocked due to the endpoint being in Isolation.
netconn_count	x	-	count	Total count of network connections by this process.

Field	Process Search	Binary Search	Field Type	Description
observed_filename	x	x (def)	path	Full path of the binary at the time of collection.
orig_mod_len	x	x	count	Size in bytes of the binary at time of collection.
original_filename	x	x (def)	text	Original name string from the class FileVersionInfo.
os_type	x	x	keyword	Type of the operating system: Windows, macOS, or Linux.
parent_id	x	-	long	The internal Carbon Black EDR process guid for the parent process.
parent_md5	x (def)	-	md5	MD5 of the executable backing the parent process.
parent_sha256	x (def)	-	sha256	SHA-256 of the executable backing the parent process (if available).
parent_name	x (def)	-	keyword	Filename of the parent process executable.
path	x (def)	-	path	Full path to the executable backing this process.
private_build	x	x (def)	text	Private build string from the class FileVersionInfo.
process_id	x	-	long	The internal Carbon Black EDR process guid for the process.
process_md5	x (def)	-	md5	MD5 of the executable backing this process.
process_sha256	x (def)	-	sha256	SHA-256 of the executable backing this process (if available).
process_name	x (def)	-	keyword	Filename of the executable backing this process.
product_desc	x	x (def)	text	Product description string from the class FileVersionInfo.
product_name	x	x (def)	text	Product name string from the class FileVersionInfo.
product_version	x	x (def)	text	Product version string from the class FileVersionInfo.
regmod	x (def)	-	path	Path of a registry key modified by this process.
regmod_count	x	-	count	Total count of registry modifications by this process.
sensor_id	x	-	long	The internal Carbon Black EDR sensor guid of the computer on which this process was executed.
server_added_ timestamp	-	x	datetime	Time this binary was first seen by the server.

Field	Process Search	Binary Search	Field Type	Description
sha256	x (def)	x (def)	sha256	SHA-256 of the process, parent, child process, loaded module, or a written file (if available).
special_build	x	x (def)	text	Special build string from the class FileVersionInfo.
start	x	-	datetime	Start time of this process in the computer's local time.
tampered	x	x	bool	True if attempts were made to modify the sensor's binaries, disk artifacts, or configuration
username	x (def)	-	keyword	User context with which the process was executed.
watchlist_<id>	x	x	datetime	Time that this process or binary matched the watchlist query with <id>.

Fields in Alert and Threat Report Searches

The following sets of fields are searchable on the Triage Alerts and Threat Report Search pages.

As with process and binary searches, if no field is specified for a term, the search is executed on all default fields. In the following table, default fields are indicated by (def).

Field	Field Type	Description
alert_severity	float	Overall score of the alert (combines report score, feed rating, sensor criticality). For more information, see Chapter 14 Threat Intelligence Feeds.
alert_type	keyword	Type of the alert: one of "watchlist.hit.ingress.binary", "wathclist.hit.ingress.process", "watchlist.hit.query.process", "watchlist.hit.query.binary", "watchlist.hit.ingress.host"
assigned_to	keyword (def)	Name of the Carbon Black EDR administrator who changed the alert status.
create_time	datetime	Date and time this feed report was created.
created_time	datetime	Creation time of the alert.
description	text (def)	Description of the feed report, whitespace tokenized so each term is individually searchable.
domain	domain (def)	A domain IOC value in the feed report.
feed_category	text (def)	Category of this report/feed, whitespace tokenized.
feed_id	int	Numeric value of the feed id (-1 for watchlists).
feed_name	keyword (def)	Name of the feed that triggered the alert. All user-created watchlists have the feed name "My Watchlists" as a special case.
group	keyword	Sensor group name of the endpoint on which the process/binary that triggered the alert was observed.

Field	Field Type	Description
hostname	keyword (def)	Hostname of endpoint that the process/binary that triggered the alert was observed on.
ioc_value	keyword (def)	Value (IP address, MD5, or SHA-256) of the IOC that caused the alert to be triggered.
ipaddr	ipaddr	An IP address IOC value in the feed report.
ipv6addr	ipv6addr	An IPv6 address IOC value in the feed report.
is_ignored	bool	Indicates whether the report has been marked to be ignored on this server.
md5	md5 (def)	MD5 of the process that triggered the alert, or an MD5 IOC value in the feed report.
observed_filename	keyword (def)	Full path name of the process triggered the alert (not tokenized).
process_name	keyword (def)	Filename of the process that triggered the alert.
process_path	path (def)	Full path to the executable backing the process.
report_id	keyword	Name or unique identifier of the threat report that is part of the field.
report_score	float	Report score of the feed that triggered the alert. For more information, see Chapter 14 Threat Intelligence Feeds.
resolved_time	datetime	Time this alert was triaged by a resolution action.
sha256	sha256 (def)	SHA-256 of the process that triggered the alert (if available), or a SHA-256 IOC value in the feed report.
status	keyword	Status of the alert: one of "resolved", "unresolved", "in progress", "false positive".
tags	text (def)	Tags related to this report/feed, whitespace tokenized.
title	text	Text title of the feed report, whitespace tokenized.
update_time	datetime	Date and time this feed report was last updated.
username	keyword (def)	Username in whose context the process that triggered the alert event was executed.
watchlist_id	int (def)	Numeric value of the watchlist id (not applicable to feeds).
watchlist_name	keyword (def)	Name of the watchlist or the report (for feeds).

Field Types

This section describes the field types for advanced queries.

Field Type: bool

Boolean fields have only two possible values: the string `true` or `false` . Searches are case-insensitive.

Field Type: cmdline

When a process launches on an endpoint, the command line for that process is sent to the Carbon Black EDR server.

If the server stored the whole command line as one item and allowed open ended queries of it, query performance would be extremely poor to the point of making search unusable. Instead, the server breaks each command line up into smaller component “tokens” to be stored for use when you enter a command line query.

Tokenization requires that decisions be made about which components of a command become their own token and which components are treated as delimiters between tokens. These decisions involve trade-offs since the same character may be used in different ways in a command. This topic describes how tokenization is done for Carbon Black Hosted EDR instances and Carbon Black EDR 6.3.0 servers (and later). If you are upgrading, see also Tokenization Changes on Server Upgrade.

Tokenization Rules

With enhanced tokenization, the following characters are converted to white spaces and removed before the command-line is tokenized.

Characters Removed Before Tokenization

```
\ “ ‘ ( ) [ ] { } , = < > & | ;
```

Several frequently used characters are intentionally **not removed** before tokenization. These include:

- Percent (`%`) and dollar (`$`), often used for variables
- Dash (`-`), period (`.`), and underscore (`_`), often found as parts of file names
- These additional characters: `^ @ # ! ?`

Parsing Forward Slashes

The forward slash (`/`) character is handled differently depending upon its position. If it is the start of the entire command line, it is assumed to be part of the path. If it is at the start of any other token in the command line, it is assumed to be a command line switch.

There is one situation in which this parsing rule may not produce the results you want. It is not efficient for the command line parser to distinguish between a command line switch and a Unix-style absolute path. Therefore, Linux and macOS absolute paths passed on the command line are tokenized as if the beginning of the path were a command line switch. So a command line of `/bin/ls /tmp/somefile` will produce the tokens `bin` , `ls` , `/tmp` and `somefile` , incorrectly considering `/tmp` a command line switch.

Parsing Colons

The colon (:) character is handled differently depending upon its position and whether it is repeated. If it is the end of a token, it is assumed to be something the user would want to search for like a drive letter, so it is included. If there are multiple colons at the end of a token or if the colons are not at the end of a token, they are converted to white space for tokenization purposes.

File Extension Tokens

File extension tokens allow searching for either just the file extension or the entire command or file name. In other words, “word.exe” in a command line becomes two tokens: “.exe” and “word.exe”.

Wildcards

There is support for the "?" and "*" characters as wildcards when used as a non-leading character in a query, allowing you to search for any single character or multiple variable characters within a token, respectively.

Note Do not use wildcards as leading characters in a search.

Tokenization Changes on Server Upgrade

This section is relevant to users upgrading from a pre-6.3.0 version of Carbon Black EDR. If 6.3.0 is your first version of Carbon Black EDR or if you are using a Carbon Black Hosted EDR instance, you do not need to review this section.

Beginning with version 6.1.0, Carbon Black EDR included tokenization option that improved command-line searches. This is standard for Carbon Black Hosted EDR instances, and beginning with version 6.3.0, it is also standard for Carbon Black EDR installations. It adds the following specific improvements, which are described in more detail below:

- More special characters are removed before tokenization.
- Forward slash “/” is interpreted as a command line switch or a path character depending upon position.
- Colon “:” is interpreted as part of a drive letter token or converted to white space depending upon position and repetition.
- File extensions are stored as a separate token as well as part of a file or path name.
- Wildcards are supported in non-leading positions within a query.

These changes result in simpler queries, better and faster search results, and reduced storage requirements for tokenized command lines.

Note If you upgraded from a pre-6.3.0 Carbon Black EDR release and configured Watchlists that use command line queries, these might require a re-write to take advantage of the new tokenization. Review your Watchlist entries to make sure they return the intended results.

Example: Enhanced vs. Legacy Tokenization

The following example shows how the enhanced tokenization in Carbon Black EDR version 6.3.0 differs from the previous version. It can help you convert some older queries to the new standard:

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

Using legacy tokenization, the command was broken into the following tokens:

```
“c:

windows

system32

rundll32.exe”

d

srrstr.dll,executescheduledsppc
```

The enhanced tokenization in Carbon Black EDR version 6.3.0 breaks the same command into the following tokens:

```
c:

windows

system32

rundll32.exe

.exe

/d

srrstr.dll

.dll

executescheduledsppc
```

Examples of new search capabilities due to this tokenization include:

- You can search for .exe or .dll as part of the command line query.
- Because of more complex parsing of the forward slash, you can explicitly search for a ‘/d’ command line argument and not worry about false positives from just searching for the letter ‘d’.
- You can use a wildcard and search for ‘"execute*’ if you want to find a specific term passed to the command line.
- You do not have to include extraneous single or double quote marks to find a drive letter or command path.

Retention Maximization and cmdline Searches

On the Edit Group page for a sensor group, you can specify **Retention Maximization** options that help control the information that is recorded on the server to manage bandwidth and processing costs.

See Advanced Settings.

As part of this feature, the process cmdline field for parent processes store also store the cmdlines of their child processes (childprocs) that are affected by a retention setting. This is done because these childprocs do not have process documents of their own to store this information and so the expanded parent cmdline provides a way to search cmdlines for processes no longer recorded separately.

A side-effect of including the cmdlines of these childprocs in the parent's cmdline info is that a cmdline search intended to match only the parent process's cmdline will also match against the children. This can result in the parent process getting falsely tagged as a feed hit based on matching a childproc that was not judged to be interesting enough to justify the creation of a complete process doc. Keep this in mind when choosing **Retention Maximization** settings.

Field Type: count

An integer value. If it exists, the values are from `0` to `MAXINT` . It supports two types of search syntaxes.

- `X`: Matches all fields with precisely `X` . For example, `modload_count:34` for processes with exactly 34 modloads.
- `[X TO Y]`: Matches all fields with counts `>=` `X` and `<=` `Y` . For example, `modload_count:[1 TO 10]` for processes with 1 to 10 modloads.

In both cases, either `X` or `Y` can be replaced by the wildcard *. For example:

`netconn_count:*` for any process where the `netconn_count` field exists.

`netconn_count:[10 TO *]` for any process with more than 10 network connections.

Field Type: datetime

Datetime fields have five types of search syntaxes

- `YYYY-MM-DD` matches all entries on this day, for example, `start:2021-12-01` for all processes started on Dec 1, 2021.
- `YYYY-MM-DDThh:mm:dd` matches all entries within the next 24 hours from this date and time, for example, `start:2021-12-01T22:15:00` for all processes started between Dec 1, 2021 at 22:15:00 to Dec 2, 2021 at 22:14:59.
- `[YYYY-MM-DD TO YYYY-MM-DD]` matches all entries between, for example, `start:[2021-12-01 TO 2021-12-31]` for all processes started in Dec 2021.

- `[YYYY-MM-DDThh:mm:ss TO YYYY-MM-DDThh:mm:ss]` matches all entries between, for example, `start:[2021-12-01T22:15:00 TO 2021-12-01:23:14:59]` for all processes started in Dec 1, 2021 within the given time frame.
- `-Xh` relative time calculations matches all entries with a time between `NOW-10h` and `NOW` . Support units supported are h: hours, m: minutes, s: seconds as observed on the host, for example, `start:-24h` for all processes started in the last 24 hours.

As with counts, `YYYYMMDD` can be replaced the wildcard *, for example, `start:[2022-01-01 TO *]` for any process started after 1 Jan 2022.

Field Type: domain

Domains are split into labels for query purposes. For example, “ `example.com` ” is split into “ `example` ” and “`com`”.

If provided in a query, “dot” separator characters (.) between labels are maintained to enable position-dependent domain searches.

This has the following results:

- *Leading dot after the label, no trailing dot* – Returns results for matching labels that are at the *end* of the domain name.
- *Trailing dot after the label, no leading dot* – Returns results for matching labels that are at the *beginning* of the domain name.
- *Leading and trailing dots surrounding the label* – Returns results for matching labels that are in the middle of the domain name (i.e., not the first or last label).
- *Two labels with a dot between them* – Treated as a search for the entire phrase, and so returns results for domains that include the entire string.
- *No dot separators* – Returns results for any domain that includes the query string anywhere in the domain name.

The following table provides examples of these different domain searches:

Search	If domain is example.com	If domain is example.com.au
domain:com	**match**	**match**
domain:.com	**match**	no match
domain:.com.	no match	**match**
domain:com.	no match	no match
domain:example.	**match**	**match**
domain:example.com	**match**	no match

Field Type: integer

Integer fields are integer values (whole numbers, including 0). If it exists, the values are from `0` to `MAXINT`.

Two types of search syntax are supported:

- `X`: Matches all fields with precisely `X`. For example, `fileless_scriptload_cmdline_length:2048` for processes with fileless scriptloads with command line contents containing exactly 2048 characters.
- `X TO Y`: Matches all fields with integer values `>=X` and `<=Y`. For example, `fileless_scriptload_cmdline_length:[1 TO 2048]` for processes with fileless scriptloads with command line contents containing between 1 and 2048 characters.

In both cases, either `X` or `Y` can be replaced with a wildcard `*` (if the **Block Searches with Leading Wildcards** setting in the **Process Search Settings** section of the Advanced Settings page is disabled). For example, `fileless_scriptload_cmdline_length:*` for any processes with fileless scriptloads where the `fileless_scriptload_cmdline_length` field exists (command line contents containing any number of characters). `fileless_scriptload_cmdline_length:[1 TO *]` for any processes with fileless scriptloads with command line contents containing more than 1 character.

Field Type: ipaddr

IP addresses are searched with a CIDR notation.

```
(ip)/(netmask)
```

If the netmask is omitted, it is presumed to be 32.

For example:

`ipaddr:192.168.0.0/16` or `ipaddr:10.0.1.1`

Field Type: ipv6addr

IPv6 addresses are searched with a CIDR notation.

(ip)/(netmask)

If the netmask is omitted, it is assumed to be 32.

For example:

`ipv6addr:fe00:b9:266:2011:28dc:43d4:3298:12e2` or `ipv6addr:fe00:b9:266:2011::0/50`

Field Type: keyword

Keywords are `text` fields with no tokenization. The term that is searched for must exactly match the value in the field; for example, `process_name:svchost.exe`.

Queries containing wildcards can be submitted with keyword queries.

For example:

`process_name:ms*.exe` .

Field Type: md5

md5 fields are keyword fields with an md5 hash value.

The term searched for must exactly match the value in the field.

For example:

`process_md5:6d7c8a951af6ad6835c029b3cb88d333` .

Field Type: path

Path fields are special text fields. They are tokenized by path hierarchy.

`path:c:\windows` .

For a given path, all subpaths are tokenized. For example:

```
c:\windows\system32\boot\winload.exe
```

is tokenized as:

```
c:\windows\system32\boot\winload.exe

windows\system32\boot\winload.exe

system32\boot\winload.exe

boot\winload.exe

winload.exe
```

Wildcard Searches

For queries involving path segments that are not tokenized, wildcard searches can be submitted.

For example, you can enter:

```
path:system*
```

for any path that has `system` as sub-path in it.

Modload Path Searches

When performing a loadable module filename (modload) search, leading forward and back slashes are tokenized.

You do not have to remove the leading slash for modload path searches, although it is recommended.

For example:

```
\boot\winload.exe
```

should be entered as:

`boot\winload.exe`

Regmod Path Searches

When performing a Windows registry (regmod) search, a few important search caveats exist.

- If a regmod search term contains `controlset001` or `controlset002` , the search term is normalized and tokenized as `currentcontrolset` . As a result, you should search by replacing `controlsetXXX` with `currentcontrolset` .

  For example:

  ```
  registry\machine\system\controlset001\services\xkzc
  ```

  should be entered as:

  ```
  regmod:registry\machine\system\currentcontrolset\services\xkzc
  ```

- The leading backslash on regmod search terms are not tokenized. For regmod searches, be sure to omit this character when submitting search terms.

  For example:

  ```
  \registry\machine\system\controlset001\services\xkzc
  ```

  should become:

  ```
  regmod:registry\machine\system\currentcontrolset\services\xkzc
  ```

Field Type: sha256

sha256 fields are keyword fields with a SHA-256 hash value.

The term searched for must exactly match the value in the field.

For example:

`process_sha256:BCB8F25FE404CDBFCB0927048F668D7958E590357930CF620F74B59839AF2A9C` .

Field Type: sign

Signature fields can be one of the eight possible values.

- `Signed`
- `Unsigned`
- `Bad Signature`
- `Invalid Signature`
- `Expired`
- `Invalid Chain`
- `Untrusted Root`
- `Explicit Distrust`

Values with whitespace must be enclosed in quotes.

For example:

`digsig_result:Signed` or `digsig_result:"Invalid Chain"`

Field Type: text

Text fields are tokenized on whitespace and punctuation. Searches are case-insensitive.

For example, the string from the product_name field:

```
Microsoft Visual Studio 2010
```

is interpreted as `microsoft AND visual AND studio AND 2010` .

Searches for any of these strings will match on the binary. Phrase queries for any two consecutive terms also match on the binary.

For example:

```
product_name: "visual studio"
```

Searching with Multiple (Bulk) Criteria

You can search for multiple IOCs by using bulk search criteria in both the Process Search and Binary Search pages.

Although you could just enter a chain of "ORed" terms, Carbon Black EDR provides special interfaces for bulk searches that do this for you when given a list of terms. You can type or paste multiple terms into a bulk search text box, following these syntax requirements:

- Each term must be on its own line.
- No punctuation is required or allowed (for example, no comma-separated lists or parentheses).
- You must use the "ipaddr:" prefix to successfully use a list of IP addresses in a bulk search.
- For most other types of data, such as md5, prefixes are optional but more efficient. See Fields in Process and Binary Searches for a table of search criteria types and their prefixes.

If a bulk search is initiated using terms without prefixes, the search is treated as a generic text search and will match the terms listed to any field. In the case of IP addresses without the "ipaddr" prefix, the search will fail because the terms are dealt with as individual numbers rather than four-part addresses.

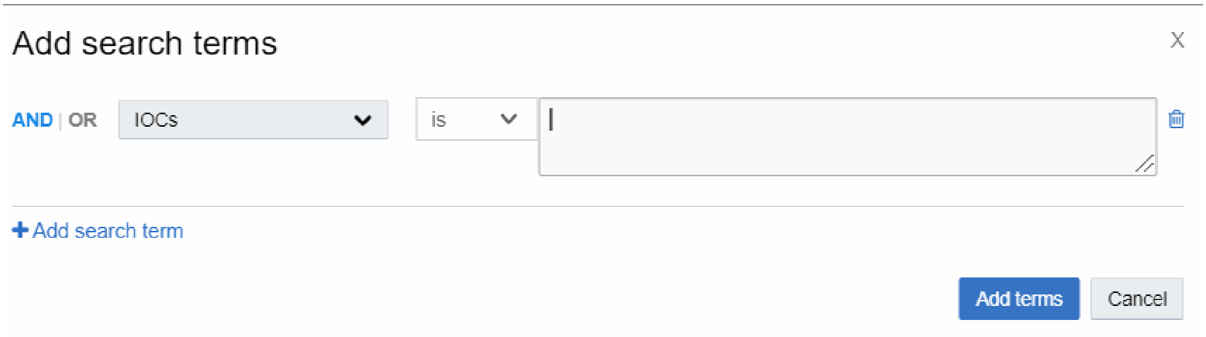
Bulk IOC searches can be added to other search criteria or used as the only criteria for a search.

Search with Multiple (Bulk) Criteria on the Process Search Page

Perform the following procedure to do a bulk IOC search on the Process Search page.

Procedure

1. On the navigation bar, click **Process Search**.
2. On the Process Search page, unless you have already entered some terms to include in your search, click the **Reset Search** button under the search box to start with a fresh search.
3. Click **Add Search Terms**. Click the **Choose Criteria** drop-down menu and click **Bulk IOC > IOCs**.
4. In the text box, type or paste the list of IOCs to search for, making sure they meet the syntax requirements described in this section.



5. For most search criteria, you are probably interested in records that match one of the items on your list; however, you also can choose to get results that do not match your terms. Use the **is / is not** toggle in the dialog to make this choice.
6. To include additional search criteria, click the **Add search term** link.
7. When you have finished defining your search, click the **Add terms** button.

   Your search is initiated and the results (if any) are shown in the table on the Process Search page. If necessary, you can continue to refine your search by using the search facet tables or you can manually enter terms.

Search with Multiple (Bulk) Criteria on the Binary Search Page

Perform the following procedure to do a bulk IOC search on the Binary Search page

Procedure

1. On the navigation bar, click **Binary Search**.
2. On the Binary Search page, unless you have already entered some terms to include in your search, click the **Reset Search Terms** button to start with a fresh search.
3. Click the **Add Criteria** dropdown menu and, under **Bulk search,** select **IOCs**.
4. In the text box, type or paste the list of IOCs to search for, making sure they meet the syntax requirements described in this section.

5 Click **Update** to apply the search terms.

Your search is initiated and any results are shown in the table on the Binary Search page. If necessary, you can continue to refine your search using the search facet tables or by manually entering terms.

Searching with Binary Joins

Some Binary Search fields can be used as part of a Process Search query.

For more information, see Fields in Process and Binary Searches.

In this case, the results returned are process instances that are backed by binaries that match the binary search criteria. This is called a joined search. For example, consider submitting the following query on the Process Search page:

```
digsig_result:Unsigned
```

This query returns all process instances that are backed by an unsigned MD5. By default, join searches are performed against the MD5 of the standalone process executable (process_md5). However, joined searches can also be performed against the MD5 of the following related events:

- `filewrites = <binary field>_filewrite`
- `parent processes = <binary_field>_parent`
- `child processes = <binary_field>_child`
- `modloads = <binary_field>_modload`

Specify the search by adding the following suffixes to the end of the binary search field:

- `filewrite`
- `parent`
- `child`
- `modload`

For example:

```
digsig_result_modload:Unsigned
```

This query returns all process instances that have loaded an unsigned module.

Note Process searches involving large binary joins are blocked by default beginning in Carbon Black EDR version 6.2.3. See Managing High-Impact Queries to modify this behavior.

Example Searches

This section provides examples of Process, Binary, and Threat Intelligence searches.

Process Search Examples

This topic provides example Process Search query strings and their results.

Example Query Strings	Result
domain:www.carbonblack.com	Returns all processes with network connections to or from domains matching the given FQDN.
domain:.com	Returns all processes with network connections to or from domains matching `*.com`
domain:.com.	Returns all processes with network connections to or from domains matching the form `*.com.*`
domain:www.	Returns all processes with network connections to or from domains matching the form `www.*`
domain:microsoft	Returns all processes with network connections to or from domains matching `*.microsoft OR *.microsoft.* OR microsoft.*`
ipaddr:127.0.0.1	Returns all processes with network connections to or from IP address `127.0.0.1`
ipaddr:192.168.1.0/24	Returns all processes with network connections to or from IP addresses in the network subnet `192.168.1.0/24`
ipv6addr:fe00:b9:266:2011:28dc:43d4:3298:12e2	Returns all processes with network connections to or from IPv6 address `fe00:b9:266:2011:28dc:43d4:3298:12e2`
ipv6addr:fe00:b9:266:2011::0/50	Returns all processes with network connections to or from IPv6 addresses in the range of network subnet `fe00:b9:266:2011::0/50`
modload:kernel32.dll	Returns all processes that loaded a module `kernel32.dll` (accepts path hierarchies).
modload:c:\windows\system32\sxs.dll	Returns all processes that loaded a module matching path and file `sxs.dll` (accepts path hierarchies).
path:c:\windows\system32\notepad. exe	Also returns all processes with the matching path (accepts path hierarchies).
regmod:\registry\machine\system\currentcontrolset\control\deviceclasses* **Notes:** Substitute “controlset001” or “controlset002” with “currentcontrolset”, as shown in this example query string. The regmod event in the process document still uses the original string, but searches must always use "currentcontrolset". regmod searches must include the complete path string or use wildcards. Searches for partial regmod paths without wildcards never yield results.	Returns all processes that modified a registry entry with the matching path (accepts path hierarchies).
path:excel.exe	Returns all processes with the matching path (accepts path hierarchies).

Example Query Strings	Result
cmdline:backup	Returns all processes with matching command line arguments.
hostname:win-5ikqdnf9go1	Returns all processes executed on the host with matching hostname.
group:"default group"	Returns all processes executed on hosts with matching group name (use of quotes are required when submitting two-word group names).
host_type:workstation	Returns all processes executed on hosts with matching type (use of quotes are required when submitting two-word host types).
username:system	Returns all processes executed with the matching user context.
process_name:java.exe	Returns all processes with matching names.
parent_name:explorer.exe	Returns all processes executed by a parent process with matching names.
childproc_name:cmd.exe	Returns all processes that executed a child process with matching names.
md5:5a18f00ab9330ac7539675f3f326cf11	Returns all processes, modified files, or loaded modules with matching MD5 hash values.
process_md5:5a18f00ab9330ac7539675f3f326cf11	Returns all processes with matching MD5 hash values.
parent_md5:5a18f00ab9330ac7539675f3f326cf11	Returns all processes that have a parent process with the given MD5 hash value.
filewrite_md5:5a18f00ab9330ac7539675f3f326cf11	Returns all processes that modified a file or module with matching MD5 hash values.
childproc_md5:5a18f00ab9330ac7539675f3f326cf11	Returns all processes that executed a child process with matching MD5 hash values.
<type>_count:*	Returns all processes that have xxx_count field > 0, where type is one of modload, filemod, regmod, netconn, or childproc.
<type>_count:10	Returns all processes that have xxx_count field = 10, where type is one of modload, filemod, regmod, netconn, or childproc.
<type>_count:[10 TO 20]	Returns all processes that have xxx_count field >= 10 and <= 20, where type is one of modload, filemod, regmod, netconn, or childproc.
<type>_count:[10 TO *]	Returns all processes that have xxx_count field >= 10, where type is one of modload, filemod, regmod, netconn, or childproc.
<type>_count:[* TO 10]	Returns all processes that have xxx_count field < 10, where type is one of modload, filemod, regmod, netconn, or childproc.
start:2011-12-31	Returns all processes with a start date of 2011-12-31 (as observed on the host).

Example Query Strings	Result
start:[* TO 2011-12-31]	Returns all processes with a start date earlier than or equal to 2011-12-31 (as observed on the host).
start:[* TO 2011-12-31T22:15:00]	Returns all processes with a start date earlier than or equal to 2011-12-31 at 22:15:00 (as observed on the host).
start:[2011-12-31 TO *]	Returns all processes with a start date later than or equal to 2011-12-31 (as observed on the host).
start:[2011-12-31T09:45:00 TO *]	Returns all processes with a start date later than or equal to 2011-12-31 at 09:45:00 (as observed on the host).
start:*	Returns processes with any start date (as observed on the host).
start:[* TO *]	Returns processes with any start date (as observed on the host).
start:-10h	Returns all processes with a start time between NOW-10h and NOW. Units supported are, h: hours, m: minutes, s: seconds (as observed on the host).
last_update:2011-12-31	Returns all processes last updated on date 2011-12-31 (as observed on the host).
last_update:[* TO 2011-12-31]	Returns all processes last updated on a date earlier than or equal to 2011-12-31 (as observed on the host).
last_update:[* TO 2011-12-31T22:15:00]	Returns all processes last updated on a date earlier than or equal to 2011-12-31 at 22:15:00 (as observed on the host).
last_update:[2011-12-31 TO *]	Returns all processes last updated on a date later than or equal to 2011-12-31 (as observed on the host).
last_server_update:[2011-12-31T09:45:00 TO *]	Returns all processes last updated on a date later than or equal to 2011-12-31 at 09:45:00 (as observed at the server).
last_server_update:*	Returns processes with any update date (as observed on the server).
last_server_update:[* TO *]	Returns processes with any update date (as observed on the server) within the range provided.
last_server_update:-10h	Returns all processes last updated between NOW-10h and NOW. Units supported are h: hours, m: minutes, s: seconds (as observed on the server).
process_id:<guid>	Returns the process with the given process id, where `<guid>` is a signed 64-bit integer.
parent_id:<guid>	Returns the process with the given parent process id, where `<guid>` is a signed 64-bit integer.
sensor_id:<guid>	Returns processes executed on host with given sensor id, where `<guid>` is an unsigned 64-bit integer.

Binary Search Examples

This topic provides example Binary Search query strings and their results.

Example Query Strings	Result
md5:5a18f00ab9330ac7539675f326cf11	Returns all binaries with matching MD5 hash values.
digsig_publisher:Oracle	Returns all binaries with a digital signature publisher field with a matching name.
digsig_issues:VeriSign	Returns all binaries with a digital signature issuer field with a matching name.
digsig_subject:Oracle	Returns all binaries with a digital signature subject field with a matching name.
digsig_prog_name:Java	Returns all binaries with a digital signature program name field with a matching name.
digsig_result:Expired	Returns all binaries with a digital signature status of `<status>` .
digsig_sign_time:2011-12-31	Returns all binaries with a digital signature date of 2011-12-31.
digsig_sign_time:[* TO 2011-12-31]	Returns all binaries with a digital signature date earlier than or equal to 2011-12-31.
digsig_sign_time:[2011-12-31 TO *]	Returns all binaries with a digital signature date later than or equal to 2011-12-31.
digsig_sign_time:*	Returns binaries with any digital signature date.
digsig_sign_time:[* TO *]	Returns binaries with any digital signature date within the range provided.
digsig_sign_time:-10h	Returns all binaries with a start time between NOW-10h and NOW. Units supported are h: hours, m: minutes, s: seconds.
<type>_version:7.0.170.2	Returns all binaries with matching version, where `<type>` is product or file.
product_name:Java	Returns all binaries with matching product name.
company_name:Oracle	Returns all binaries with matching company name.
internal_name:java	Returns all binaries with matching internal name.
original_filename:mtxoci.dll	Returns all binaries with matching filename.
observed_filename:c:\windows\system32\mtxoci.dll	Returns all binaries that have been observed to run on or were loaded with the given path.
<type>_mod_len:[* TO 10]	Returns all binaries that have `<type>_mod_len` (module length in bytes) field < 4096, where type is original or copied.
<type>_desc:“database support”	Returns all binaries that have `<type>_desc` field with matching text, where type is file or product.
legal_<type>:Microsoft	Returns all binaries with matching `legal_<type>` field text, where type is trademark or copyright.
<type>_build:“Public version”	Returns all binaries with matching `<type>_build` field text, where type is special or private.
is_executable_image:True or False	Boolean search (case insensitive) returning all binaries that are executable or not executable.

Example Query Strings	Result
is_64bit_:True or False	Boolean search (case insensitive) returning all binaries that are 64-bit or not 64-bit.
watchlist_4:[2014-04-01 TO 2014-09-31]	Returns all binaries that matched watchlist 4 during the time period shown.

Threat Intelligence Search Examples

This topic provides example Threat Intelligence Search query strings and their results.

Any document matching a threat intelligence feed is tagged with an `alliance_score_<feed>` field, where the value is a score from -100 to 100.

<feed> is the “short name” of the threat intelligence feed, such as **`nvd`** or **`isight`** .

For any threat intelligence feed, you can click the **View Hits** button to discover the feed’s short name.

For more information, see Chapter 14 Threat Intelligence Feeds.

Example Query Strings	Result
alliance_score_ *<feed>* : `*`	Returns all binaries that have *<feed>* score > 0.
alliance_score__score_ *<feed>* : `10`	Returns all binaries that have *<feed>* score = 10.
alliance_score__score_ *<feed>* : `[10 TO 20]`	Returns all binaries that have *<feed>* score >= 10 and <= 20.
alliance_score__score_ *<feed>* : `[10 TO *]`	Returns all binaries that have *<feed>* score >= 10.
alliance_score__score_ *<feed>* : `[* TO 10]`	Returns all binaries that have *<feed>* score < 10.

Ingress Filtering

13

Ingress filtering is a technique to manage data retention. This section describes how to perform common tasks related to ingress filters.

In earlier versions of Carbon Black EDR, ingress filters could only be set up by using an API. With the release of Carbon Black EDR version 7.1.0, Global Administrators can view, create, modify, and delete ingress filters in the Carbon Black EDR console.

Note Not all features or fields are currently exposed in the console. Advanced usage might require using the API. See Ingress Filter Examples for information about using the API.

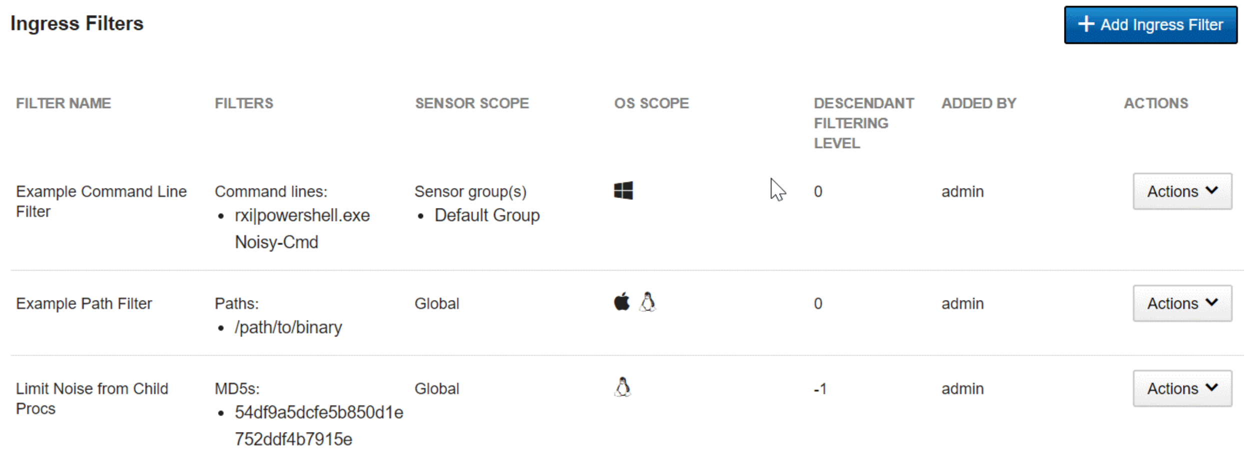
Read the following topics next:

- Viewing and Configuring Ingress Filters
- Adding an Ingress Filter

Viewing and Configuring Ingress Filters

This topic describes how to view and configure ingress filters.

- To access the Ingress Filters page, click ***Username*** **> Settings** and then click **Ingress Filters**.



Ingress Filters

FILTER NAME	FILTERS	SENSOR SCOPE	OS SCOPE	DESCENDANT FILTERING LEVEL	ADDED BY	ACTIONS
Example Command Line Filter	Command lines: • rxi|powershell.exe Noisy-Cmd	Sensor group(s) • Default Group	⊞	0	admin	Actions ⌄
Example Path Filter	Paths: • /path/to/binary	Global	🍎 🐧	0	admin	Actions ⌄
Limit Noise from Child Procs	MD5s: • 54df9a5dcfe5b850d1e752ddf4b7915e	Global	🐧	-1	admin	Actions ⌄

The Ingress Filters page shows the following fields:

- **Filter Name** – A unique name for this ingress filter.

- **Filters** – The ingress filter type and definition.
- **Sensor Scope** – Defines whether the ingress filter applies to specific sensor groups, individual sensors, or to all sensors.
- **OS Scope** – Defines the operating systems to which the ingress filter applies.
- **Descendant Level** – When a process is filtered, you can also filter its children, their children, etc., up to the set number of levels. For example, you can filter by:
  - -1: All descendants
  - 0: Matched process only
  - 1: Matched process and children
  - 2: Matched process, children, and next layer of descendants

  ... and so on
- **Added by** – The user who added the ingress filter
- **Actions** – Provides a dropdown menu that lets you modify and delete ingress filters.

Regex Filters

A filter can match a portion of a field. This is useful when you filter programs that are frequently executed with different command lines, such as powershell.exe or bash. To specify a regex pattern, prefix the path, command line, or MD5 value with `rx|` (or `rxi|` to specify a case-insensitive match). Regex patterns must be compatible with the Java 8 Pattern class. See https://docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html.

For example, the following ingress filter blocks processes where the command line contains "powershell.exe Noisy-Cmd". Other invocations of powershell.exe are not blocked.

Command lines:

```
rxi|powershell.exe Noisy-Cmd
```

Adding an Ingress Filter

You can add an ingress filter on the Ingress Filters page or on the Process Search page.

Note You cannot modify the name or filter type after you have added the ingress filter. You cannot change the filter type (for example, MD5), but you can change the *value* of the filter (the MD5 hash).

Add an Ingress Filter on the Ingress Filters Page

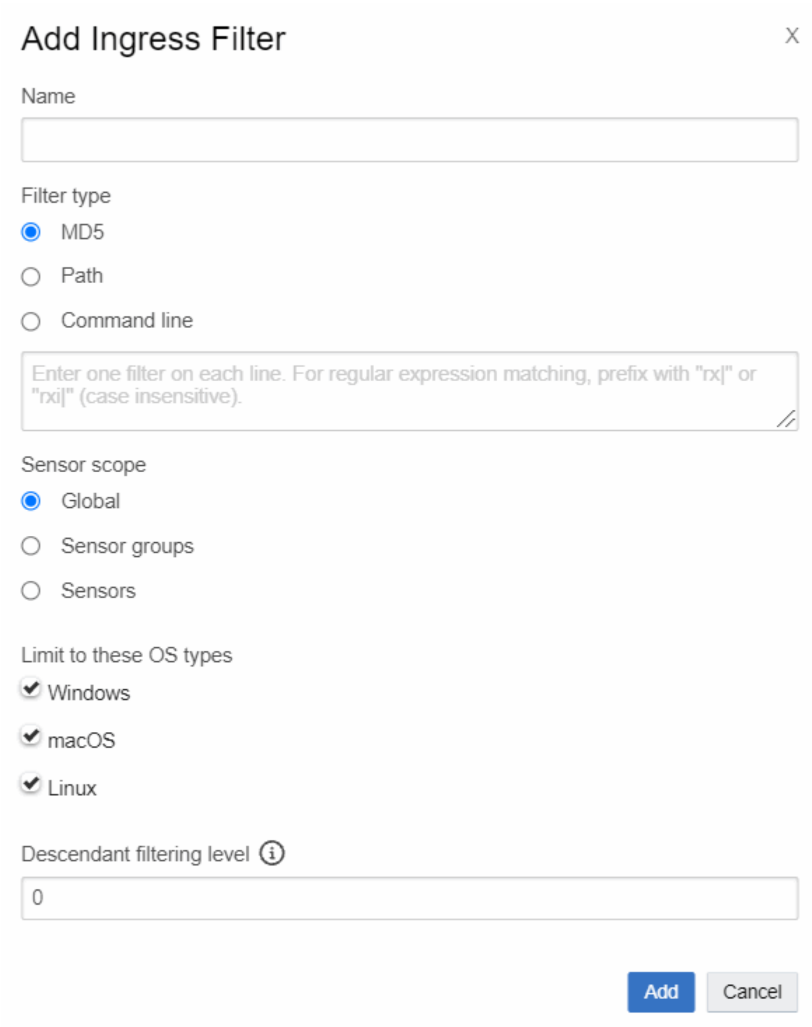
Perform the following procedure to add an ingress filter on the Ingress Filters page.

Procedure

1 Click *Username* > **Settings** and then click **Ingress Filters**.

2 Click the **Add Ingress Filter** button.

3 Fill out the following form and then click **Add**.



Add Ingress Filter

Name

Filter type

● MD5

○ Path

○ Command line

Enter one filter on each line. For regular expression matching, prefix with "rx|" or "rxi|" (case insensitive).

Sensor scope

● Global

○ Sensor groups

○ Sensors

Limit to these OS types

☑ Windows

☑ macOS

☑ Linux

Descendant filtering level

0

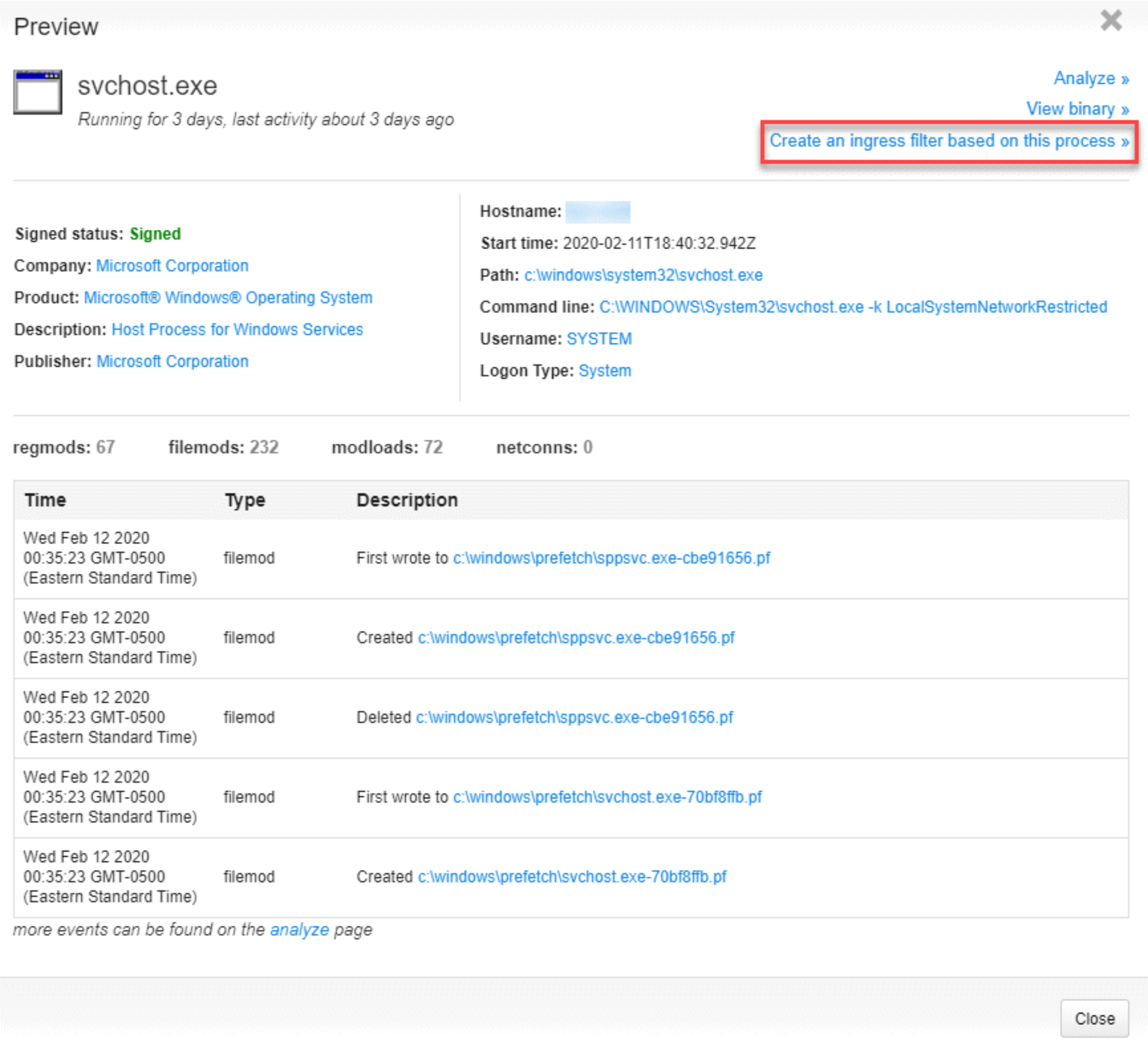
Add an Ingress Filter on the Process Search Page

Perform the following procedure to add an ingress filter on the Process Search page.

Procedure

1 On the navigation bar, click **Process Search**.

2 Perform your search and then select the process from which to create an ingress filter.



Preview

svchost.exe

Running for 3 days, last activity about 3 days ago

Analyze »

View binary »

Create an ingress filter based on this process »

Signed status: Signed

Company: Microsoft Corporation

Product: Microsoft® Windows® Operating System

Description: Host Process for Windows Services

Publisher: Microsoft Corporation

Hostname: [redacted]

Start time: 2020-02-11T18:40:32.942Z

Path: c:\windows\system32\svchost.exe

Command line: C:\WINDOWS\System32\svchost.exe -k LocalSystemNetworkRestricted

Username: SYSTEM

Logon Type: System

regmods: 67 **filemods:** 232 **modloads:** 72 **netconns:** 0

Time	Type	Description
Wed Feb 12 2020 00:35:23 GMT-0500 (Eastern Standard Time)	filemod	First wrote to c:\windows\prefetch\sppsvc.exe-cbe91656.pf
Wed Feb 12 2020 00:35:23 GMT-0500 (Eastern Standard Time)	filemod	Created c:\windows\prefetch\sppsvc.exe-cbe91656.pf
Wed Feb 12 2020 00:35:23 GMT-0500 (Eastern Standard Time)	filemod	Deleted c:\windows\prefetch\sppsvc.exe-cbe91656.pf
Wed Feb 12 2020 00:35:23 GMT-0500 (Eastern Standard Time)	filemod	First wrote to c:\windows\prefetch\svchost.exe-70bf8ffb.pf
Wed Feb 12 2020 00:35:23 GMT-0500 (Eastern Standard Time)	filemod	Created c:\windows\prefetch\svchost.exe-70bf8ffb.pf

more events can be found on the analyze page

3 Select which filter type to use. It will be pre-filled from the selected process. You can edit the values. You must provide a unique filter name.

After you have added ingress filters, you can modify or delete them by using the **Actions** dropdown menu options on the Ingress Filters page.

Threat Intelligence Feeds

14

This section describes threat intelligence feeds that can be enabled on a Carbon Black EDR server to enhance the verification, detection, visibility, and analysis of threats on your endpoints.

Threat intelligence feeds are streams of reports about IOCs and patterns of behaviors found in the wild by a variety of services and products. One or more feeds can be integrated into the Carbon Black EDR server and console to enhance the verification, detection, visibility, and analysis of threats on your endpoints.

The source of a feed may be from:

- Carbon Black Threat Intel and the Carbon Black Threat Research Team
- A third-party Carbon Black partner
- The information and analysis collected by:
  - Carbon Black Threat Intel Reputation
  - Carbon Black App Control threat detection tools
- Shared data collected from Carbon Black EDR customer enterprises

You can also create new feeds if needed. Some feeds do not require data collection from your server, while others require that you share information from your enterprise back to the feed provider to improve community intelligence data.

Available feeds appear on the Threat Intelligence Feeds page. You can enable or disable any feed on that page. The Carbon Black EDR server supports the following types of IOCs:

- Binary MD5s
- Binary SHA-256s
- IPv4 addresses
- IPv6 addresses
- JA3 fingerprints
- JA3S fingerprints
- DNS names

- Query-based feeds using the Carbon Black EDR process/binary search syntax to define an IOC

When a feed is enabled and IOCs from it are received, the following information and capabilities are added in Carbon Black EDR:

- **Feed results added to process and binary records** – If an IOC from a feed report matches processes or binaries reported by sensors on your endpoints, the feed results are added to the records for those processes/binaries in Carbon Black EDR. You can search and filter for processes or binaries using a feed report or score. For example, you can create a table of all processes whose National Vulnerability Database score is greater than 4.
- **Feed-based watchlists** – You can create a Carbon Black EDR Watchlist that tags a process or binary found on one of your endpoints when the score of a feed matches a specified score or falls within a specified score range.
- **Feed-based alerts** – You can configure console and email alerts when a process or binary, which is the subject of a specified feed report, is identified on an endpoint.
- **Links to feed sources** – You can link back to the source of a feed for more information, which can range from a general feed description to specific details about an IOC reported by that feed.
- **Threat Report Search** – You can search for individual threat reports from any feed that is or has been enabled.

Threat Intelligence Feed Scores

The threat intelligence feed score spectrum is as follows:

- A negative 100 (-100) score means a feed is extremely trustworthy (not malicious in any way). These scores are rare.
- A positive 100 (100) score means that a feed is extremely malicious.

Most scores will be within the 0-100 range.

Firewall Configuration for Feeds

To receive all threat intelligence that is available from Carbon Black Threat Intel, you must allow SSL access (port 443) through your firewall to the following domains:

- api.alliance.carbonblack.com:443
- threatintel.bit9.com:443

Blocking either of these domains will prevent your Carbon Black EDR server from receiving intelligence from specific feeds as well as data, such as IP location and icon matching for files.

Read the following topics next:

- Managing Threat Intelligence Feeds

- Enable and Configure a Threat Intelligence Feed
- Disable a Threat Intelligence Feed
- On-Demand Feeds from Carbon Black Threat Intel
- Creating and Adding New Threat Intelligence Feeds
- Searching for Threat Reports
- Threat Report Searches and Results
- Threat Report Details
- Ignoring Future Threat Reports
- Carbon Black EDR Airgap Feed

Managing Threat Intelligence Feeds

This section describes how to manage threat intelligence feeds.

On the Threat Intelligence Feeds page, you can:

- View the available feeds and get more information about them
- Enable or disable feeds
- Configure alerts and logging for feeds
- Change the rating used to calculate the severity assigned to IOCs from a feed
- Sync one or all feeds
- Check for new feeds
- Add a new feed
- Delete user-defined feeds
- Search for threat reports

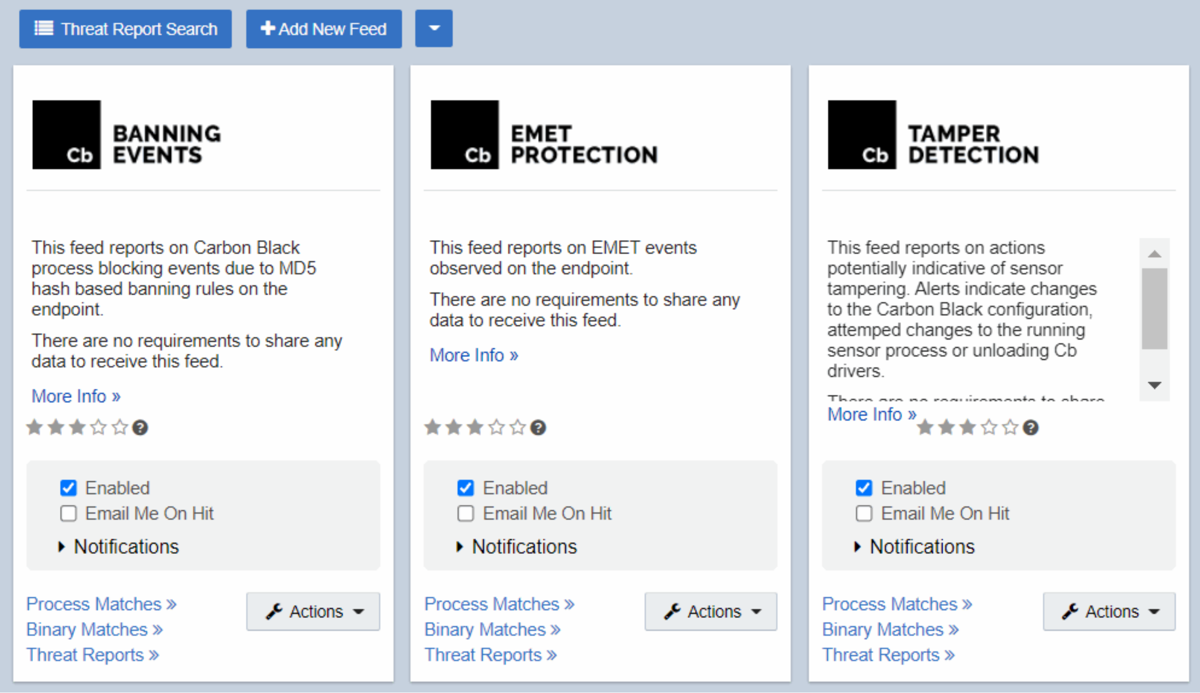
Carbon Black Threat Intel feeds are feeds that Carbon Black EDR makes available from Carbon Black EDR sources and third-party partners. These feeds can be enabled and (in some cases) disabled, but they cannot be deleted from the page.

Certain reports come from Carbon Black Threat Intel as on-demand feeds, and these do not provide their data until a process on the Process Analysis page matches their information. See On-Demand Feeds from Carbon Black Threat Intel for more details.

The EMET Protection and Banning Events feeds send their respective events to the Carbon Black EDR server regardless of whether they are enabled, but they must be enabled if you want to configure alerts and logging.

- To view the Threat Intelligence Feed page, on the navigation bar, click **Threat Intelligence**.

  The Threat Intelligence Feeds page appears:



BANNING EVENTS

This feed reports on Carbon Black process blocking events due to MD5 hash based banning rules on the endpoint.

There are no requirements to share any data to receive this feed.

☑ Enabled
☐ Email Me On Hit
▸ Notifications

EMET PROTECTION

This feed reports on EMET events observed on the endpoint.

There are no requirements to share any data to receive this feed.

☑ Enabled
☐ Email Me On Hit
▸ Notifications

TAMPER DETECTION

This feed reports on actions potentially indicative of sensor tampering. Alerts indicate changes to the Carbon Black configuration, attemped changes to the running sensor process or unloading Cb drivers.

There are no requirements to share

☑ Enabled
☐ Email Me On Hit
▸ Notifications

The **Tamper Detection** feed is enabled by default. It alerts on endpoint activity that indicates tampering with sensor activity:

You must enable other feeds. See Enable and Configure a Threat Intelligence Feed. See Creating and Adding New Threat Intelligence Feeds for information about adding user-defined feeds.

Check for New Threat Intelligence Feeds

Carbon Black EDR works with a variety of partners to provide threat intelligence feeds for the Carbon Black EDR server. You can add new partners and feeds to your server.

The **Check for new feeds** option on the Threat Intelligence Feeds page lets you check for the most recent feeds. It also removes feeds if they are no longer available. (However, existing reports and tagged processes/binaries will still identify these feeds.)

Procedure

1 On the navigation bar, click **Threat Intelligence**.

2 Click the **Action** menu (down-arrow at the top of the page).

3 Click **Check for new feeds**.

If new feeds are available, they are added to the Threat Intelligence Feeds page.

Synchronize Threat Intelligence Feeds

Threat intelligence feeds are updated periodically by the feed sources. To make certain that all feeds are up-to-date, use the **Sync All** command.

Procedure

1 On the navigation bar, click **Threat Intelligence**.

2 On the Threat Intelligence Feeds page, click the **Action** menu (down-arrow at the top of the page) and click **Sync All**.

All feeds are synced with the latest data on the Threat Intelligence Feeds page.

Note You can sync feeds individually. Sync commands for individual feeds are available on the **Action** menu in the feed panel.

Threat Intelligence Data Sharing Settings

Most of Carbon Black's threat intelligence feed partners provide a list of all of the IOCs they track, and almost all feeds require that you enable communication on the Sharing Settings page. Also, some feeds require that you enable data sharing.

Important Management of Sharing Settings is only available to Carbon Black EDR Global Administrators and Carbon Black Hosted EDR Administrators.

Enable Sharing Communications

Perform the following procedure to enable sharing communications.

Procedure

1 Click ***Username*** > **Sharing Settings**.

2 Under **General Sharing Settings**, specify sharing settings for Alliance:

- **Enable Alliance Communication** – Enables communication with Carbon Black. It also allows download of binaries from the Alliance and the ability to retrieve Alliance feeds.
- **Support the Alliance Threat Intelligence Community** – Enables your server to send threat intelligence statistics to Carbon Black, including alert resolutions, ignored reports, and feed ratings. These statistics improve the efficacy of Carbon Black-provided threat intelligence and give you a community consensus on the ratings of feeds and threat indicators.

3 Specify sharing settings for statistics and diagnostics data:

- **Enable Performance Statistics** – Enables sharing of usage, resource, and sensor statistics with Carbon Black.
- **Allow Unattended Background Upload of Diagnostics Data** – Enables the Carbon Black EDR server to do background collection of diagnostics data such as application logs and configuration files to facilitate troubleshooting with Broadcom Carbon Black Support. Requires **Enable Performance Statistics** to be enabled.

- **Allow Upload of Sensor Diagnostics Data** – This setting determines whether the Carbon Black EDR server can upload diagnostics data gathered from deployed endpoint sensors to Carbon Black for troubleshooting. Options are as follows:
  - **Disabled** – No sensor diagnostics data can be uploaded to Carbon Black.
  - **Manual** – You can manually upload sensor diagnostics data by using a utility that is installed on the sensor.
  - **Automatic** – Sensor diagnostics data is automatically uploaded when fault conditions are detected on the sensor.

Note Manual or Automatic upload of sensor diagnostics data requires the **Enable Performance Statistics** option to be selected.

4

Enable Data Sharing with Carbon Black Threat Intel Feed Partners

Perform the following procedure to enable data sharing with Carbon Black Threat Intel feed partners.

Prerequisites

To enable data sharing with feed partners in the Carbon Black Alliance, **Enable Alliance Communication** must first be selected.

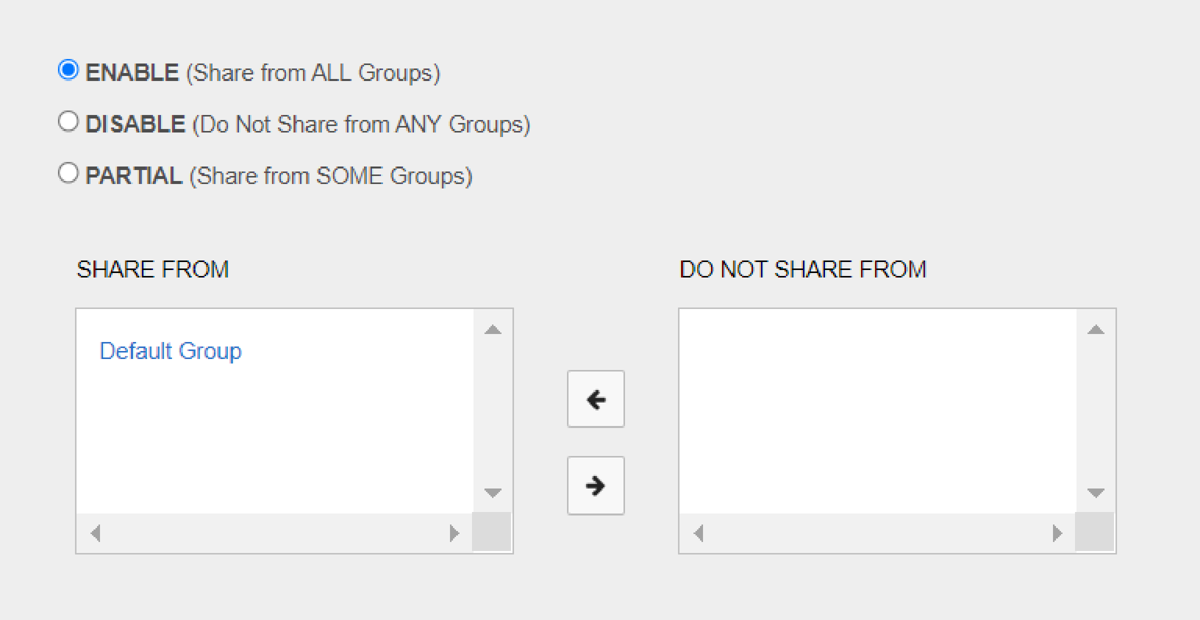
Procedure

1. Click *Username* > **Sharing Settings**.
2. Scroll to **Endpoint Activity Sharing**.
3. Decide whether to share the following types of data with Carbon Black Inspection (Complete Binaries only) or Carbon Black:
   - Binary Hashes & Metadata
   - Complete Binaries
   - Response Event Data
4. Click the current setting (**Enabled**, **Disabled**, or **Partial**) to specify the default sharing setting for sensor groups.

   Important In the resulting **Share** dialog, read the **Summary**, **Data Shared**, and **Privacy** sections carefully before making further selections.

5. The following options are available at the bottom of each **Share** dialog:
   - Select **ENABLE (Share from ALL Groups)** to share data from endpoints in all sensor groups.

- Select **DISABLE (Do Not Share from ANY Groups)** to disable data sharing from endpoints in all sensor groups.
- Select **PARTIAL (Share from SOME Groups)** to share data from endpoints in some sensor groups. Use the arrows between the **SHARE FROM** and **DO NOT SHARE FROM** windows to choose the groups that are allowed to share data.



● **ENABLE** (Share from ALL Groups)

○ **DISABLE** (Do Not Share from ANY Groups)

○ **PARTIAL** (Share from SOME Groups)

SHARE FROM: Default Group

DO NOT SHARE FROM:

6 Click **Share** to begin sharing the data.

Enable and Configure a Threat Intelligence Feed

Perform the following procedure to enable and configure a threat intelligence feed.

Procedure

1 On the navigation bar, click **Threat Intelligence**.

2 Locate the feed and click **Enabled**.

3 Configure the feed using the following options and controls:

Field/Menu	Description
More info	Link to the feed provider’s website for technical information about the feed and general information about the provider and its products.
(Rating)	Rating of this threat intelligence feed by the community of Carbon Black EDR users. The default rating is three stars. You can click a star to modify the rating of this feed on your server. The rating affects the severity assigned to alerts coming from this feed, which can affect the order of alerts if they are sorted by severity.

Field/Menu	Description
Enabled	If selected, the threat intelligence feed is enabled; otherwise it is disabled. **Note** Most feeds also require that you select **Enable Alliance Communication** on the Sharing page. Also, feeds that upload data from your server require that you opt into hash sharing with that feed. See Threat Intelligence Data Sharing Settings.
Email Me on Hit	IOCs from this feed that reference a process or binary that is recorded on this Carbon Black EDR server cause an email alert to be sent to the logged-in console user. See Enabling Email Alerts. **Note** Only Carbon Black EDR Global Administrators or Carbon Black Hosted EDR Administrators can change this setting.
Notifications menu	■ **Create Alert** – Indicators from this feed that reference a process or binary that is recorded on this Carbon Black EDR server cause a console alert. See Enabling Console Alerts. ■ **Log to Syslog** – IOCs from this feed that reference a process or binary that is recorded on this Carbon Black EDR server are included in Syslog output from this Carbon Black EDR server. See the *Carbon Black EDR Integration Guide* for details on configuring SYSLOG output. **Note** Only Carbon Black EDR Global Administrators or Carbon Black Hosted EDR Administrators can change this setting.
Process Matches	Link to the Process Search page with the results of a search that shows each process that matches IOCs from this feed. See Chapter 10 Process Search and Analysis.
Binary Matches	Link to the Binary Search page with the search results showing each binary that matches IOCs from this feed. See Chapter 11 Binary Search and Analysis.
Threat Reports	Link to the Threat Reports search page filtered to show any Threat Reports from this feed. See Searching for Threat Reports.
Actions menu	The **Actions** menu includes the following commands: ■ **Create Watchlist** – Creates a Watchlist, which is a saved search whose results are processes or binaries that match IOCs that this feed reports. ■ **Incremental Sync** – Adds report data from this feed that has been observed since the previous synchronization. ■ **Full Sync** – Rewrites all report data from this feed.

Disable a Threat Intelligence Feed

Perform the following procedure to disable a threat intelligence feed.

Procedure

1 On the navigation bar, click **Threat Intelligence**.

2 Identify the feed to disable.

3 Deselect the **Enabled** check box within that feed panel.

If you disable a feed, its reports remain on the server and incoming data will be tagged against the locally existing IOCs that it reported. However, when you disable a feed:

- Reports from these feeds about IOCs will not be downloaded for scanning.

- For feeds that require data to be sent to them, new binary MD5s from your sensors will not be sent.

On-Demand Feeds from Carbon Black Threat Intel

Carbon Black Threat Intel provides a rich variety of intelligence and capabilities about files, domain names, IP addresses, and associated patterns of compromise, including IOCs, reputation, and attack classification.

Examples of these intelligence types include:

- Trust and threat ratings
- Domain/IP reputation and context
- Icon matching to help identify threats masquerading as files of another type
- Detection feeds of behavioral patterns of compromise

Some of this intelligence can be enabled or disabled through feeds listed on the Threat Intelligence Feeds page, and this information is added to process data as soon as the feed is received.

Other intelligence is made available to the Carbon Black EDR server when a process, pattern, or other IOC that is part of the Carbon Black Threat IntelCarbon Black Threat Intel database is viewed on the Process Analysis page. The information in these on-demand feeds includes the following:

- **Damballa malware classification and context** – Carbon Black Threat Intel provides an enhanced network-to-endpoint attack classification through its integration with Damballa's threat intelligence on malicious destinations, advanced threat actor groups, and command-and-control communications. This information is added to attack classifications when a Process Analysis page containing a relevant domain name is displayed.
- **Geolocation information for network connections** – The location of addresses identified in both inbound and outbound connections is provided.
- **Domain and IP reputation** – Carbon Black Threat Intel computes a reputation score for domains using various inputs, information, and algorithms inside the cloud. This reputation score is displayed for domain names for which a score is available.

Note For on-demand feed information to become available and displayed for a process, the sensor group for which the process was reported must be configured to send relevant data to the Carbon Black Threat Intel for analysis. This requires explicitly opting in to share Carbon Black EDR events with Carbon Black Threat Intel. This is not enabled by default; you can enable it in the **Response Event Data** row in the **Endpoint Activity Sharing** section of the Sharing page. See Threat Intelligence Data Sharing Settings.

Creating and Adding New Threat Intelligence Feeds

You can create and add new threat intelligence feeds to a Carbon Black EDR server.

A threat intelligence feed can be created in any language that allows for building JSON, or you can build it by hand. One way to build a feed is to use the Carbon Black Feeds API (CBFAPI), which is located on github at:

https://github.com/carbonblack/cbfeeds.

The CBFAPI is a collection of documentation, example scripts, and a helper library to help create and validate Carbon Black EDR feeds. Regardless of how a feed is created, the feed file must match the feed structure (or schema) that the *Feed Structure* section of the CBFAPI documentation defines.

You have several options about the specification you provide when adding a new feed to a Carbon Black EDR server. The minimum requirement is that you provide a URL to the feed.

Add a New Threat Intelligence Feed

Perform the following procedure to add a newly created threat intelligence feed.

Prerequisites

Confirm that the feed you have created follows the *Feed Structure* instructions in the CBFAPI documentation on github.

Procedure

1 On the navigation bar, select **Threat Intelligence**.

2 On the Threat Intelligence Feeds page, click **Add New Feed**.

3 In the **Edit Alliance Feed** dialog box, do one of the following:

- To add a feed from a URL, click the **Add from URL** tab and complete the following settings:

  Table 14-1.

Field	Description
Feed URL	Enter the URL for the feed that will be providing IOC reports.
Use Proxy	Select this option to use a proxy for the feed URL. The configuration for this proxy must be configured in advance by Carbon Black Technical Support.

Table 14-1. (continued)

Field	Description
Validate Server Cert	Select this option to require a validation check on the feed server’s certificate.
Show/Hide Feed Server Authentication Options	If the server that is providing the feed requires authentication, click the **Show ServerAuthentication Options** link and provide the following authentication information: ■ Username ■ Password ■ Public Cert ■ Private Key

- To manually add a feed, click the **Add Manually** tab and complete the following settings:

Field	Description
Name	Enter the feed name to appear in the panel.
Feed URL	Enter the URL that the Carbon Black EDR server will use to sync the data in the feed.
Provider URL	Enter the URL to the page to open when the user clicks **More Info** on the feed panel.
Summary	Enter the text that will appear in the panel to describe this feed.
Use Proxy	If the Carbon Black EDR server must access the feed URL through a proxy, the proxy is added in the proxy field.
Validate Server Cert	Indicates if the Carbon Black EDR server should validate the Feed Server certificate.
Show/Hide Feed Server Authentication Options	If the server providing the feed requires authentication, click the **Show ServerAuthentication Options** link and provide the following authentication information: ■ Username ■ Password ■ Public Cert ■ Private Key

4 Click **Save**.

If the settings you entered provide access to a feed server, the new feed appears on the Threat Intelligence Feeds page.

Searching for Threat Reports

You can search for all reports or perform a search on a page that is pre-filtered for one feed.

You can also obtain more information on the report types that are provided by a particular feed, or you might want to explore specific reports. See Threat Report Searches and Results and Threat Report Details.

Suppose you want to filter out a high volume of uninteresting reports from a feed that you otherwise you find useful. You can search for those reports on the Threat Intelligence Feeds page and mark them to be ignored in the future. See Ignoring Future Threat Reports.

Open the Search Threat Reports Page (Unfiltered)

Perform this procedure to view threat reports.

Procedure

1. On the navigation bar, click **Threat Intelligence** and then click **Threat Report Search**.
2. In the Search Threat Reports page, enter search criteria to search for the reports in which you are interested. See Threat Report Searches and Results.

Display a Table of Reports from One Threat Intelligence Feed

Perform this procedure to display a table of reports from one threat intelligence feed.

Procedure

1. On the navigation bar, click **Threat Intelligence** and then click **Threat Report Search**.
2. On the Threat Intelligence Feeds page, click the **Threat Reports** link at the bottom of the panel for the feed that contains reports you want to view.

   The Search Threat Reports page displays reports from the selected feed.

   You can further refine the search by using the available search options.

Threat Report Searches and Results

This topic describes the Search Threat Reports page.

The Search Threat Reports page is divided into three major sections:

- The top section includes the following:
  - The **Search** field and button.
  - The **Add Criteria** button, which opens a Search Criteria page.
  - The **Reset search terms** button, which resets the search and removes any search criteria you have added.
  - The **Actions** menu, which applies to the entire page.

- The middle section contains a series of filters that include the following:
  - **Feed Name** – A list of the short names (for example, “nvd” for National Vulnerability Database) of each feed that has produced a report, and the percentage of all reports that have been produced by each feed.
  - **Feed Category** – A list of feed categories and the percentage of all reports that each feed category produces. Categories can include:

    Open Source – For example, Tor or Malware Domain List.

    Partner – A member of the Carbon Black Threat Intel Partners.

    Carbon Black EDR **first party** – Feeds supplied directly from Carbon Black App Control or Carbon Black EDR products or services.
  - **Report Score** – A graph of the number of reports at different score levels.
  - **Report Creation Time** – A graph of the number of reports by creation date.
- The **Reports** table shows details for reports that match the search criteria. You can sort the reports by severity, most recently updated, or most recently added.

The Search Threat Reports page presents the following report data:

Column	Description
Description	This column includes: ■ The name of the feed that provided the report ■ The name of the specific report ■ The time elapsed since the report was received
Indicators	The column includes the number of certain elements in the report that were identified as threats: ■ MD5s – the number of suspicious files matching the MD5 hash ■ SHA-256s – the number of suspicious files matching the SHA-256 hash ■ IPs – the number of suspicious IP addresses ■ Domains – the number of suspicious domains ■ Queries – the number of queries in the report; depending on the feed, this value might be empty.
Report Score	The threat score of this report. Report scores range from minus 100 to 100, with lower scores indicating a lower threat and higher scores indicating a higher threat. Threat scores are used in the calculation of alert severity.
Ignore	Ignore any future instances of this report, so that they do not trigger alerts. See Ignoring Future Threat Reports.
Details link	Opens a Threat Report Details page for the report in this row. See Threat Report Details.

Threat Report Details

Click the **Details** link in the far right column of a threat report in the **Threat Report Search Reports** table to see details for that threat report, if available.

The information on the Threat Report Details page varies depending on the feed source and type of indicator. The following table describes the fields on this page.

Field	Description
Title	The feed name and the unique ID of the report.
Report Details	This section includes: ■ ID – the unique ID of the report ■ Link – if available, a link to the report on the website of the feed source ■ Updated – when the report was last updated ■ MD5s – the number of suspicious MD5s ■ SHA-256s – the number of suspicious SHA-256s ■ IPs – the number of suspicious IP addresses ■ Domains – the number of suspicious domains ■ Queries – the number of queries in the report
Report Tags	One or more descriptive strings from the feed provider to help explain what the report is about. For example, tags can describe a specific threat, a threat category, a targeted industry, a known threat actor, or geographic information. Not all reports have tags.
Feed Description	The description of the feed given by the provider.
Report Description	The description of this report from the feed provider.
Report Score	The threat score of this report. Report scores range from minus 100 to 100, with lower scores meaning lower threat and higher scores meaning higher threat. Threat scores are one factor in the calculation of Alert severity.
Ignore this Report?	Ignore any future instances of this report so that they do not trigger alerts.
Report Indicators	A table of indicators that the report references (IPs, MD5s, SHA-256s, domains, queries). If the Type is MD5, clicking the indicator name links to the Binary Search page for that MD5.

Ignoring Future Threat Reports

This topic presents options that you can use to ignore threat reports.

Feeds use a variety of criteria to decide whether a file or site is a threat, and you might not agree with the threat level that is indicated by all of the reports generated by certain feeds. When you review reports and determine that a report is not reporting an actual threat, you can ignore any future instances of reports by the same name.

To ignore reports, select one of the following options:

- On the Search Threat Reports page, on the **Actions** menu, click **Ignore all reports matching this search**.
- In the **Search Results** table on the Search Threat Reports page, set the **Ignore** button for one report to **Yes**.
- On the Threat Report Details page, set the **Ignore this Report** button for one report to **Yes**.

You can also mark events to be ignored using the Triage Alerts page. See Ignore Future Events for False Positive Alerts.

Carbon Black EDR Airgap Feed

The Airgap tool helps you import Carbon Black EDR-provided threat intelligence feeds into Carbon Black EDR servers that are installed inside an isolated network.

This script exports a subset of the Carbon Black Collective Defense Cloud Threat Intelligence Feeds into a set of JSON files that can be copied and imported into an airgapped Carbon Black EDR server.

The following feeds are supported by this tool:

- abuse.ch Indicators of Compromise
- Malware Domain List
- Tor exit nodes
- Carbon Black Advanced Threat Indicators
- Carbon Black Community Feed
- Carbon Black Early Access Feed
- Carbon Black Suspicious Indicators
- Carbon Black Endpoint Visibility Feed
- Carbon Black Known IOC Feed
- SANS Threat Hunting Feed
- AlienVault Open Threat Exchange
- Facebook Threat Exchange TLP White Indicators
- ThreatConnect
- MITRE ATT&CK Feed

Other Carbon Black Collective Defense Cloud feeds cannot be exported because they require the target Carbon Black EDR server to be online and actively communicating with the Collective Defense Cloud.

For support of the Airgap tool:

- View all API and integration offerings on the Carbon Black EDR Developer Network, together with reference documentation, video tutorials, and how-to guides.
- Use the Developer Community Forum to discuss issues and get answers from other API developers in the Carbon Black Community.
- Report bugs and change requests to Broadcom Carbon Black Support.

Run the Airgap Tool

Use the Airgap tool to provide feeds from the Carbon Black Collective Defense Cloud to an airgapped Carbon Black EDR server.

The source server runs the script in export mode to download the feeds from the Carbon Black Collective Defense Cloud and save them to a local directory. This directory is then burned to CD, copied to USB, or otherwise transferred to the destination server through a secure means. The folder includes a copy of the script plus the contents of all the feeds exported from the Carbon Black Collective Defense Cloud.

After the folder arrives at the destination server, the script is run in import mode to import the feed contents into the isolated Carbon Black EDR server. This process can be repeated on a regular basis to keep the copies of the feeds on the destination server synchronized with the feeds from the Carbon Black Collective Defense Cloud.

Prerequisites

To use this tool, you need two Carbon Black EDR servers: one that has Internet access and the Carbon Black Collective Defense Cloud (the source), and one Carbon Black EDR server that is disconnected from the Internet (the destination).

Procedure

1 Run the `/usr/share/cb/cbfeed_airgap` script on the source system using an `-f` argument to indicate the folder where the feeds should be saved. This folder can be on a mounted USB stick, or a temporary directory that will be burned to CD-ROM. For example:

```
# ./usr/share/cb/cbfeed_airgap export -f /tmp/blah
exporting threat intelligence feeds to /tmp/blah
# cp -rp /tmp/blah /media/USB
# umount /media/USB
...
```

Note Include a `-v` option for verbose logging to `/var/log/cb/cli/cli.log.`

2 Copy the files to the destination server.

3 Go to the directory that contains the script and feeds folder that you copied from the source server.

4 Run the / `usr/share/cb/cbfeed_airgap` script on the destination system in import mode. For example:

```
# ./usr/share/cb/cbfeed_airgap import
importing threat intelligence feeds from /media/USB
...
```

Antimalware Scan Interface

15

A sensor event called the `fileless script load` event is recorded by the Carbon Black EDR Windows sensor.

Antimalware Scan Interface (AMSI) support is available in Carbon Black EDR Server 7.2 and later releases, together with the Carbon Black EDR Windows 7.1+ sensor.

The `fileless script load` event leverages the Antimalware Scan Interface (AMSI) (external link) support that is available in Windows 10 and Windows 2016. Endpoints must be running Windows 10 RS2 or higher for Carbon Black EDR sensors to record AMSI data.

The `fileless_scriptload` event represents each occasion when the sensor detected AMSI-decoded script content that was executed by any process on the endpoint. This consists only of fileless script content that was not stored in a file on the file system when that content was executed.

For example, you can detect when the PowerShell runtime was loaded into another process by malware, which obtains encoded PowerShell script content from a remote network server and then executes that script content directly from memory.

The sensor reports a fileless script load event to the Carbon Black EDR Server only if it originates from a script load that is not backed by an on-disk file. File-based scripts are logged locally.

Support for decoding fileless script content through AMSI is dependent on the script interpreter that integrates with the AMSI interface in Windows. Carbon Black EDR currently supports PowerShell. For information about the AMSI API, see https://docs.microsoft.com/en-us/windows/win32/amsi/dev-audience (external link).

AMSI Data

AMSI data is part of process execution metadata. A generic event type is added as part of the AMSI data stream.

All AMSI content is logged locally on the endpoint as a text file. The log is located in the sensor installation directory and is named `AmsiEvents.log` . This log contains all AMSI content that is detected by the sensor, including events that are not reported to the Carbon Black EDR server due to privacy reasons.

`AMSIEvents.log` on the endpoint is capped at 50 MB, unzipped. After that limit is reached, the log contents are migrated to a new file (`AMSIEvents.old.log`) before recreating `AMSIEvents.log`. After the second 50 MB log fills up, Carbon Black overwrites `AMSIEvents.old.log` again. Therefore, no more than two 50 MB local log files exist.

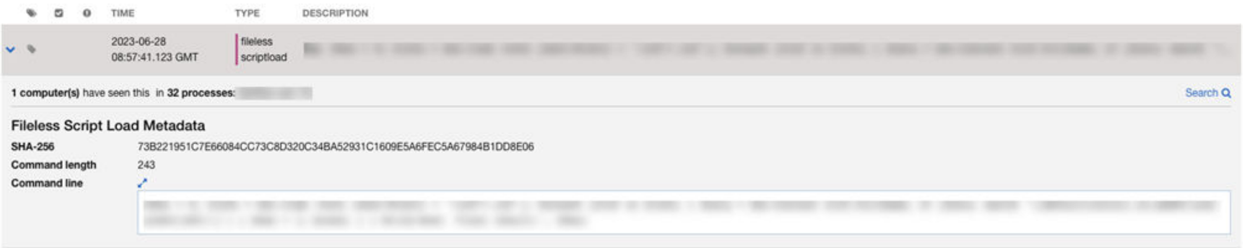
Read the following topics next:

- Using AMSI with Carbon Black EDR

Using AMSI with Carbon Black EDR

Fileless script load events collected through integration with Windows AMSI can be reported in the console and forwarded through the Event Forwarder in JSON and LEEF.

To see the raw AMSI data in the console, you can expand a `fileless_scriptload` event. The metadata that the fileless script load event captures includes the unique SHA-256 hash of the fileless script load event, the command length, and the command line content (which can be expanded to view the full content if it is truncated).



TIME	TYPE	DESCRIPTION
2023-06-28 08:57:41.123 GMT	fileless scriptload	

1 computer(s) have seen this in **32 processes:**

Fileless Script Load Metadata

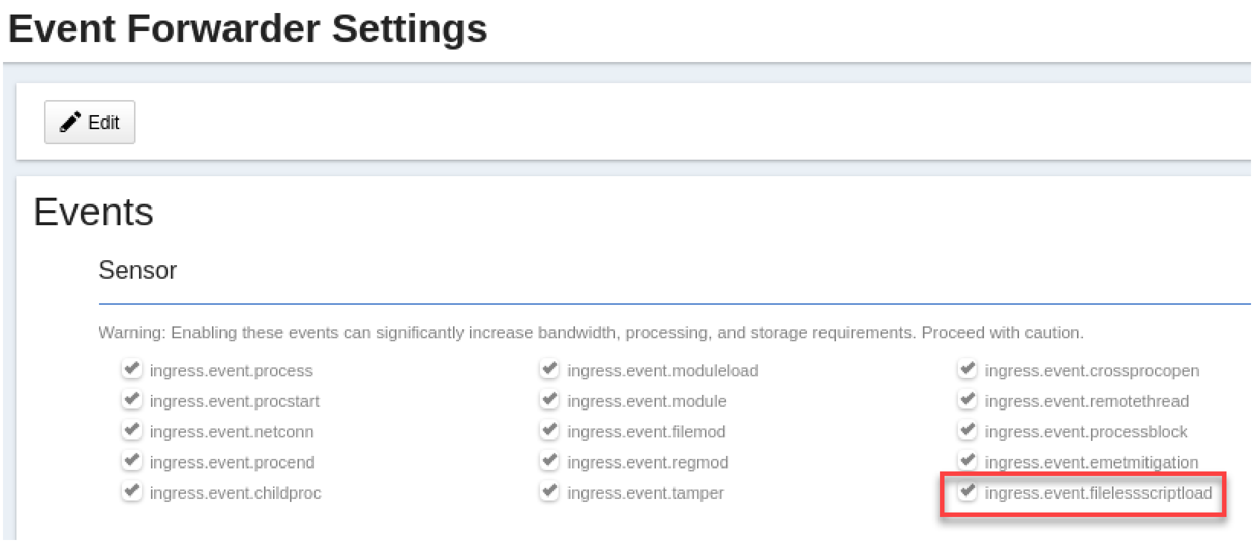
SHA-256	73B221951C7E66084CC73C8D320C34BA52931C1609E5A6FEC5A67984B1DD8E06
Command length	243
Command line	

Carbon Black EDR Windows sensors perform asynchronous RPC calls; the sensor captures commands and script contents that PowerShell executes.

Event Forwarder Settings

In the Carbon Black EDR console, you can enable fileless script load events in the Event Forwarder by checking the `ingress.event.filelessscriptload` option.

See Chapter 16 Event Forwarder.



Event Forwarder Settings

Events

Sensor

Warning: Enabling these events can significantly increase bandwidth, processing, and storage requirements. Proceed with caution.

☑ ingress.event.process
☑ ingress.event.procstart
☑ ingress.event.netconn
☑ ingress.event.procend
☑ ingress.event.childproc
☑ ingress.event.moduleload
☑ ingress.event.module
☑ ingress.event.filemod
☑ ingress.event.regmod
☑ ingress.event.tamper
☑ ingress.event.crossprocopen
☑ ingress.event.remotethread
☑ ingress.event.processblock
☑ ingress.event.emetmitigation
☑ ingress.event.filelessscriptload

Enable Fileless Script Load Events for a Sensor Group

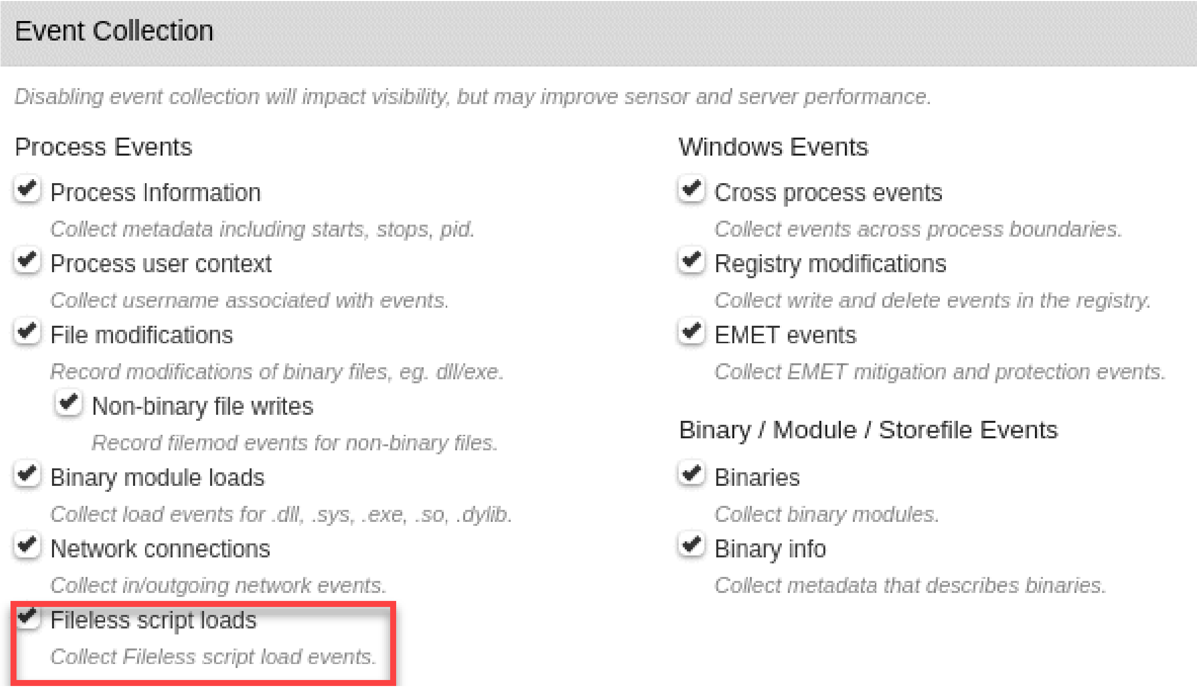
In the Carbon Black EDR console, you can toggle the collection of fileless script load events per sensor group. This is disabled by default.

Procedure

1 On the left navigation bar, click **Sensors**.

2 Select the sensor group.

3 In the **Event Collection Settings** section, select the checkbox for **Fileless script loads**.



Event Collection

Disabling event collection will impact visibility, but may improve sensor and server performance.

Process Events

☑ Process Information
Collect metadata including starts, stops, pid.
☑ Process user context
Collect username associated with events.
☑ File modifications
Record modifications of binary files, eg. dll/exe.
☑ Non-binary file writes
Record filemod events for non-binary files.
☑ Binary module loads
Collect load events for .dll, .sys, .exe, .so, .dylib.
☑ Network connections
Collect in/outgoing network events.
☑ Fileless script loads
Collect Fileless script load events.

Windows Events

☑ Cross process events
Collect events across process boundaries.
☑ Registry modifications
Collect write and delete events in the registry.
☑ EMET events
Collect EMET mitigation and protection events.

Binary / Module / Storefile Events

☑ Binaries
Collect binary modules.
☑ Binary info
Collect metadata that describes binaries.

4 Click **Save Group**.

See Chapter 6 Sensor Groups.

Event Forwarder

16

The Carbon Black EDR Event Forwarder is a standalone service that can export events (both watchlist/feed hits and raw endpoint events, if configured) from the Carbon Black EDR enterprise bus in a normalized JSON or LEEF format.

The events can be saved to a file, delivered to a network service, or automatically archived to an Amazon AWS S3 bucket. These events can be consumed by any external system that accepts JSON or LEEF, including Splunk and IBM QRadar.

The list of events to collect is configurable. By default, all feed and watchlist hits, alerts, binary notifications, and raw sensor events are exported into JSON. The configuration file for the connector is stored in `/etc/cb/integrations/event-forwarder/cb-event-forwarder.conf` .

For details on installing and manually configuring the Event Forwarder, see https://github.com/carbonblack/cb-event-forwarder .

With the release of the Carbon Black EDR version 7.1.0 Server, administrators can customize the Event Forwarder from directly within the Carbon Black EDR console. Carbon Black EDR customers must install Carbon Black EDR Event Forwarder 3.6.2 or higher (available here) to use this feature. This version of Event Forwarder is automatically available for Carbon Black Hosted EDR customers.

By default, this feature is enabled for Carbon Black Hosted EDR instances, and disabled for Carbon Black EDR deployments. You can enable the feature for Carbon Black EDR by adding `EventForwarderEnabled=true` in `cb.conf` and restarting services. For more information about `cb.conf`, see the *Carbon Black EDR Server Configuration Guide*.

Forwarding of `fileless_scriptload` events by an integration with Microsoft Antimalware Scan Interface (AMSI) through the Event Forwarder was introduced in Carbon Black EDR Server version 7.2.0 and Carbon Black EDR Windows Sensor version 7.1. However, full support of this event type, including storage, console display, and API support, is available in Carbon Black EDR Server version 7.6.0. The `fileless_scriptload` event represents each occasion when the sensor detects PowerShell script content that was executed by any process on a supported endpoint. For more information about AMSI, see the *Carbon Black EDR Integration Guide*.

Read the following topics next:

- Configure the Event Forwarder in the Console

Configure the Event Forwarder in the Console

Perform the following procedure to configure the Event Forwarder in the Carbon Black EDR console.

You do not have to stop the service to configure the Event Forwarder settings; however, you must stop and restart the service for saved changes to take affect.

Note You cannot save the configuration until after you have established a valid configuration in the **Output** section of the Event Forwarder Settings page.

Prerequisites

You must have set up the receiving service and credentials before you configure the Event Forwarder for the first time. See https://github.com/carbonblack/cb-event-forwarder.

Carbon Black validates the connection as soon as you click **Save**; therefore, it is important that the connection is viable before you set up forwarded events. If the connection is not viable, the configuration is not saved.

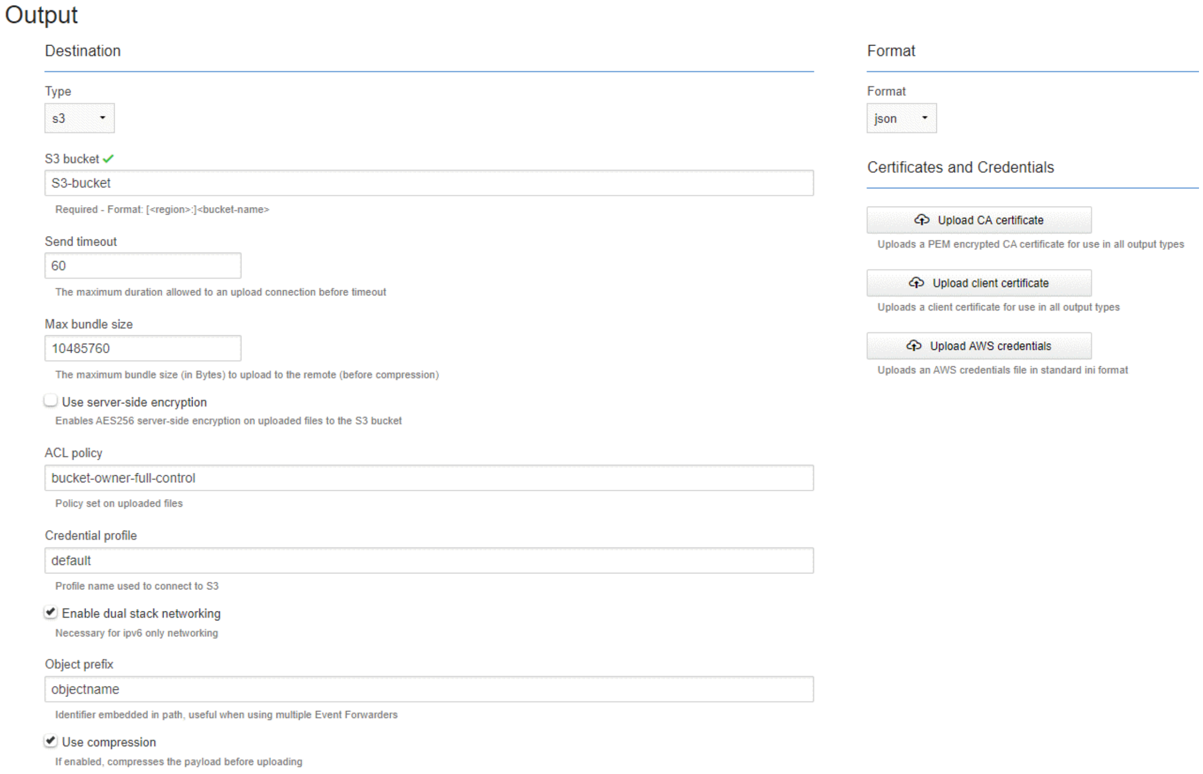
Procedure

1 On the navigation bar, click **Event Forwarder**.

   The Event Forwarder Settings page consists of four sections:

   - **Edit and status**: Allows you to edit and save or cancel changes to the configuration, displays the service status, and lets you stop/start the service.
   - **Events**: Identifies the events that will be forwarded.
   - **Output**: Configures the format and destination for the output.
   - **Certificates**: Identifies certificates to use for validation.

2 Click **Edit** at the top of the page and configure your output in the **Output** panel.



Output

Destination

Type

s3

S3 bucket ✓

S3-bucket

Required - Format: [<region>:]<bucket-name>

Send timeout

60

The maximum duration allowed to an upload connection before timeout

Max bundle size

10485760

The maximum bundle size (in Bytes) to upload to the remote (before compression)

☐ Use server-side encryption

Enables AES256 server-side encryption on uploaded files to the S3 bucket

ACL policy

bucket-owner-full-control

Policy set on uploaded files

Credential profile

default

Profile name used to connect to S3

☑ Enable dual stack networking

Necessary for ipv6 only networking

Object prefix

objectname

Identifier embedded in path, useful when using multiple Event Forwarders

☑ Use compression

If enabled, compresses the payload before uploading

Format

Format

json

Certificates and Credentials

Uploads a PEM encrypted CA certificate for use in all output types

Uploads a client certificate for use in all output types

Uploads an AWS credentials file in standard ini format

a Click the **Format** drop-down menu and select the output format. The output can be in either LEEF or JSON format. The default format is JSON.

b Select the destination type.

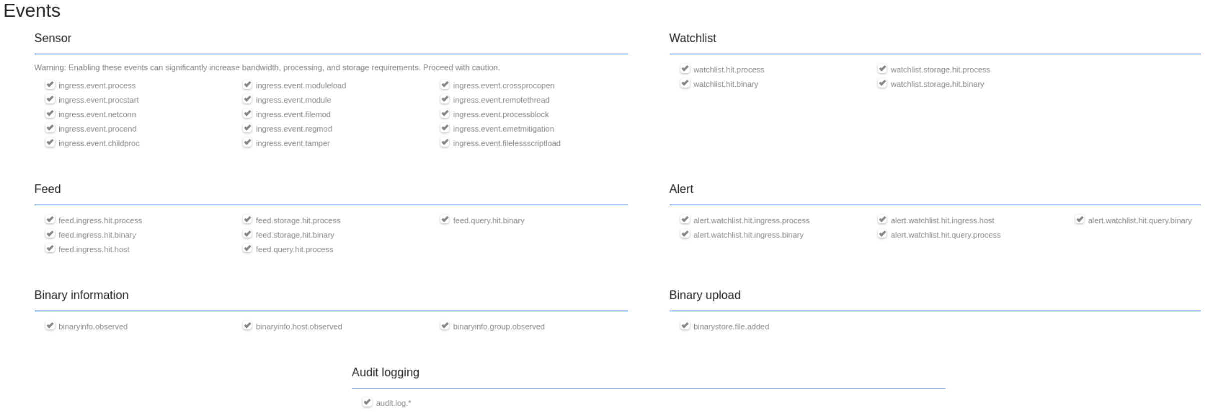
The required output parameters depend on the destination type. The default destination type is Splunk. Your options are:

- Splunk; see Event Forwarder Splunk Output.
- S3; see Event Forwarder S3 Output.
- HTTP; see Event Forwarder HTTP Output.
- Syslog; see Event Forwarder Syslog Output.

3 Click **Save**.

If the connection is viable, the configuration is saved and you can proceed. Saving the configuration immediately updates the configuration file on the server to reflect the changes.

4 Click **Edit**. In the **Events** panel, select the items to be forwarded by checking the checkboxes next to each item. To deselect an item, uncheck the checkbox.



Events

Sensor

Warning: Enabling these events can significantly increase bandwidth, processing, and storage requirements. Proceed with caution.

☑ ingress.event.process
☑ ingress.event.procstart
☑ ingress.event.netconn
☑ ingress.event.procend
☑ ingress.event.childproc
☑ ingress.event.moduleload
☑ ingress.event.module
☑ ingress.event.filemod
☑ ingress.event.regmod
☑ ingress.event.tamper
☑ ingress.event.crossprocopen
☑ ingress.event.remotethread
☑ ingress.event.processblock
☑ ingress.event.emetmitigation
☑ ingress.event.filelessscriptload

Watchlist

☑ watchlist.hit.process
☑ watchlist.hit.binary
☑ watchlist.storage.hit.process
☑ watchlist.storage.hit.binary

Feed

☑ feed.ingress.hit.process
☑ feed.ingress.hit.binary
☑ feed.ingress.hit.host
☑ feed.storage.hit.process
☑ feed.storage.hit.binary
☑ feed.query.hit.process
☑ feed.query.hit.binary

Alert

☑ alert.watchlist.hit.ingress.process
☑ alert.watchlist.hit.ingress.binary
☑ alert.watchlist.hit.ingress.host
☑ alert.watchlist.hit.query.process
☑ alert.watchlist.hit.query.binary

Binary information

☑ binaryinfo.observed
☑ binaryinfo.host.observed
☑ binaryinfo.group.observed

Binary upload

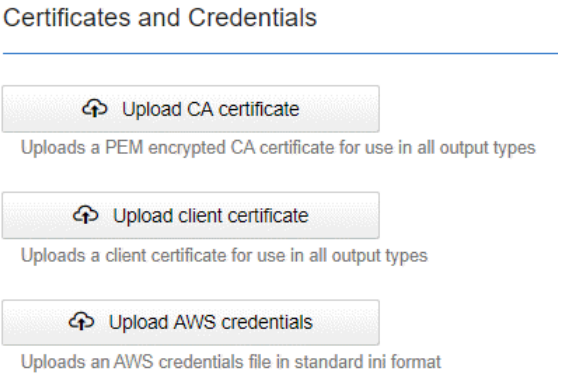
☑ binarystore.file.added

Audit logging

☑ audit.log.*

5 Optionally upload certificates and AWS credentials to validate connections.

a In the **Output** section, click the button for the certificate or credential to upload.



Certificates and Credentials

Upload CA certificate

Uploads a PEM encrypted CA certificate for use in all output types

Upload client certificate

Uploads a client certificate for use in all output types

Upload AWS credentials

Uploads an AWS credentials file in standard ini format

b In the dialog, specify the file to upload. Click **Upload**.

6 Click **Save**.

7 Stop and restart the service.

Note You can click **Cancel** to revert to the previous settings.

Event Forwarder Splunk Output

The Event Forwarder Splunk destination type requires the following information.

Parameter	Description
Splunk HEC Endpoint	Required URL of the Splunk destination endpoint; for example, `http://www.example.com`.
HEC Token	Required token for HEC authorization.
Server Common Name	Optional. Common name (CN) of the destination server.

Parameter	Description
Send Timeout	Optional. Maximum duration of an upload connection. The default value is 60 seconds.
Upload Empty Files	Optional. Determines whether zero byte length files are uploaded. The default setting is false.
Max Bundle Size	Optional. The maximum bundle size (in bytes) to upload to the remote destination before compression is applied. The default value is 10MB.
Certificates and Credentials	Optional. Determines whether an uploaded certificate (identified by type) is required for the connection. Also allows you to specify AWS credentials.

Event Forwarder S3 Output

The Event Forwarder S3 destination type requires the following information.

Parameter	Description
S3 Bucket	Required. The name of the S3 bucket to receive the output. The format must be `[<region>:]<bucket-name>` .
Send Timeout	Optional. Maximum duration of an upload connection. The default value is 60 seconds.
Upload Empty Files	Optional. Determines whether zero byte length files are uploaded. The default setting is false.
Max Bundle Size	Optional. The maximum bundle size (in bytes) to upload to the remote destination before compression is applied. The default value is 10MB.
Server-Side Encryption	Optional. Identifies the type of encryption that is required on the server. The default type is AES256.
ACL Policy	Required. Settings can be READ, WRITE, READ_ACP, WRITE_ACP, or FULL_CONTROL. These are typical permissions: the _ACP permissions also allow read/write to the ACL of the bucket itself. See Access Control List (ACL) Overview . We recommend that you use the default policy, which is “bucket-owner-full-control".
Credential Profile	Required. The profile name that is used to connect to S3 as defined in the **Certificates and Credentials** section of this page.
Dual Stack Networking	Optional. If unchecked, this setting indicates that this connection will consist of IPv4 addresses only. Enable this setting for IPv6 connections only.
Object Prefix	Optional. Embedded identifier that is useful for multiple Event Forwarders. The object prefix helps distinguish between output from multiple Event Forwarder instances, each of which has a distinct prefix.
Use Compression	Optional. If enabled, the payload is compressed.

Event Forwarder HTTP Output

The Event Forwarder HTTP destination type requires the following information.

Parameter	Description
Host Address	Required. URL of the HTTP destination endpoint.
Server Common Name	Optional. Common name (CN) of the destination server.

Parameter	Description
Send Timeout	Optional. Maximum duration of an upload connection. The default value is 60 seconds.
Upload Empty Files	Optional. Determines whether zero byte length files are uploaded. The default setting is false.
Max Bundle Size	Optional. The maximum bundle size (in bytes) to upload to the remote destination before compression is applied. The default value is 10MB.
Upload Template	Required. Template for formatting the output messages. The required template format to enter in this field is: `{"filename": "{{.FileName}}", "service": "carbonblack", "alerts":[{{range .Events}}{{.EventText}}{{end}}]}`
Content-type HTTP Header	Required. The HTTP header helps the HTTP client consume the output properly. For example: `application/json, application/text, application/xml` . We recommend that you use the default value for this setting.
HTTP Authorization Token	Optional. Token to communicate with the remote destination.
OAUTH JWT - Client Email	Required if using OAUTH authentication. Oauth client identifier for communicating with the configured OAuth provider.
OAUTH JWT - Private Key	Required if using OAUTH authentication. PEM-encoded private key to sign the JWT payloads.
OAUTH JWT - Token URL	Required if using OAUTH authentication. The endpoint that is required to complete the 2-legged JWT flow.
OAUTH JWT - Private Key ID	Optional. A hint that indicates which key is being used.
OAUTH JWT - Permission Scopes	Optional. A comma-delimited list of requested permission scopes.
Send Events as Binary	Optional. If enabled, event JSON is sent in a byte array field instead of plain text.
Use Compression	Optional. If enabled, compresses the HTTP payload before uploading.
Certificates	Optional. Determines whether an uploaded certificate is required for the connection.

Event Forwarder Syslog Output

The Event Forwarder Syslog destination type requires the following information.

Parameter	Description
Syslog Destination	Required. The format is: `[{{protocol]:<fqdn>[:<port>]}}`
Server Common Name	Optional. Common name (CN) of the destination server.
Certificates	Optional. Determines whether an uploaded certificate is required for the connection.

YARA Manager

17

The Carbon Black Yara Manager provides a web-based user interface that is integrated with the Carbon Black EDR server to configure, control, and assess the status of the YARA Connector.

Yara Manager allows you to perform the following administrative actions on the Carbon Black YARA Connector that is installed on the Carbon Black EDR server:

- Get current status of the YARA Connector
- Restart the YARA Connector
- Delete all threat reports
- Upload new YARA Rules
- View the YARA Connector configuration

Important YARA Connector must be installed for YARA Manager to work properly. If YARA Manager is enabled through `cb.conf` on a system where YARA Connector is not installed, the user interface presents an HTML 404 error code. See https://github.com/carbonblack/cb-yara-connector for instructions on installing the YARA Connector. See the *Carbon Black EDR Server Configuration Guide* for information about `cb.conf`.

Read the following topics next:

- Install YARA Manager
- Controlling the YARA Manager Service
- Using the YARA Manager

Install YARA Manager

Perform the following procedure to install YARA Manager.

Procedure

1 Install the CbOpenSource repository:

```
cd /etc/yum.repos.d
curl -O https://opensource.carbonblack.com/release/x86_64/CbOpenSource.repo
```

2 Use Yum to install YARA Manager:

```
yum install python-cb-yara-manager
```

3 Copy the example config file:

```
cp /etc/cb/integrations/cb-yara-manager/config.py.example /etc/cb/integrations/cb-yara-
manager/config.py
```

4 Verify the config file.

5 Create an authentication configuration file at `/etc/cb/integrations/cb-yara-manager/auth.conf` . Add the following lines to the file:

```
[auth]
api_token=adequately_long_and_complex_password_or_token
```

6 Replace *adequately_long_and_complex_password_or_token* with a passphrase or token.

7 Start the YARA Manager.

8 Add the configured API token and YARA Management support to `cb.conf` :

```
YaraManagerEnabled=true
YaraManagerToken=adequately_long_and_complex_password
```

For information about `cb.conf`, see the *Carbon Black EDR Server Configuration Guide*.

9 Restart cb-coreservices to apply the changes:

```
 /usr/share/cb/cbservice cb-coreservices restart
```

10 Log in to your Carbon Black EDR console and browse to `https://<cb_server_url>/connectors/yara`, or click **YARA Manager** on the navigation bar.

Controlling the YARA Manager Service

This topic provides commands you can use to control the YARA Manager service.

CentOS / Red Hat 6

Action	Command
Start the service	`service cb-yara-manager start`
Stop the service	`service cb-yara-manager stop`
Display service status	`service cb-yara-manager status`

CentOS / Red Hat 7

Action	Command
Start the service	`systemctl start cb-yara-manager`
Stop the service	`systemctl stop cb-yara-manager`
Display service status	`systemctl status -l cb-yara-manager`
Display verbose logs	`journalctl -u cb-yara-manager`

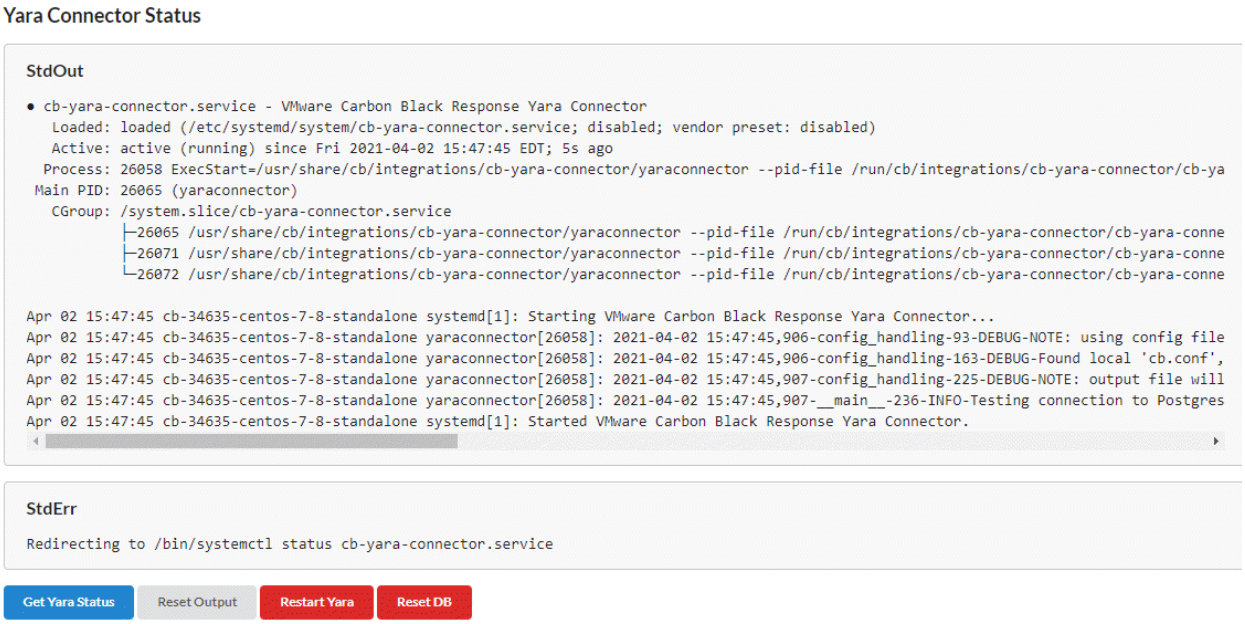
Using the YARA Manager

This topic describes how to use the YARA Manager.

- Log in to your Carbon Black EDR console and browse to `https://` *<cb_server_url>* / `connectors/yara` , or click **YARA Manager** on the navigation bar.

YARA Status

The YARA Status page displays YARA Connector status information. The output is taken directly from the Linux service command.



Yara Connector Status

StdOut

```
● cb-yara-connector.service - VMware Carbon Black Response Yara Connector
   Loaded: loaded (/etc/systemd/system/cb-yara-connector.service; disabled; vendor preset: disabled)
   Active: active (running) since Fri 2021-04-02 15:47:45 EDT; 5s ago
  Process: 26058 ExecStart=/usr/share/cb/integrations/cb-yara-connector/yaraconnector --pid-file /run/cb/integrations/cb-yara-connector/cb-ya
 Main PID: 26065 (yaraconnector)
   CGroup: /system.slice/cb-yara-connector.service
           ├─26065 /usr/share/cb/integrations/cb-yara-connector/yaraconnector --pid-file /run/cb/integrations/cb-yara-connector/cb-yara-conne
           ├─26071 /usr/share/cb/integrations/cb-yara-connector/yaraconnector --pid-file /run/cb/integrations/cb-yara-connector/cb-yara-conne
           └─26072 /usr/share/cb/integrations/cb-yara-connector/yaraconnector --pid-file /run/cb/integrations/cb-yara-connector/cb-yara-conne

Apr 02 15:47:45 cb-34635-centos-7-8-standalone systemd[1]: Starting VMware Carbon Black Response Yara Connector...
Apr 02 15:47:45 cb-34635-centos-7-8-standalone yaraconnector[26058]: 2021-04-02 15:47:45,906-config_handling-93-DEBUG-NOTE: using config file
Apr 02 15:47:45 cb-34635-centos-7-8-standalone yaraconnector[26058]: 2021-04-02 15:47:45,906-config_handling-163-DEBUG-Found local 'cb.conf',
Apr 02 15:47:45 cb-34635-centos-7-8-standalone yaraconnector[26058]: 2021-04-02 15:47:45,907-config_handling-225-DEBUG-NOTE: output file will
Apr 02 15:47:45 cb-34635-centos-7-8-standalone yaraconnector[26058]: 2021-04-02 15:47:45,907-__main__-236-INFO-Testing connection to Postgres
Apr 02 15:47:45 cb-34635-centos-7-8-standalone systemd[1]: Started VMware Carbon Black Response Yara Connector.
```

StdErr

```
Redirecting to /bin/systemctl status cb-yara-connector.service
```

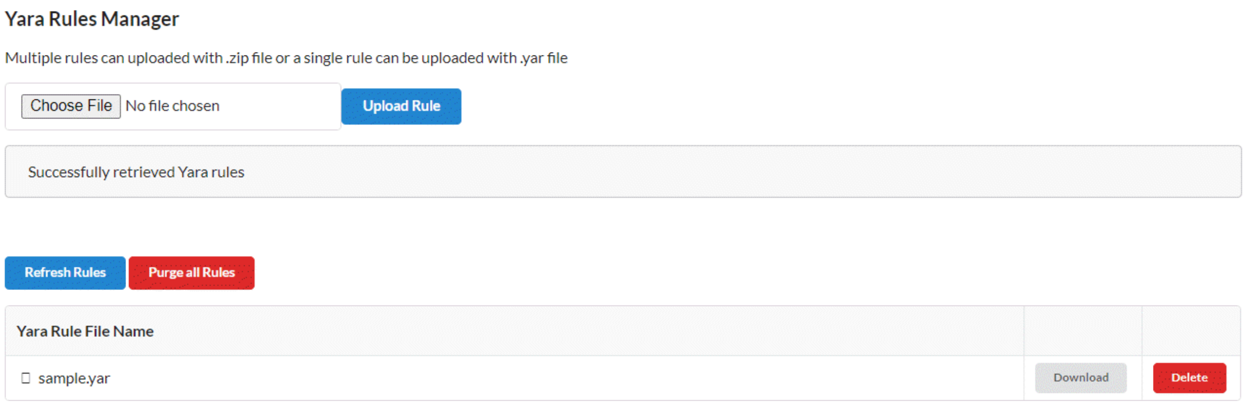
You can perform the following actions on this page:

- **Get YARA Status** — Retrieves the current status of the YARA connector and displays the results in the **StdOut** and **StdErr** text boxes.
- **Reset Output** — Resets the output.
- **Restart YARA** — Restarts the YARA connector.

- **Reset DB** — Resets the threat reports database to its empty state. This is typically used after adding YARA rules.

YARA Rules Manager

On the YARA Rules Manager page, you can upload, delete, and download YARA rule files.



Yara Rules Manager

Multiple rules can uploaded with .zip file or a single rule can be uploaded with .yar file

Successfully retrieved Yara rules

Yara Rule File Name		
sample.yar	Download	Delete

To upload a new YARA rule, click the **Choose File** button, select the appropriate `.yar` file, and click the **Upload Rule** button.

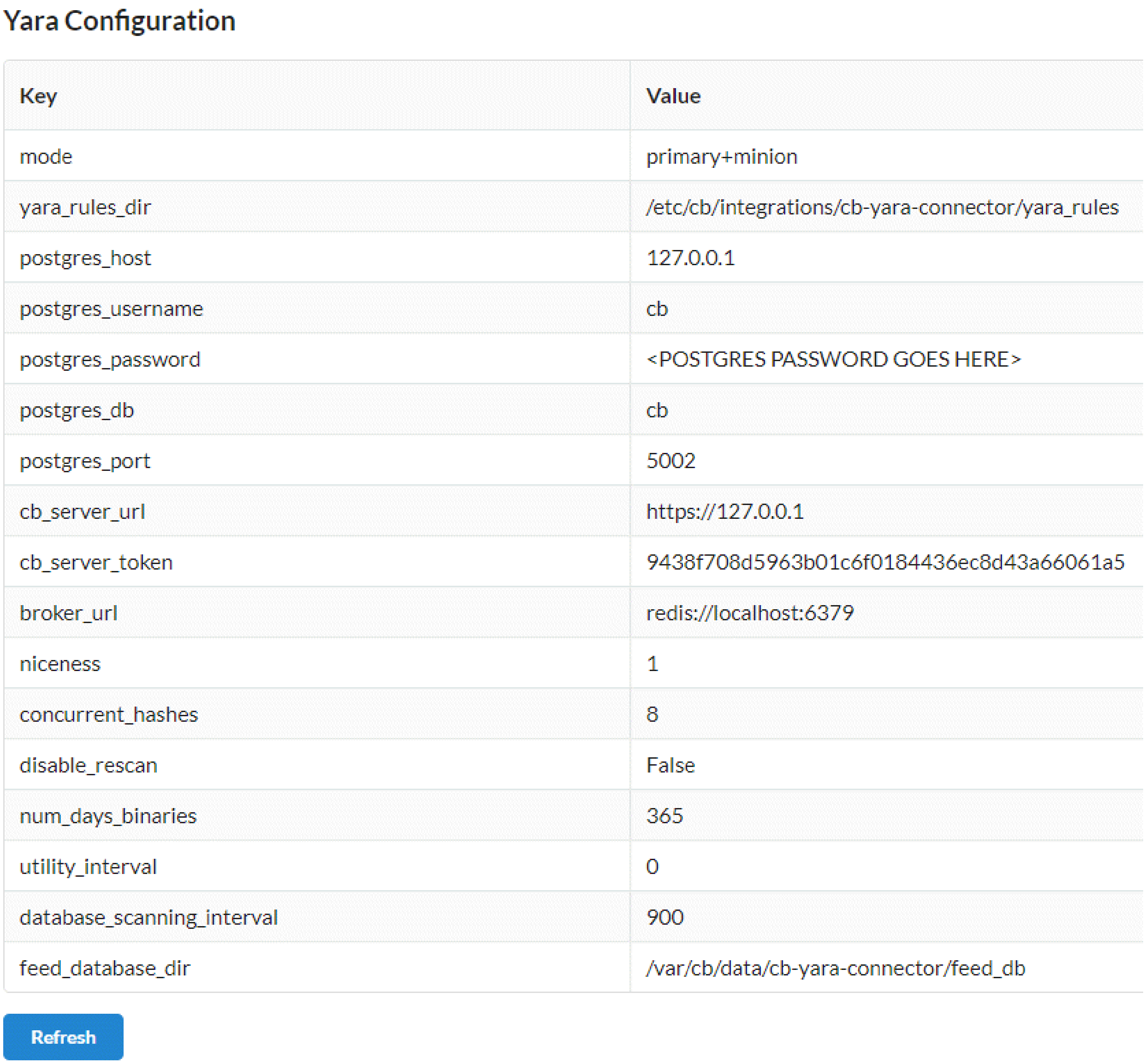
The YARA Manager supports the upload of multiple YARA rules. You can upload a zip file that contains multiple YARA rules. The YARA Manager extracts the zip file and puts all the rules in the path that the YARA connector configuration file specifies.

To delete all YARA rules click the **Purge all Rules** button. Alternatively, you can individually download or delete YARA rules.

YARA Configuration

The YARA Configuration page displays the current configuration of the YARA connector.

This information is gathered from the YARA connector's configuration file. You cannot edit this page; you can only make changes through the `connector.conf` file.

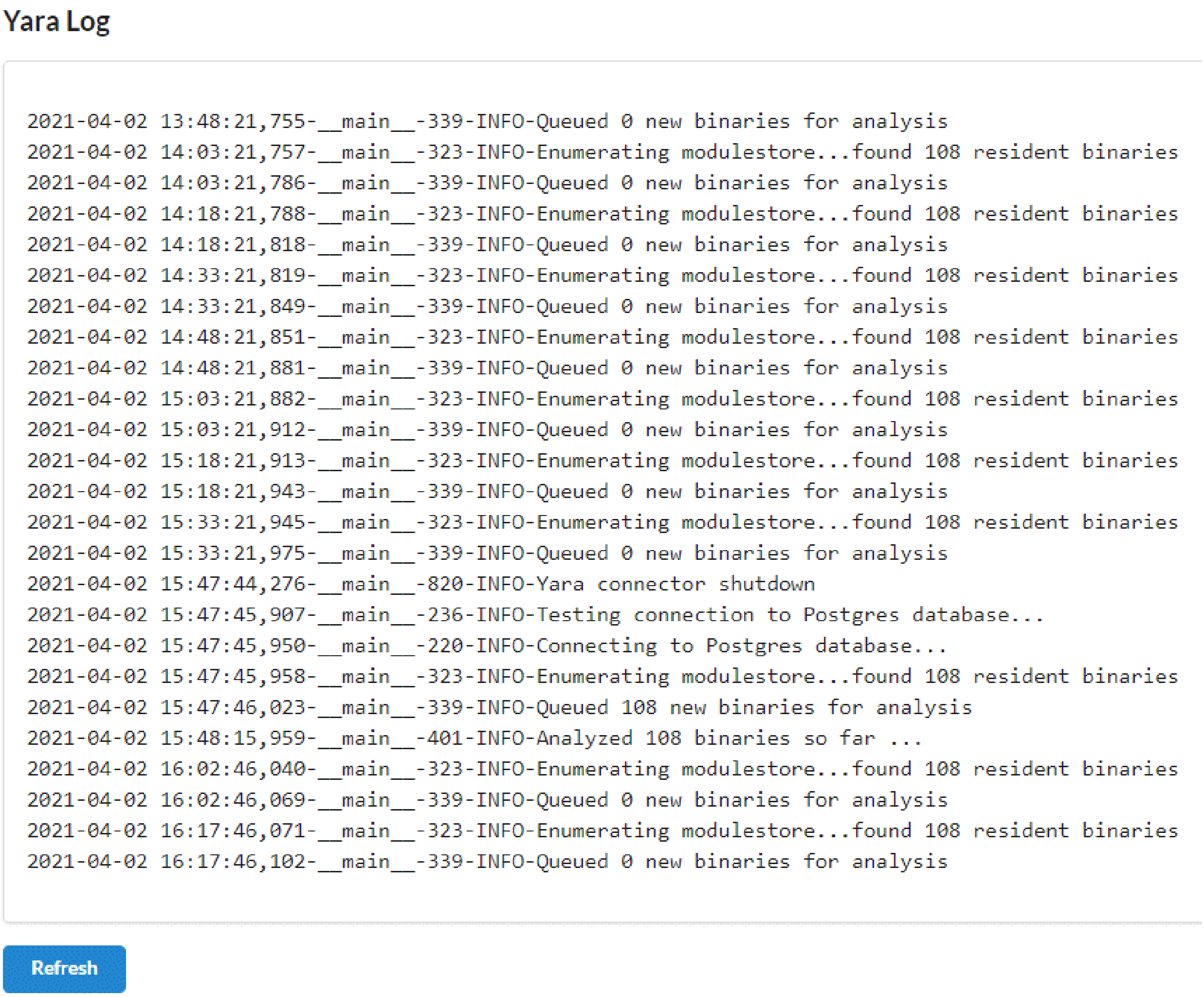


Yara Configuration

Key	Value
mode	primary+minion
yara_rules_dir	/etc/cb/integrations/cb-yara-connector/yara_rules
postgres_host	127.0.0.1
postgres_username	cb
postgres_password	<POSTGRES PASSWORD GOES HERE>
postgres_db	cb
postgres_port	5002
cb_server_url	https://127.0.0.1
cb_server_token	9438f708d5963b01c6f0184436ec8d43a66061a5
broker_url	redis://localhost:6379
niceness	1
concurrent_hashes	8
disable_rescan	False
num_days_binaries	365
utility_interval	0
database_scanning_interval	900
feed_database_dir	/var/cb/data/cb-yara-connector/feed_db

YARA Log

The YARA Log page displays the contents of the `/var/log/cb/integrations/yara-manager/debug.log` file.



Yara Log

```
2021-04-02 13:48:21,755-__main__-339-INFO-Queued 0 new binaries for analysis
2021-04-02 14:03:21,757-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 14:03:21,786-__main__-339-INFO-Queued 0 new binaries for analysis
2021-04-02 14:18:21,788-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 14:18:21,818-__main__-339-INFO-Queued 0 new binaries for analysis
2021-04-02 14:33:21,819-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 14:33:21,849-__main__-339-INFO-Queued 0 new binaries for analysis
2021-04-02 14:48:21,851-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 14:48:21,881-__main__-339-INFO-Queued 0 new binaries for analysis
2021-04-02 15:03:21,882-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 15:03:21,912-__main__-339-INFO-Queued 0 new binaries for analysis
2021-04-02 15:18:21,913-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 15:18:21,943-__main__-339-INFO-Queued 0 new binaries for analysis
2021-04-02 15:33:21,945-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 15:33:21,975-__main__-339-INFO-Queued 0 new binaries for analysis
2021-04-02 15:47:44,276-__main__-820-INFO-Yara connector shutdown
2021-04-02 15:47:45,907-__main__-236-INFO-Testing connection to Postgres database...
2021-04-02 15:47:45,950-__main__-220-INFO-Connecting to Postgres database...
2021-04-02 15:47:45,958-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 15:47:46,023-__main__-339-INFO-Queued 108 new binaries for analysis
2021-04-02 15:48:15,959-__main__-401-INFO-Analyzed 108 binaries so far ...
2021-04-02 16:02:46,040-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 16:02:46,069-__main__-339-INFO-Queued 0 new binaries for analysis
2021-04-02 16:17:46,071-__main__-323-INFO-Enumerating modulestore...found 108 resident binaries
2021-04-02 16:17:46,102-__main__-339-INFO-Queued 0 new binaries for analysis
```

Investigations

18

Investigations allow you to group data for reporting, compliance, or retention purposes. This section describes how to work with investigations.

Investigations are collections of process events that share a common focus. They can include details and notes, and provide a way to group data for reporting, compliance, or retention purposes. Investigations are not particular to any user, so all investigations are available to each Carbon Black EDR administrator.

It is a best practice to start an investigation whenever you begin any type of search — for example, after you discover a suspicious indicator and start searching for related process activity on your hosts.

You can create an investigation to keep an ongoing record of the scope and effects of the threat, so that you can stop it before it causes damage. There is no cost involved in creating an investigation, and if you tag process events during your search, you have a built-in record of the steps that provided the end result.

A default investigation comes with the Carbon Black EDR server installation and is always available to collect any data that you tag. The default investigation cannot be deleted, so it is best used as a repository for data that interests you but does not warrant a dedicated investigation of its own.

The first time that you open the Investigations page, the default investigation appears.

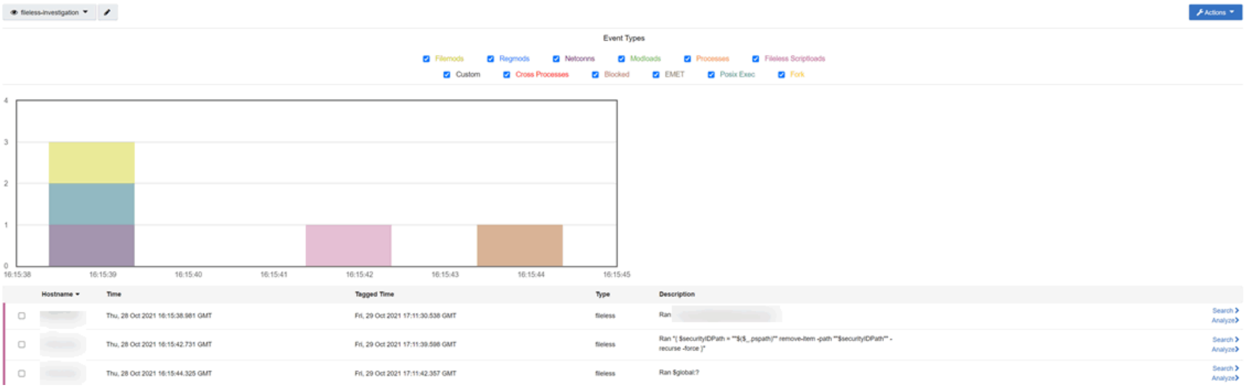
Read the following topics next:

- Viewing Investigations
- Create an Investigation
- Add an Event to an Investigation
- Add a Custom Event to an Investigation
- Remove an Event from an Investigation
- Delete an Investigation

Viewing Investigations

This section describes the Investigations page.

- On the navigation bar, click **Investigations**.



	Hostname	Time	Tagged Time	Type	Description	
□	[redacted]	Thu, 28 Oct 2021 16:15:38.981 GMT	Fri, 29 Oct 2021 17:11:30.538 GMT	fileless	Ran [redacted]	Search › Analyze›
□	[redacted]	Thu, 28 Oct 2021 16:15:42.731 GMT	Fri, 29 Oct 2021 17:11:39.598 GMT	fileless	Ran "($securityIDPath = "\"$($_.pspath)\"" remove-item -path "\"$securityIDPath\"" -recurse -force)"	Search › Analyze›
□	[redacted]	Thu, 28 Oct 2021 16:15:44.325 GMT	Fri, 29 Oct 2021 17:11:42.357 GMT	fileless	Ran $global:?	Search › Analyze›

Investigations Page Menu Bar

The Investagations page menu bar contains a drop-down list of available investigations and an **Actions** menu.

The **Actions** menu contains the following options:

- **Remove Events** – See Remove an Event from an Investigation.
- **Add Custom Event** – See Add a Custom Event to an Investigation.
- **Add Investigation** – See Create an Investigation.
- **Delete Investigation** – See Delete an Investigation.
- **Export timeline to PNG** – Exports data from the graph to a `.png` file and downloads it to your computer.
- **Export events to CSV** – Exports data to a `.csv` file and downloads it to your computer.

Investigations Page Event Types

Select and deselect checkboxes next to event types to sort the events that display in the timeline and table. Only selected events will appear.

The following event types appear:

- **Filemods** – The number of files that were modified by process executions. Color-coded as yellow.
- **Regmods** – The number of Windows registry modifications that were made by processes executions. Color-coded as blue.
- **Netconns** – The number of network connections that process executions either attempted or established. Color-coded as purple.

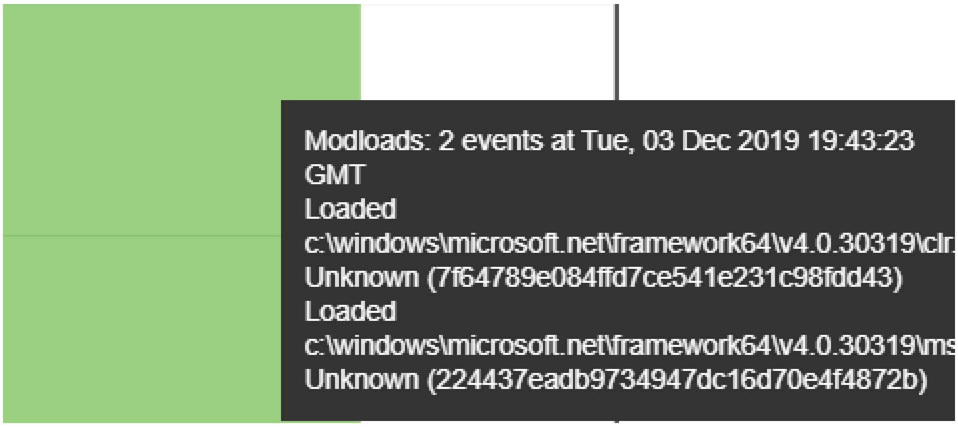
- **Modloads** – The number of modules that were loaded by process executions. Color-coded as green.
- **Processes/Child Processes** – The number of child processes that were generated from process executions. Color-coded as orange.
- **Fileless Scriptloads** – The fileless_scriptload event represents each occasion when the sensor detected PowerShell script content that was executed by any process on a supported endpoint.
- **Custom** – A custom event that you can create using the **Add Custom Event** option in the **Actions** menu. Color-coded as black.
- **Cross Processes** – (Windows only) A process that crosses the security boundary of another process. Color-coded as red.
- **Blocked** – Represents events that are related to the Ban Hash functionality (see Banning Process Hashes). If an administrator bans a hash and the sensor sees that binary and tries to stop it (already running) or prohibits it from running (blocks it), then the sensor generates a Blocked event. Color-coded as brown.
- **EMET** – Represents a specific type of event that deals with the Microsoft Enhanced Mitigation Experience Toolkit (EMET) software. Color-coded as gray.
- **Posix_Exec** – (macOS and Linux only) The instance’s process that is loaded and the new binary image. Color-coded as green.
- **Fork** – (macOS and Linux only) The instance’s parent process, forked with a different Process ID (PID). Color-coded as yellow orange.

Investigations Page Bar Graph

The bar graph contains a timeline of the events that are tagged for the investigation.

The events appear in color-coded bars (according to the event types). Events are stacked when they occur at the same time.

The color coding indicates which events happen at which times. Hovering over the color indicators on the timeline produces pop-up text, which explains what the block of color represents. For example:



Modloads: 2 events at Tue, 03 Dec 2019 19:43:23 GMT
Loaded c:\windows\microsoft.net\framework64\v4.0.30319\clr.
Unknown (7f64789e084ffd7ce541e231c98fdd43)
Loaded c:\windows\microsoft.net\framework64\v4.0.30319\ms
Unknown (224437eadb9734947dc16d70e4f4872b)

Investigations Page Events Table

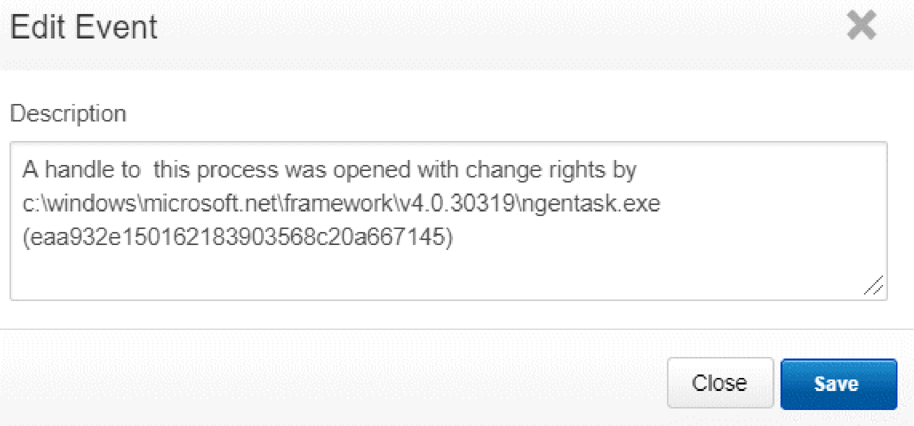
The **Events** table shows the events that are contained in investigations. A colored bar on the left border of each row indicates the event type.

Column	Description
Hostname	The name of the host on which the event occurred.
Time	The date and time that the event occurred.
Tagged Time	The time that the event was tagged for this investigation.
Type	The event type (filemod, regmod, netconn, modload, child process, fork, posix_exec, custom, crossproc, blocked, EMET).
Description	Description of the event; for example, paths to files and registry elements that were modified, signature status, and hash values.
Search	Opens the event in the Process Search page. See Overview of Process Search.
Analyze	Opens the event in the Process Analysis page. See Process Analysis Page.

Investigations Page Event Description

Each event on the Investigations page includes an editable description.

When you hover over the description in a row, an **Edit** icon appears. Click the **Edit** icon to open the **Edit Event** window:



A handle to this process was opened with change rights by c:\windows\microsoft.net\framework\v4.0.30319\ngentask.exe (eaa932e150162183903568c20a667145)

Use the **Edit Event** window to add context to the technical description or insights to share with the rest of your investigative team. Edits are visible within the investigation, but do not appear in the process execution data when viewed outside the Investigations page.

Investigation Page Child Processes

Rows that represent child processes contain a **Search** (magnifying glass) icon. This option displays a preview of the Process Analysis page for the child process.

See Process Analysis Page.

Create an Investigation

You can create customized investigations and then add events to those investigations.

Procedure

1 On the navigation bar, click **Investigations**.

The Default investigation displays.

2 Click **Actions > Add Investigation**.

The **Add Investigation** window displays.

3 Enter a name for the investigation and click **Save**.

Note The name must be alpha-numeric. Special characters are not allowed.

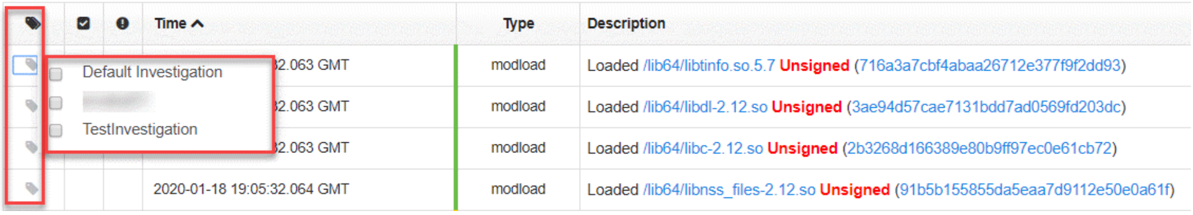
The new investigation displays in the **Investigations** window, but it is empty until you add events to it. See Add an Event to an Investigation.

Add an Event to an Investigation

Perform the following procedure to add events to an investigation.

Procedure

1. On the navigation bar, click **Process Search**.
2. Select a process from the Process Search page (see Overview of Process Search) and open the Process Analysis page (see Process Analysis Page).
3. Scroll to the list of events. Click the **Tag** icon in an event row and select an investigation.



□ Default Investigation
□ [redacted]
□ TestInvestigation

	☑	●	Time ^	Type	Description
			2.063 GMT	modload	Loaded /lib64/libtinfo.so.5.7 **Unsigned** (716a3a7cbf4abaa26712e377f9f2dd93)
			2.063 GMT	modload	Loaded /lib64/libdl-2.12.so **Unsigned** (3ae94d57cae7131bdd7ad0569fd203dc)
			2.063 GMT	modload	Loaded /lib64/libc-2.12.so **Unsigned** (2b3268d166389e80b9ff97ec0e61cb72)
			2020-01-18 19:05:32.064 GMT	modload	Loaded /lib64/libnss_files-2.12.so **Unsigned** (91b5b155855da5eaa7d9112e50e0a61f)

Add a Custom Event to an Investigation

Perform the following procedure to add a custom event to an investigation.

You can create a custom event that you can use to:

- Add a new event type to the system.
- Add a note that displays on its own line in the rows at the bottom of the Investigations page.

Procedure

1. On the navigation bar, click **Investigations**.
2. Click **Actions** > **Add Custom Event**.
3. In the **Description** field, type a description for the event.
4. In the **Start Time** field, enter the date and time for the event.
5. Click **Save**.

Remove an Event from an Investigation

When you remove an event from an investigation, it continues to exist in the system but is no longer included in the investigation.

Procedure

1. On the navigation bar, click **Investigations**.

   The Default investigation displays.

2. Click the **Investigation** drop-down menu in the top-left corner and open an investigation.
3. Click **Actions** > **Remove Events**.

Delete an Investigation

When you delete an investigation, only the grouping, tagging, and edited descriptions are deleted. The deletion has no other effect on the process executions that were a part of the investigation, or on how those processes display in other pages.

Procedure

1. On the navigation bar, click **Investigations**.
2. Select the investigation to delete.
3. Click **Actions > Delete Investigation**.
4. Click **OK** to confirm the deletion.

Watchlists

19

Watchlists are saved searches that run periodically against the process or binary data in Carbon Black EDR. Watchlists are visible to all users.

Watchlists are named process or binary searches that the server runs periodically (approximately every 10 minutes) without user action. When those saved searches produce new results, the server notifies users about them in a configurable way.

First responders can use the Watchlists page to quickly see items that are potentially interesting. For example, the **Newly Executed Applications** watchlist gives you rapid access to a list of the latest applications that were executed. If known recent issues occur with any new applications, you can quickly scan the results of that watchlist to find potential problems.

For watchlists that are based on threat intelligence feeds, you can factor scoring into a saved search. These watchlists tag a process or binary that is found on one of your endpoints when the score from a specified feed matches a specified score or falls within a specified score range. The score is the rating that is used to calculate the severity that is assigned to IOCs from a feed.

The severity calculation for alerts uses the following inputs:

- Feed rating
- Report score
- Confidence
- Criticality

For watchlist alerts, the first three values are constants; only the criticality varies, based on the sensor group (which defaults to 3):

- Feed rating = 3
- Report score = 75
- Confidence = 0.5

Using the default values, the severity is always 51.

The listed report score is not the score of the report that triggered the alert; instead, it is the score of the watchlist. The details for the watchlist alert on the Watchlists page shows the *watchlist* report score, not the *report's* report score. For example, if criticality is set to 5, the calculated severity is 61. If you perform a query requesting IOCs that have a watchlist score of over 80, the generated report shows all IOCs that have a severity over 61 even though the watchlist data is 80 and above.

Additional information about enabling and using watchlists in specific contexts displays in the following pages:

- Chapter 8 Responding to Endpoint Incidents
- Chapter 10 Process Search and Analysis
- Chapter 11 Binary Search and Analysis
- Chapter 14 Threat Intelligence Feeds
- Chapter 20 Console and Email Alerts

Read the following topics next:

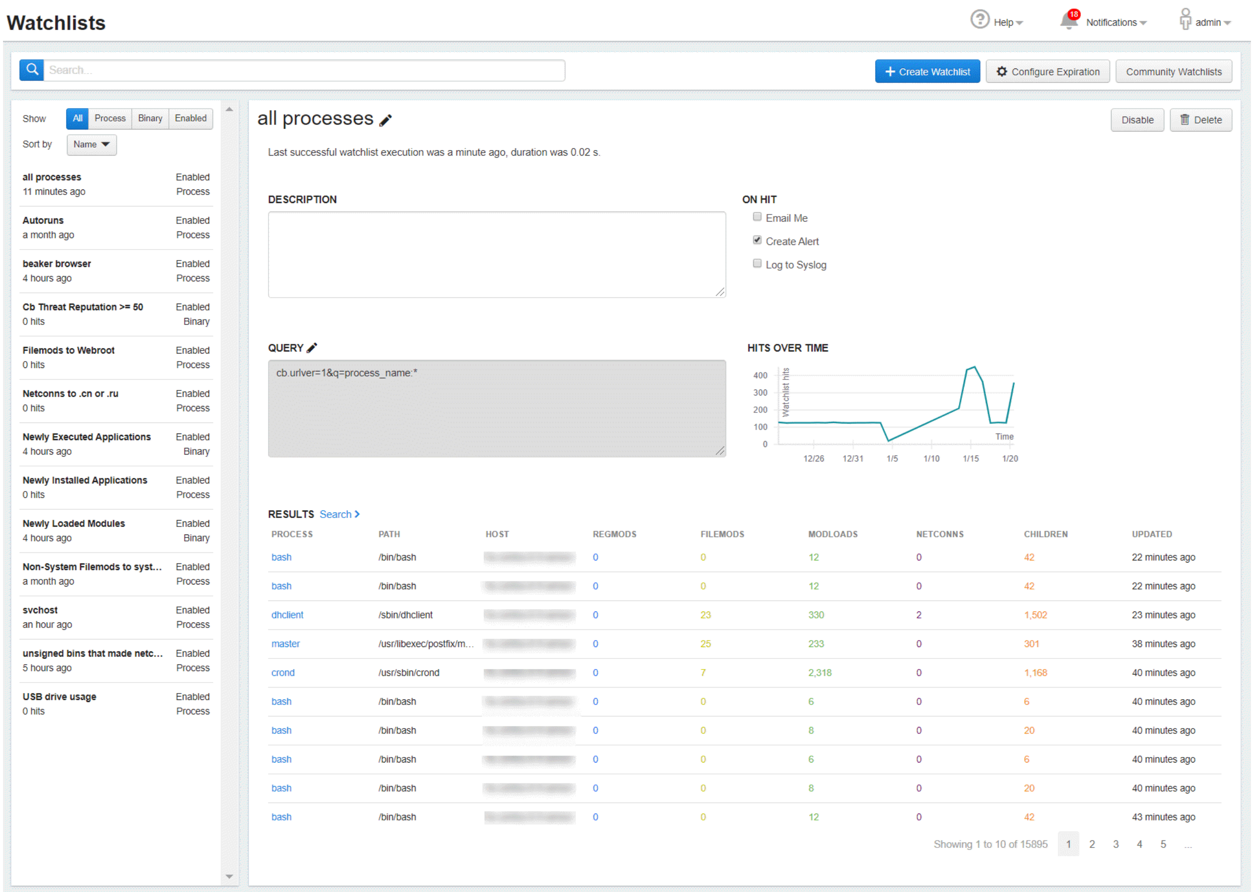
- Viewing Watchlists
- Built-in and Community Watchlists
- Creating Watchlists
- Managing Watchlists

Viewing Watchlists

This topic describes the Watchlists page in the Carbon Black EDR console.

- On the navigation bar, click **Watchlists**.

On the Watchlists page, names of existing watchlists appear in a table on the left. Details and results for one watchlist (by default, the first one in the table) appear on the right. You can display the details and results of a different watchlist by clicking its name.



Show: All | Process | Binary | Enabled
Sort by: Name

all processes 11 minutes ago	Enabled Process
Autoruns a month ago	Enabled Process
beaker browser 4 hours ago	Enabled Process
Cb Threat Reputation >= 50 0 hits	Enabled Binary
Filemods to Webroot 0 hits	Enabled Process
Netconns to .cn or .ru 0 hits	Enabled Process
Newly Executed Applications 4 hours ago	Enabled Binary
Newly Installed Applications 0 hits	Enabled Process
Newly Loaded Modules 4 hours ago	Enabled Binary
Non-System Filemods to syst... a month ago	Enabled Process
svchost an hour ago	Enabled Process
unsigned bins that made netc... 5 hours ago	Enabled Process
USB drive usage 0 hits	Enabled Process

all processes

Last successful watchlist execution was a minute ago, duration was 0.02 s.

DESCRIPTION

ON HIT
☐ Email Me
☑ Create Alert
☐ Log to Syslog

QUERY

```
cb.urlver=1&q=process_name:*
```

HITS OVER TIME

RESULTS Search >

PROCESS	PATH	HOST	REGMODS	FILEMODS	MODLOADS	NETCONNS	CHILDREN	UPDATED
bash	/bin/bash	[redacted]	0	0	12	0	42	22 minutes ago
bash	/bin/bash	[redacted]	0	0	12	0	42	22 minutes ago
dhclient	/sbin/dhclient	[redacted]	0	23	330	2	1,502	23 minutes ago
master	/usr/libexec/postfix/m...	[redacted]	0	25	233	0	301	38 minutes ago
crond	/usr/sbin/crond	[redacted]	0	7	2,318	0	1,168	40 minutes ago
bash	/bin/bash	[redacted]	0	0	6	0	6	40 minutes ago
bash	/bin/bash	[redacted]	0	0	8	0	20	40 minutes ago
bash	/bin/bash	[redacted]	0	0	6	0	6	40 minutes ago
bash	/bin/bash	[redacted]	0	0	8	0	20	40 minutes ago
bash	/bin/bash	[redacted]	0	0	12	0	42	43 minutes ago

Showing 1 to 10 of 15895 1 2 3 4 5 ...

The left panel on the Watchlists page shows all available watchlists, their status and type, the number of hits, and either the time of their last run or another status message if they have not run recently. There are two tools for filtering these watchlists:

- At the top of the Watchlists page, use the **Search** box to search for watchlists by name.
- Immediately above the table of Watchlists on the left, filters and sorting controls can modify what is shown in the table. In the **Show** field, you can chose to show all watchlists, process watchlists only, binary watchlists only, or enabled watchlists only. In the **Sort by** field, you can sort by name, by the time the watchlist was created, by duration (how long it took the query to run), or by when each watchlist was most recently triggered. See Managing Watchlists for how you can use these features to effectively manage watchlists.

Watchlist Details Panel

The **Watchlist Details** panel on the right shows details for the currently selected watchlist. It includes the following information:

- Name and Description (if provided) of the watchlist.
- If the most recent execution was successful, its time and duration. For unsuccessful executions, this line shows either timeout or error information. Typically watchlists are scheduled to run every 10 minutes, but if a previous watchlist session is still running, the next one will be delayed and try to start periodically (every 10 minutes).

- Query used to match events to the watchlist.
- The **On Hit** settings determine how (or if) you are notified when an event matches the query.
- A graph that shows the number of hits on this watchlist over time.
- Table of results showing details for each hit.

Note For each watchlist that is run, the number of matching events that are tagged is limited to 100, even if more events actually match the watchlist. This limit prevents performance issues and eliminates the potential for excessive numbers of notification emails that are unlikely to add useful information.

Click the **Search** link to show query results in the context of the Process Search page or the Binary Search page.

The Watchlist Details panel also provides buttons in the top right to disable or delete the watchlist. When you click the **Disable** button, the watchlist is disabled and no longer runs. New results that match the search query do not result in any notification or record that they triggered a hit for the watchlist.

When you click the **Delete** button and confirm the deletion, the watchlist is permanently removed.

Built-in and Community Watchlists

Carbon Black EDR provides access to two sources of pre-configured watchlists, in addition to the watchlists you create.

- The Carbon Black Communityprovides a forum for sharing watchlists.
- The Watchlists page in the console includes a list of default watchlists, which include the following:
  - Autoruns
  - Filemods to Webroot
  - Netconns to .cn or .ru
  - Newly Executed Applications
  - Newly Installed Applications
  - Newly Loaded Modules
  - Non-System Filemods to system
  - USB drive usage
  - Carbon Black Threat Reputation

In addition to the built-in Watchlists, in the top-right corner of the **Watchlists** page, you can click the **Community Watchlists** button to access Threat Research on the Carbon Black User Exchange. Threat Research is a central portal where watchlist users can publish and discuss watchlists that might eventually be included as a feed.

Creating Watchlists

You can create your own customized watchlists from the Watchlists, Process Search, Binary Search, or Threat Intelligence Feeds page. The information you provide varies depending on the location from where you start.

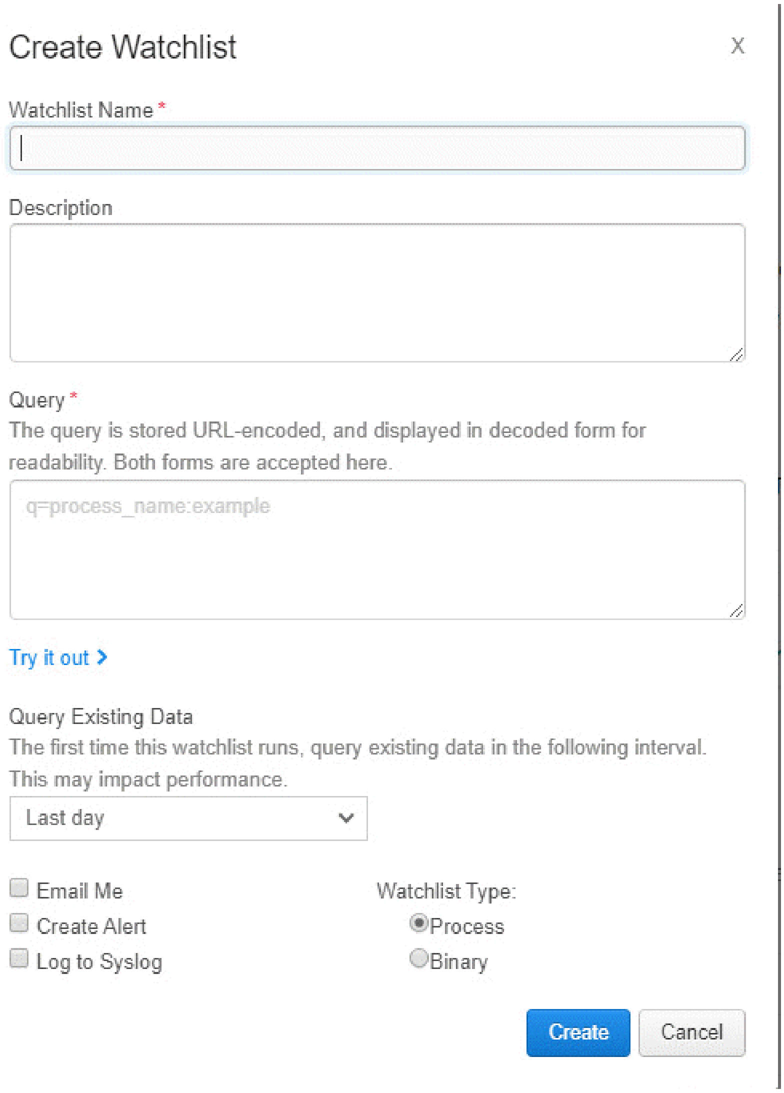
Create a Watchlist on the Watchlists Page

Perform the following procedure to create a watchlist on the Watchlists page in the Carbon Black EDR console.

Procedure

1 On the navigation bar, click **Watchlists**.

2 Click the **Create Watchlist** button.



Create Watchlist

Watchlist Name *

Description

Query *
The query is stored URL-encoded, and displayed in decoded form for readability. Both forms are accepted here.

q=process_name:example

Try it out >

Query Existing Data
The first time this watchlist runs, query existing data in the following interval. This may impact performance.

Last day

□ Email Me
□ Create Alert
□ Log to Syslog

Watchlist Type:
● Process
○ Binary

- **Watchlist Name:** Enter a meaningful name for the watchlist.
- **Description**: Provide the purpose of the watchlist (optional).
- **Query**: The query that is currently open, if any.
- **Query Existing Data**: Define the time period for which existing data is queried on the first run of the watchlist. The longer the timeframe that is selected, the longer it will take the query to run directly after this watchlist is created. A longer time can also stress other product services, such as process search, while the watchlist is running. After the watchlist has run one time, it will run on new data in 10 minute intervals thereafter.
- **Email Me**: Select the checkbox to receive email notifications for matching hits.
- **Create Alert**: Select the checkbox to send an alert when conditions matching the watchlist occur. Triggered alerts are reported in the Alert Dashboard page and the Triage Alerts page. For more information on alerts, see Chapter 20 Console and Email Alerts.

- **Log to Syslog**: Select the checkbox to log all hits `syslog`. Syslogs are written to `/var/log/cb/notifications/`. In this case, the log filenames have the format `cb-notifications-`*<watchlist ID>*`.log`.
- **Watchlist Type**: Identify the type as **Process** or **Binary**.

3 Click **Create**.

Create a Watchlist on the Process Search or Binary Search Page

Perform the following procedure to create a watchlist on the Process Search or Binary Search page in the Carbon Black EDR console.

Procedure

1 On the navigation bar, select either **Process Search** or **Binary Search**.

2 Enter the query for the processes or binaries from which to create a watchlist. The syntax should match a search box query in the Process or Binary Search page.

Caution Use of leading wildcards is discouraged because of performance issues.

You cannot edit several aspects of a watchlist search query, so examine the results carefully before proceeding. For more information on editing queries, see Edit a Watchlist.

For more information on performing searches, see:

- Chapter 10 Process Search and Analysis
- Chapter 11 Binary Search and Analysis
- Chapter 12 Advanced Search Queries

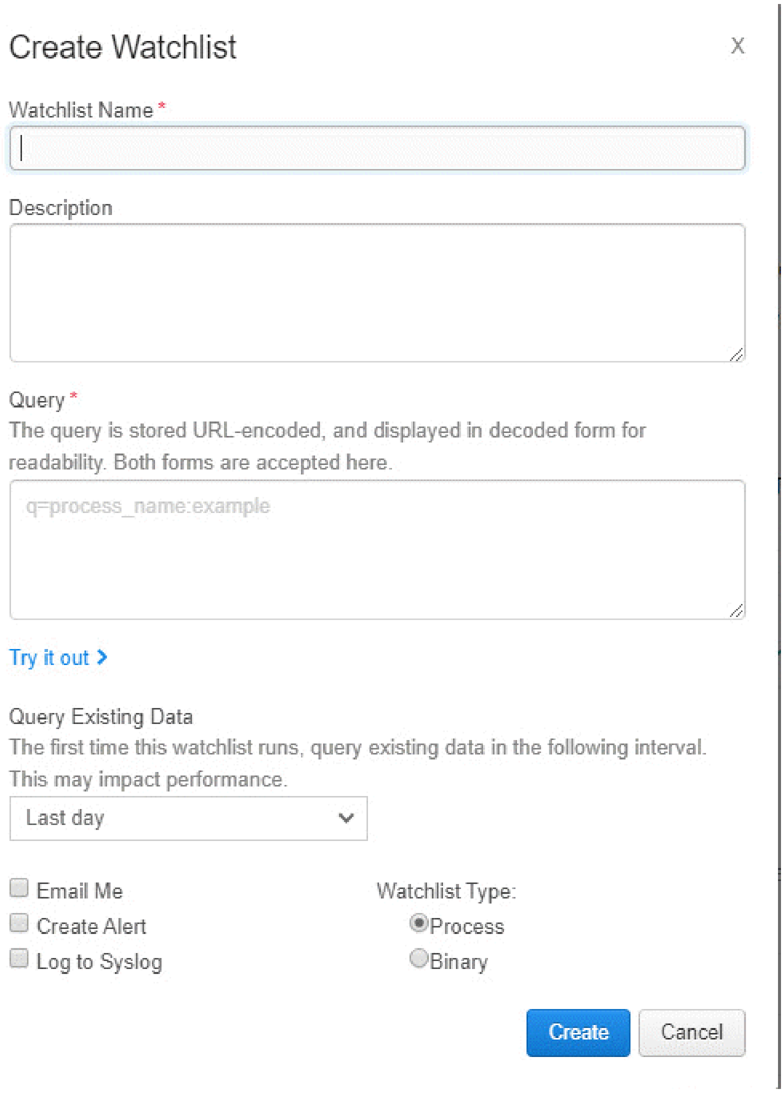
If you are using multiple MD5 or SHA-256 hash values for search criteria to create a watchlist, you must enclose the values in parentheses ().

For example:

```
(md5:45cc061d9581e52f008e90e81da2cfd9 md5:829e4805b0e12b383ee09abdc9e2dc3c
md5:ac9fa2ba34225342a8897930503ae12f md5:5f7eaaf5d10e2a715d5e305ac992b2a7)
```

If you do not enclose the list in parentheses, the only value that is tagged for the watchlist is the last value in the list.

3 On the Process Search page, click **Create Watchlist** or, on the Binary Search page, click **Create Watchlist** from the **Action** menu.



Create Watchlist

Watchlist Name *

Description

Query *

The query is stored URL-encoded, and displayed in decoded form for readability. Both forms are accepted here.

q=process_name:example

Try it out >

Query Existing Data

The first time this watchlist runs, query existing data in the following interval. This may impact performance.

Last day

□ Email Me

□ Create Alert

□ Log to Syslog

Watchlist Type:

● Process

○ Binary

- **Watchlist Name:** Enter a meaningful name for the watchlist.
- **Description**: Provide the purpose of the watchlist (optional).
- **Query**: The query that is currently open, if any.
- **Query Existing Data**: Define the time period for which existing data is queried on the first run of the watchlist. The longer the timeframe that is selected, the longer it will take the query to run directly after this watchlist is created. A longer time can also stress other product services, such as process search, while the watchlist is running. After the watchlist has run one time, it will run on new data in 10 minute intervals thereafter.
- **Email Me**: Select the checkbox to receive email notifications for matching hits.
- **Create Alert**: Select the checkbox to send an alert when conditions matching the watchlist occur. Triggered alerts are reported in the Alert Dashboard page and the Triage Alerts page. For more information on alerts, see Chapter 20 Console and Email Alerts.

- **Log to Syslog**: Select the checkbox to log all hits `syslog`. Syslogs are written to `/var/log/cb/notifications/` . In this case, the log filenames have the format `cb-notifications-`*<watchlist ID>*`.log`.
- **Watchlist Type**: Identify the type as **Process** or **Binary**.

4 Click **Create**.

Create a Watchlist on the Threat Intelligence Feeds Page

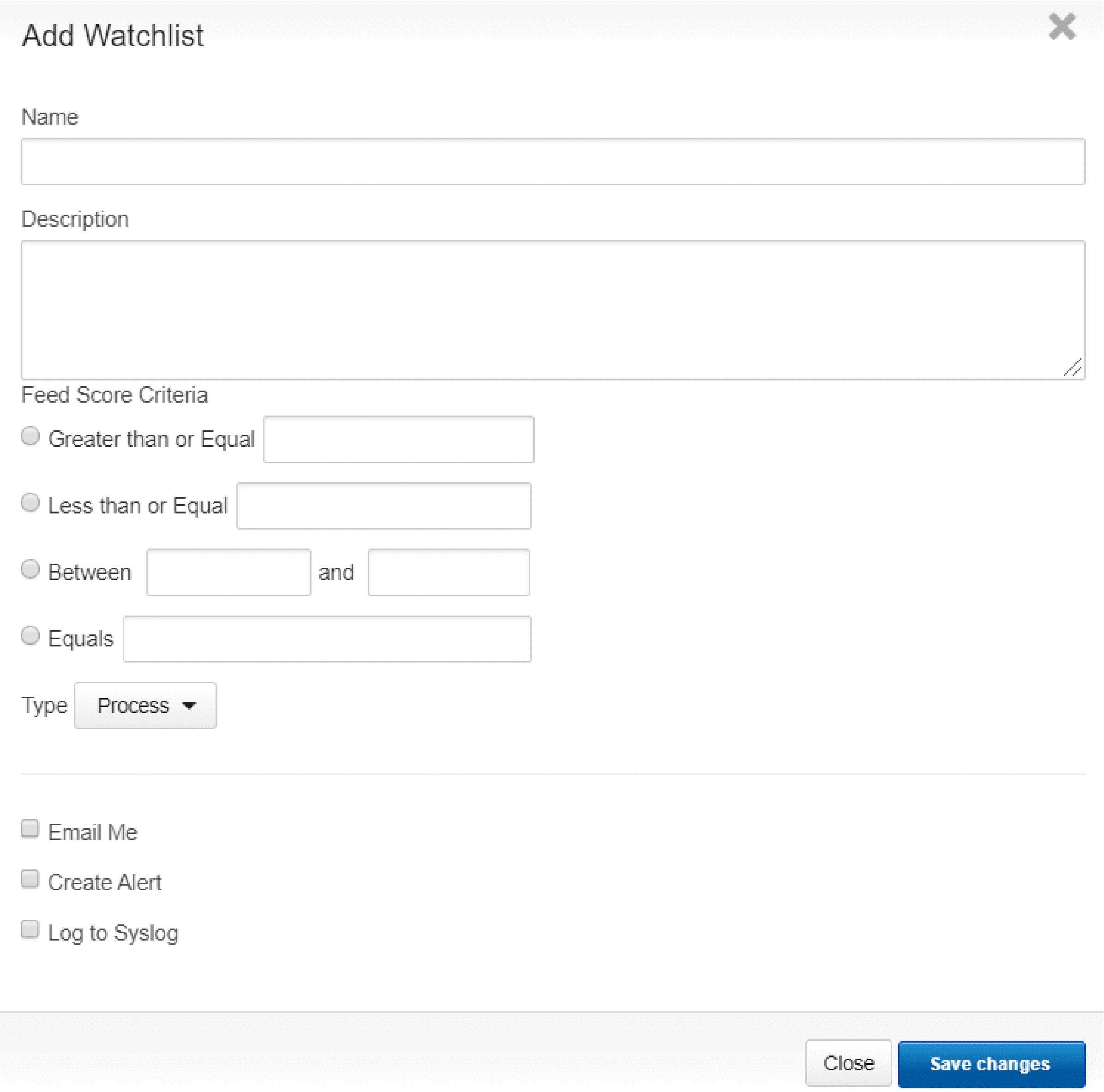
Perform the following procedure to create a watchlist on the Threat Intelligence Feeds page in the Carbon Black EDR console.

Procedure

1 On the navigation bar, click **Threat Intelligence**.

2 Select the feed for which to create a watchlist.

3 From the **Actions** menu, click **Create Watchlist**.



- **Watchlist Name:** Enter a meaningful name for the watchlist.
- **Description**: Provide the purpose of the watchlist (optional).
- In the **Feed Score Criteria** section, use the fields to enter the score criteria for the severity of IOCs to track.
- On the **Type** drop-down menu, click **Process** or **Binary**.
- **Email Me**: Select the checkbox to receive email notifications for matching hits.
- **Create Alert**: Select the checkbox to send an alert when conditions matching the watchlist occur. Triggered alerts are reported in the Alert Dashboard page and the Triage Alerts page. For more information on alerts, see Chapter 20 Console and Email Alerts.

- **Log to Syslog**: Select the checkbox to log all hits `syslog`. Syslogs are written to `/var/log/cb/notifications/`. In this case, the log filenames have the format `cb-notifications-`*<watchlist ID>*`.log`.

4 Click **Save changes**.

Managing Watchlists

Watchlists can provide you with valuable information about conditions that matter in your environment. You might need to fine-tune watchlists for your environment, based on their performance and the quality of the information they provide.

You can monitor the status of a watchlist to see whether and when it has executed, and whether there are any error conditions associated with the watchlist. If you find that the watchlist is not performing as expected, you can edit, disable, or delete it.

Watchlist Status

Watchlists show the following status in the table view:

- **Queued** – A watchlist was recently created and is waiting to be executed.
- **Timeout** – A watchlist does not execute successfully (or generates an error) after two minutes. A timed-out watchlist will be re-tried, but will only be run on events that appeared between its failed execution and the retry time.
- **Expired** – The watchlist has not had any hits in the specified period. See Configure Watchlist Expiration.
- **Error** – An error happens during watchlist execution and indicates that the watchlist did not execute successfully. If you are unable to resolve an error condition, consider contacting Carbon Black Support.

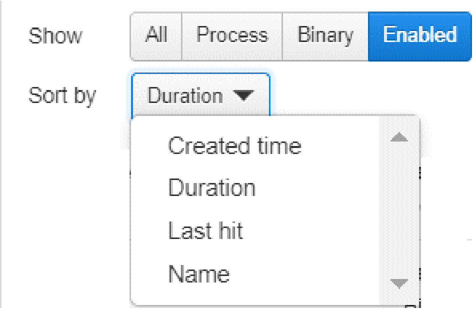
In the **Watchlist Details** panel, descriptive messages display if the last execution of the watchlist resulted in an error or a timeout. For successful executions, the **Watchlist Details** panel shows the following:

- **Last execution** – The time of the last successful execution.
- **Duration** – The duration required to complete execution

Slow or Error-producing Watchlists

Temporary conditions might cause a watchlist to timeout or fail with an error message. However, if a watchlist continues to fail, you might need to investigate it and consider modifying the query or deleting the watchlist.

You can identify slow or error-producing watchlists on the watchlist table by using the **Duration** choice on the **Sort by** menu.



This action produces the following results:

- Watchlists that have not executed successfully, including disabled, queued, errored out or timed out watchlists, appear first. Because you are not usually interested in disabled watchlists, consider clicking the **Enabled** tab to eliminate disabled watchlists from your results.
- After the non-executed watchlists, watchlists that have been executed successfully are listed, beginning with the slowest (longest duration) watchlist and then in descending order of duration.

Duration, timeout and error status is also displayed underneath the watchlist name in the **Watchlist Details** panel.

After you identify a problematic watchlist, you can examine its **Query** field or **Feed Score Criteria** to see whether there are any obvious issues, such as leading wildcards in the query. Chapter 12 Advanced Search Queries includes guidelines for creating queries, including query usage that could cause difficulties.

If you are unable to modify a watchlist in a way that produces efficient, successful performance, you can contact Carbon Black Support for further troubleshooting.

Configure Watchlist Expiration

You can configure Carbon Black EDR to notify you when a watchlist has not received any hits over a specified period of time. This might be a sign that the watchlist is not useful and can be deleted, or perhaps that the query in the watchlist must be modified to be effective.

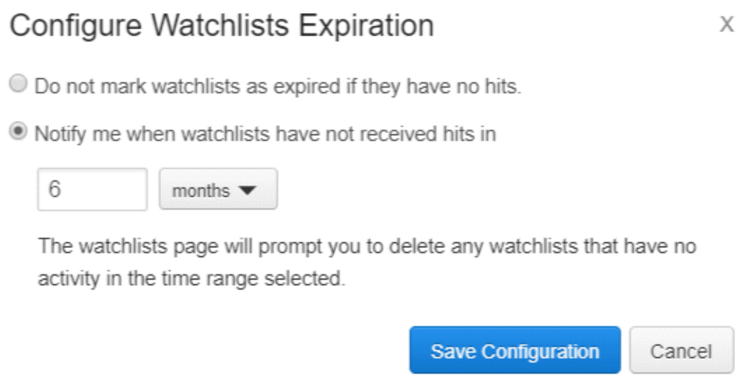
Watchlist expiration is informational only. When a watchlist expires, you are prompted to take action on it, but it is still fully functional unless you delete or disable it.

A single watchlist expiration configuration applies to all watchlists.

Procedure

1 On the navigation bar, click **Watchlists**.

2 Click **Configure Expiration**.



Configure Watchlists Expiration

○ Do not mark watchlists as expired if they have no hits.

● Notify me when watchlists have not received hits in

6 months

The watchlists page will prompt you to delete any watchlists that have no activity in the time range selected.

3 By default, watchlists are marked as **Expired** if they have received no hits over a six-month period. To use a different time period for expiration or to reconfigure a watchlist page that had expiration turned off:

   a Make sure the **Notify me when watchlists have not received hits in** radio button is selected.

   b Enter a number in the box and select the units (days, months, or years).

4 If you do not want any watchlists to be designated as expired, click the radio button that reads **Do not mark watchlists as expired if they have no hits**.

5 Click **Save Configuration**.

Edit a Watchlist

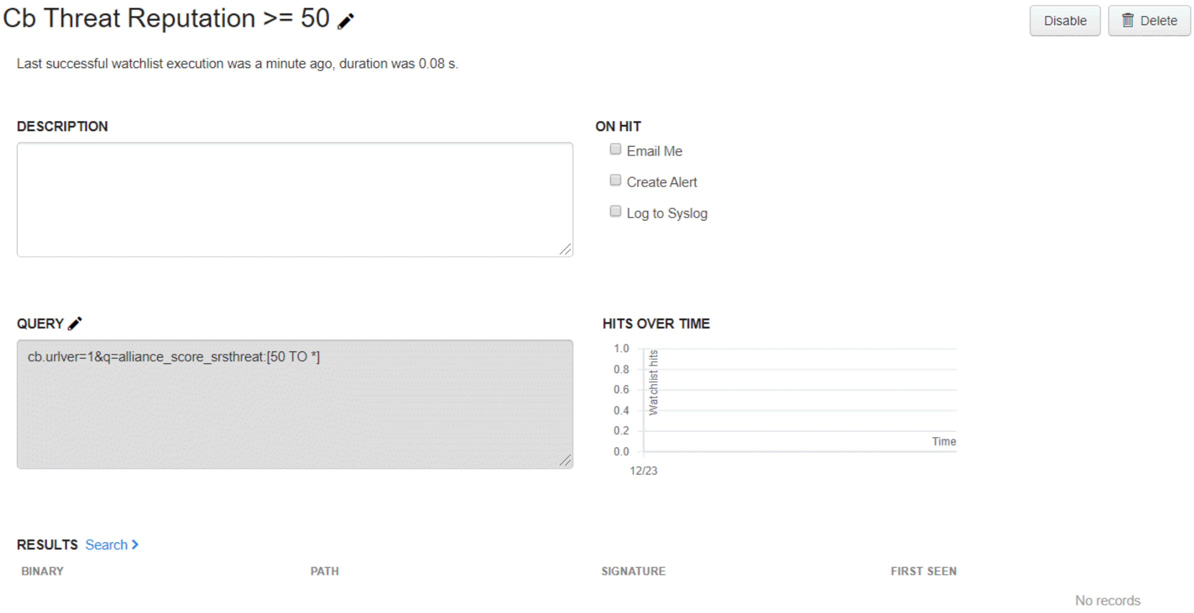
You can edit a watchlist in the **Watchlist Details** panel of the Watchlists page in the Carbon Black EDR console.

For most watchlist changes, the underlying ID that uniquely identifies the watchlist remains the same. However, if you edit the watchlist search query, it effectively becomes a new watchlist.

Procedure

1 On the navigation bar, click **Watchlists**.

2 In the left panel, select the watchlist to edit. Its details appear in the right panel.



Cb Threat Reputation >= 50

Disable | Delete

Last successful watchlist execution was a minute ago, duration was 0.08 s.

DESCRIPTION

ON HIT

□ Email Me

□ Create Alert

□ Log to Syslog

QUERY

cb.urlver=1&q=alliance_score_srsthreat:[50 TO *]

HITS OVER TIME

RESULTS Search >

BINARY	PATH	SIGNATURE	FIRST SEEN

No records

3 You can edit the following attributes of the watchlist:

- To change the name of the watchlist, click the pencil icon next to the name at the top of the page.
- To edit the watchlist query, click the pencil icon for the **Query** box. In the **Edit Watchlist Query** dialog, modify the query and then click **Save Changes**.

  Note Saving a modified watchlist query overwrites the watchlist ID even if the watchlist name is the same. Therefore, any references to the older version of the watchlist, such as in alerts or through the API, are no longer connected.

- To disable the watchlist, click **Disable**. To enable it, click **Enable**.
- To receive email notifications when there are hits that match your search, select **Email Me**. Deselect the checkbox to stop receiving email notifications.
- To send an alert when conditions matching the watchlist occur, select **Create Alert**. Deselect the checkbox to stop sending alerts.
- To log all hits that match the search to `syslog` , select **Log to Syslog**. Syslogs are written to `/var/log/cb/notifications/`. In this case, the log filenames have the form `cb-notifications-`*<watchlist ID>*`.log`.

Delete a Watchlist

You can delete a watchlist in the **Details** panel for that watchlist on the Watchlists page in the Carbon Black EDR console.

Procedure

1 On the navigation bar, click **Watchlists**.

2 In the left panel, select the watchlist to delete. Its details appear in the right panel.

3 In the top-right corner, click **Delete** and click **OK** to confirm the deletion.

Console and Email Alerts

20

This section describes how to create and manage Carbon Black EDR alerts in the Carbon Black EDR console, and how to enable email alerts to report events.

Alerts can be triggered based on watchlist or Carbon Black Threat Intel feed events.

You can create alerts to indicate in the Carbon Black EDR console when suspicious or malicious activity appears on your endpoints. Alerts are available for two types of events:

- Watchlist hits – Watchlists can be configured to send an alert when conditions matching the watchlist occur. See Chapter 19 Watchlists.
- Threat intelligence feed hits – Threat intelligence feeds can be configured to send an alert when that feed reports an IOC. See Chapter 14 Threat Intelligence Feeds.

Triggered alerts are reported in two locations in the Carbon Black EDR console:

- The Head-UP Display (HUD) page contains a summary that shows the number of unresolved alerts, the number of hosts that have unresolved alerts, and other alert-related data, including the alerts for each host. See Viewing Alert Activity on the HUD Page.
- The Triage Alerts page contains more details about triggered alerts and provides a filter and search interface to find alerts that match different criteria. It also allows you to manage the alert workflow, marking the status of each alert from its initial triggering to its resolution. See Managing Alerts on the Triage Alerts Page.

You can configure watchlists and threat intelligence feeds to send email alerts when there is a hit on data from a Carbon Black EDR sensor that matches the watchlist or feed. You can enable email alerts in addition to or instead of the Carbon Black EDR console-based alerts. See Enabling Email Alerts for more information.

Read the following topics next:

Enabling Console Alerts

You can enable Carbon Black EDR console alerts for any watchlist or threat intelligence feed. This topic explains how to enable console alerts.

Consider how many hits you are likely to receive when you enable alerts. Some watchlists or feeds might generate too many hits to be useful, making it more difficult to identify significant alerts. Ideally, an alert should get your attention for issues that you need to follow up on. No alerts are enabled by default.

Enable Console Alerts for a Watchlist

Perform the following procedure to enable Carbon Black EDR console alerts for a watchlist.

Watchlists are user-created, custom, saved searches that are based on process search, binary search, or feed results. You can use watchlists to monitor endpoints for detected IOCs. You can also select the most important watchlists to monitor and add console alerts. Then, you can then view and manage these key watchlist and feed hits in the Triage Alerts page.

Procedure

1. On the navigation bar, click **Watchlists**.
2. In the left panel of the Watchlists page, select the watchlist for which to create an alert.
   Use the **Search** box at the top of the panel to locate a watchlist that does not immediately display.
3. In the right panel, click the **Enable** button if the watchlist is disabled, and then select the **Create Alert** check box.

   The watchlist will begin generating alerts.

Enable or Disable Console Alerts for a Threat Intelligence Feed

Adding a Carbon Black EDR console alert to a feed allows you to highlight hits matching reported malware from a specific source. You can then view and manage high-importance feed and watchlist hits on the Triage Alerts page.

Threat intelligence feeds provide information that helps you identify malware and its sources. Carbon Black EDR integrates with third-party and internal feeds (such as the Carbon Black Threat Intel Reputation and Carbon Black EDR Tamper Detection) that identify hosts on which tamper attempts have occurred.

Prerequisites

Important Make sure you understand the volume of reports that you will receive from any feed before enabling alerts for it. Be sure to read the description of a feed on the Threat Intelligence Feeds page. Some feeds include a specific recommendation not to enable alerts, because of the report volume or percentage of false positives that can occur.

Procedure

1 On the navigation bar, click **Threat Intelligence**.

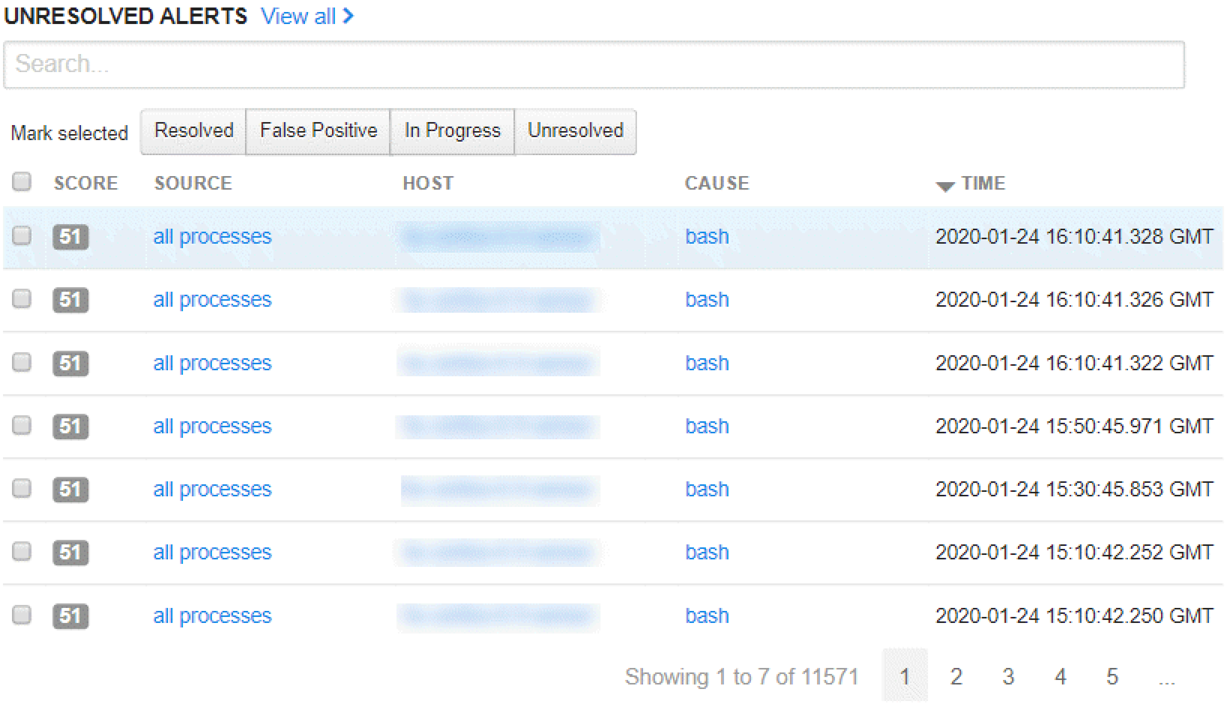
2 Click the **Notifications** drop-down menu.

- Select the **Create Alert** check box for each feed panel to enable console alerts.
- Deselect the **Create Alert** check box for each feed panel to disable console alerts.

Viewing Alert Activity on the HUD Page

You can view alert activity on the Head-Up Display (HUD) page.

The HUD page is a customizable page that provides a summary of alerts on hosts that report to your Carbon Black EDR server. See Chapter 21 Head-Up Display Page.



□	SCORE	SOURCE	HOST	CAUSE	TIME
□	51	all processes	[redacted]	bash	2020-01-24 16:10:41.328 GMT
□	51	all processes	[redacted]	bash	2020-01-24 16:10:41.326 GMT
□	51	all processes	[redacted]	bash	2020-01-24 16:10:41.322 GMT
□	51	all processes	[redacted]	bash	2020-01-24 15:50:45.971 GMT
□	51	all processes	[redacted]	bash	2020-01-24 15:30:45.853 GMT
□	51	all processes	[redacted]	bash	2020-01-24 15:10:42.252 GMT
□	51	all processes	[redacted]	bash	2020-01-24 15:10:42.250 GMT

By default, the **Unresolved Alerts** panel displays all unresolved alerts for a sensor. You can also display resolved, false positive, and in- progress alerts by clicking a button at the top of the **Unresolved Alerts** panel:

- **Resolved**
- **False Positive**
- **In Progress**

- **Unresolved**

Note You can enlarge the **Unresolved Alerts** panel to display more details by holding your left mouse button down on the bottom-right expansion icon and dragging the panel to the desired size.

The **Unresolved Alerts** panel contains the following columns:

Note Some columns in this panel are sortable, such as the **Score** and **Time** columns. You can determine if columns are sortable by hovering your cursor over the column name; sortable column names will turn black and your cursor will change to a hand icon. An arrow appears, indicating the sort direction (ascending/descending).

Pane	Description
Score	Displays the alert severity, where 100 is a severe threat and 1 is not a threat.
Source	Displays the feed that is associated with the alert, such as threat intelligence and watchlist feeds. Clicking a link in this column opens the associated page.
Host	Displays the host that is associated with the alert. Clicking a link in this column opens the Sensors page.
Cause	When the alert is caused by a binary, this column displays the binary's MD5 hash. Clicking on this link takes you to the Binary Search page. See Chapter 11 Binary Search and Analysis. When the alert is caused by a process, this column displays the process name. Clicking on this link takes you to the Process Search page. See Chapter 10 Process Search and Analysis.
Time	Displays the time when the alert occurred.

The **Unresolved Alerts** panel also contains a **View all** link in the top-right corner. Clicking this link displays the Triage Alerts page.

Managing Alerts on the Triage Alerts Page

This section describes how to manage alerts on the Triage Alerts page in the Carbon Black EDRconsole.

When an alert is received that indicates suspicious or malicious activity, incident responders must:

- Determine the seriousness of the alert.
- Determine whether the alert indicates a sufficiently severe threat.
- Find a way to resolve a serious threat.

These tasks might involve using the following Carbon Black EDR features:

- Endpoint Isolation; see Isolating an Endpoint.
- Live Response; see Using Live Response.
- Banning; see Banning Process Hashes.

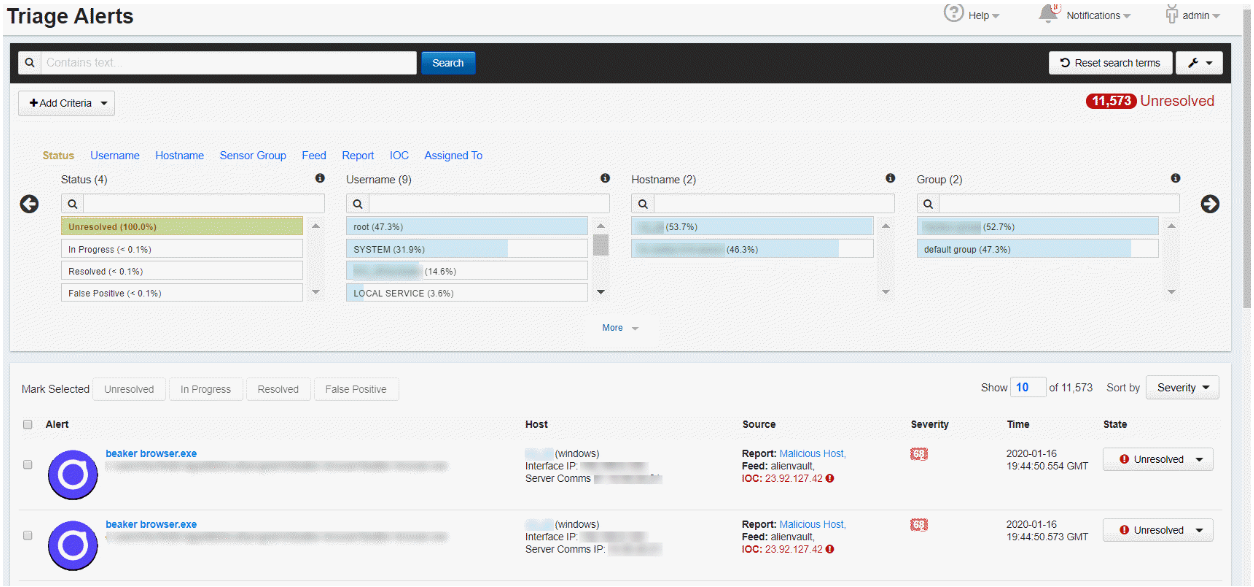
It might also require using other tools.

Given the high volume of threat reports, it is critical to prioritize, investigate, and keep track of alert statuses. After an alert is resolved, it should be removed from the list of threats requiring attention so that ongoing threats can be addressed.

The Triage Alerts page provides features for alert management. It includes search and filtering capabilities for locating specific alerts or alert types. It also allows you change alert status.

- On the navigation bar, click **Triage Alerts**.

Note You also can navigate to the Triage Alerts page from the HUD page by clicking **View all** in the **Unresolved Alerts** panel. See Viewing Alert Activity on the HUD Page.



□ Alert	Host	Source	Severity	Time	State
□ beaker browser.exe [redacted]	[redacted] (windows) Interface IP: [redacted] Server Comms [redacted]	**Report:** Malicious Host, **Feed:** alienvault, **IOC:** 23.92.127.42	68	2020-01-16 19:44:50.554 GMT	Unresolved
□ beaker browser.exe [redacted]	[redacted] (windows) Interface IP: [redacted] Server Comms IP: [redacted]	**Report:** Malicious Host, **Feed:** alienvault, **IOC:** 23.92.127.42	68	2020-01-16 19:44:50.573 GMT	Unresolved

The Triage Alerts page is divided into three major sections:

- The top section includes the **Search** field and button, **Add Criteria** button, **Reset search items** button, and **Actions** menu.
- The middle section contains filters that are category-specific lists (**Status**, **Username**, and so on). These filters show the percentage of alerts that match different values in each category, and allow you to filter the view to show alerts that match values.
- The bottom section contains the **Alerts** table, which contains details for alerts that match the search criteria that is entered in the first two sections.

Display the Report Name on the Triage Alerts Page

The middle section of the Triage Alerts page on the Carbon Black EDR console lets you filter by various criteria, including Reports.

By default, the Reports display shows the report ID (for example, dbe2eab5-3829-45df-b6c4-3dfb7a215d69). You can change the display to show the report name (for example, “PowerShell executed with encoded instructions”).

To change the display, you must change a setting in the `cb.conf` file. The default value of this setting is `False`. For more information about `cb.conf`, see the *Carbon Black EDR Server Configuration Guide*.

Caution If you enable this setting, additional memory will be used in proportion to the number of reports on your server.

Procedure

1. On the Carbon Black EDR server, open `/etc/cb/cb.conf` for editing.
2. Set `FeedHitLoadReportTitles=True.`
3. Set the number of characters (from -1 to 80) for the report name in the `FeedHitMaxReportTitleLength` field. The default (and maximum) number of characters is 80. A value of **-1** keeps the report name from being truncated in bus events, syslog, and email notifications.

   ```
   FeedHitLoadReportTitles=True
   FeedHitMaxReportTitleLength=80
   ```

4. Restart cb-enterprise services.

Results

After you have changed the `cb.conf` setting and restarted cb-enterprise services, the report names are populated in the following places:

- In the Triage Alerts page Records filter.
- Bus events.
- Syslog notifications.
- Email notifications. Both report ID and report name are displayed in the email. If the feature is turned off, the report name is displayed as “Unknown”.

Reviewing Alerts on the Triage Alerts page

This section describes how to review alerts on the Triage Alerts page in the Carbon Black EDR console.

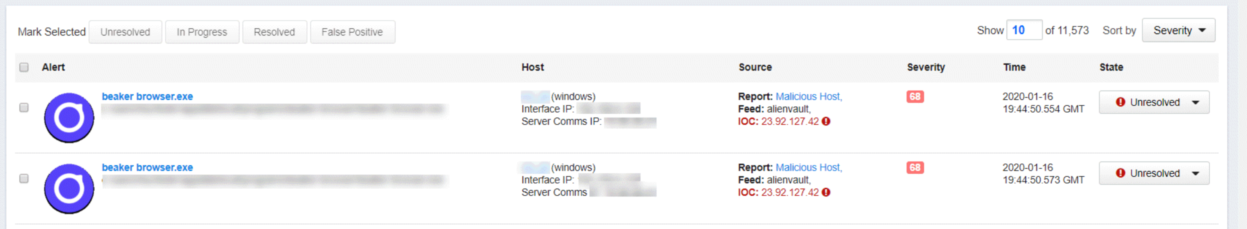
Each row in the **Alerts** table shows the description and data for an individual alert. The description and data that appears can vary depending on a variety of factors, including:

- The source and type of the alert.
- Whether the binary for a process has been signed.
- Whether a binary is “Trusted” by the Carbon Black EDR Alliance.

The **Alerts** table has multiple tools for adjusting the table display:

- **Sort order** – You can sort the **Alerts** table using the **Sort By** button You can sort by:
  - Severity (default)
  - Most Recent
  - Least Recent
  - Alert Name Ascending
  - Alert Name Descending
- **Page navigator** – You can use the page navigation bar in the bottom-right corner of the **Alerts** table to move between pages for table views that do not fit on a single page.

Alerts Table Data



□	Alert	Host	Source	Severity	Time	State
□	beaker browser.exe [redacted]	[redacted] (windows) Interface IP: [redacted] Server Comms IP: [redacted]	**Report:** Malicious Host, **Feed:** alienvault, **IOC:** 23.92.127.42	68	2020-01-16 19:44:50.554 GMT	Unresolved
□	beaker browser.exe [redacted]	[redacted] (windows) Interface IP: [redacted] Server Comms [redacted]	**Report:** Malicious Host, **Feed:** alienvault, **IOC:** 23.92.127.42	68	2020-01-16 19:44:50.573 GMT	Unresolved

- **Alert** – Contains the following details:
  - An icon that represents the process or binary that caused the threat alert, if available. If there is no special icon for this binary, a generic file icon is used.

    Note Tamper alerts show what feed is triggered; the icon is of the host type.
  - The directory path where the process or binary is installed.
  - If this is a binary, the blue process link takes you to the Binary Details page. If this is a process, the blue process link takes you to the Process Analysis page.
- **Host** – Shows host details and a link to the Sensor Details page.
- **Source** – The watchlist or feed that triggered the alert with a link to that watchlist or feed.
- **Severity** – The severity score of the alert that Carbon Black EDR produces based on underlying alert data. Click the **Severity** number to show additional details by which the severity is calculated:
  - Feed rating
  - Report scores
  - Confidence
  - Criticality

  The **Severity** numbers are color-coded, with red being severe threats and green being low threats. A score of 100 represents the most severe alerts.

- **Time** – The time when the alert was triggered.
- **State** – The alert state, which includes:
  - **Mark as Resolved** – Select this when the alert is resolved.
  - **Mark as Unresolved** – By default, only unresolved alerts are displayed.
  - **Mark as In-Progress** – Select this for alerts that have resolutions in progress.
  - **Mark as False Positive** – Select this for alerts that were not true threats.
- See Managing Alert Status on the Triage Alerts Page.

Managing Alert Status on the Triage Alerts Page

You can change the status of individual alerts or all alerts in the current view on the Triage Alerts page in the Carbon Black EDR console.

Changing alert status is strictly for alert management purposes. It helps you organize alerts that need attention, are being investigated, have been resolved, or are false positives.

Change an alert status to indicate what you are doing or have done based on your review of an alert. An alert status has no effect on the actual issue that caused the alert.

In the **Alerts** table on the Triage Alerts page, the far-right column includes an icon representing the current alert status and a drop-down list for changing that status.

Change Status for Multiple Alerts

Perform the following procedure to change the status of all alerts matching a search and/or filter on the Triage Alerts page in the Carbon Black EDR console.

Procedure

1. On the navigation bar, click **Triage Alerts**.
2. On the Triage Alerts page, enter the search string and/or filter criteria for alerts to change.
3. From the **Actions** menu, select the **Mark all** menu option for the status to assign.
4. Click **OK** in the confirmation window to change the status of all of the alerts on the page.

   Note When using the **Mark all** commands, be sure that you want to change all of the alerts matching the current filter and search, including those alerts that are on pages that are not displayed. After you change the status, there is no “undo” command. Be especially careful about changing alert status when the view is unfiltered (showing all alerts).

Change Status for a Single Alert

Perform the following procedure to change the status of a single alert on the Triage Alerts page in the Carbon Black EDR console.

Procedure

1. On the navigation bar, click **Triage Alerts**.

2 In the **Alerts** table, select the check box to the left of the alert that has a status that you want to change.

3 From the **Actions** drop-down list, select the appropriate option for the status you want to assign.

4 Click **OK** in the confirmation window to change the status of the selected alert.

> **Note** Changed alerts will disappear from the current view if you have filtered the page for a different status.

Ignore Future Events for False Positive Alerts

Carbon Black EDR lets you ignore future instances of a false positive alert from a threat feed. You can choose to ignore an individual alert, or specify that all alerts matching your search criteria should be ignored in the future.

Feeds use a variety of criteria to determine if a file or event is a threat, and you might not agree with all of the alerts that are generated by certain feeds. When you review alerts and determine that an alert is not reporting an actual threat, you can mark that alert as a “false positive”, so you can eliminate it from the list of alerts that require your attention.

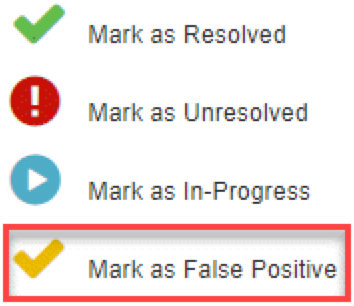
> **Note** Only threat feed alerts can be designated as alerts to ignore. Alerts from watchlist matches are always triggered, since watchlists are assumed to use criteria that your Carbon Black EDR users select.

Procedure

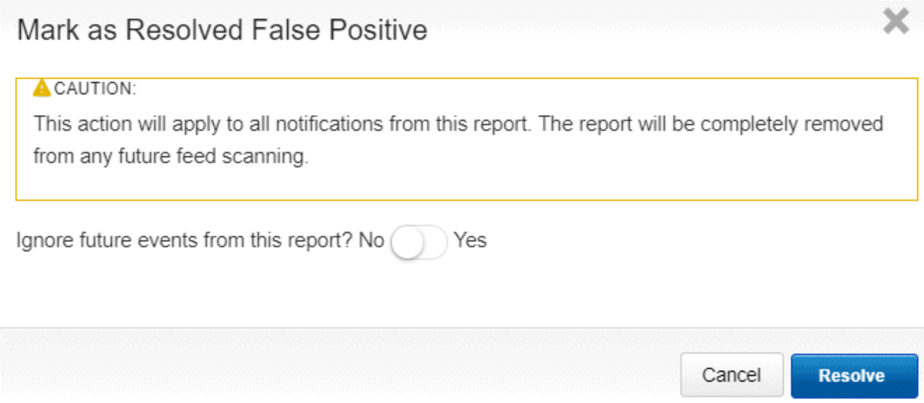
1 On the navigation bar, click **Triage Alerts**.

2 In the **Alerts** table, select the check box to the left of the alert.

3 In the **Status** column, select **Mark as False Positive** in the drop-down menu.



4 In the **Mark as Resolved False Positive** dialog, you can ignore future events from this report by moving the slider button to **Yes**. Click **Resolve**.



Mark as Resolved False Positive

CAUTION:

This action will apply to all notifications from this report. The report will be completely removed from any future feed scanning.

Ignore future events from this report? No ○ Yes

Note Marking events from multiple alerts to be ignored involves searching for the alerts you want to ignore, confirming that the results are what you expect, and then making a bulk resolution.

Enabling Email Alerts

You can enable email alerts to report events that trigger watchlist and threat intelligence feed alerts.

This feature informs you of events of interest, even when you are not logged into the Carbon Black EDR console. You can then go to the console to investigate and resolve the alert. The email alerts feature is enabled on a per-console user basis.

Configure an Email Server for Alerts

Perform the following procedure to configure an email server for alerts.

Prerequisites

Before enabling email alerts for specific watchlists or feeds, you should decide which email server to use. You can:

- Use your own email server
- Use the Carbon Black EDR external email server
- Opt out of email alerts

If you use the Carbon Black EDR external email server, the following information is sent through the server and stored by Carbon Black EDR:

- Your server ID
- The time of the email

- The name of the watchlist or feed are that triggered the hit

Important Carbon Black strongly recommends that you use your own email server because email sent through the Carbon Black EDR external email server is sent over the Internet in clear text.

Carbon Black also encourages you to use your own email server if you are concerned about the potential delays that can occur during times of high processing of alerts for all Carbon Black EDR servers subscribed to the Carbon Black EDR external email server. Using your own email server reduces the possibility of delayed emails.

Procedure

1. Log in to Carbon Black EDR as a Global Administrator or as a Carbon Black Hosted EDR Administrator.
2. Cick ***Username***> **Settings**.
3. On the Settings page, click **E-Mail**.
   - Select whether to use the Carbon Black external mail server.
   - If you want to use your own mail server, select that option and configure the mail server with the connection criteria.

   All email alerts for all console users will use these settings.

Enabling Specific Email Alerts

After you have configured an email server, any watchlist or feed can be configured to send email alerts when it gets a hit on a Carbon Black EDR sensor.

You can turn on/off email alerts for individual watchlists and feeds as needed (for example, if you find that a watchlist or feed is creating too much email traffic). Email alerts for any specific watchlist or feed are enabled on a per-user basis.

Note If you have upgraded from a previous release of Carbon Black EDR, any email alerts that you had enabled for watchlists and threat intelligence feeds remain enabled after the upgrade.

Enable Email Alerts for a Watchlist

Perform the following procedure to enable email alerts for a watchlist.

Procedure

1. On the navigation bar, click **Watchlists**.
2. Select the watchlist for which to enable email alerts. If the watchlist name is not visible or you are not sure what the name is, use the **Search** field.
3. Confirm that the watchlist is enabled.
4. Check the **Email Me** check box.

Enable Email Alerts for a Threat Intelligence Feed

Perform the following procedure to enable email alerts for a Threat Intelligence feed.

Procedure

1 On the navigation bar, click **Threat Intelligence**.

2 To activate email alerts for a feed, select the **Email Me on Hit** check box.

Head-Up Display Page

21

This section describes how to use the Head-Up Display (HUD) page, which is a customizable dashboard in the Carbon Black EDR console.

The HUD page provides a summary of alerts on hosts that report to your Carbon Black EDR server. It provides a quick reference view of the system.

- To view the HUD page, on the navigation bar, click the logo at the top left corner of the console.

You can customize the HUD page in the following ways:

- To reposition HUD panels, hold down your left mouse button on a panel and drag the panel to the desired location.
- To resize HUD panels, hold down your left mouse button on the bottom-right resizing icon and drag the panel to the desired size. Note that larger panels display more details.

As you customize the HUD layout, the layout is automatically saved for future use per device.

Some columns are sortable. You can determine if a column is sortable by hovering your cursor over the column name. Sortable column names change colors, and your cursor changes to a hand icon. An arrow appears that indicates the sort direction (ascending or descending).

The following topics describe HUD page panels. However, the **Unresolved Alerts** panel is more fully described in Viewing Alert Activity on the HUD Page.

Read the following topics next:

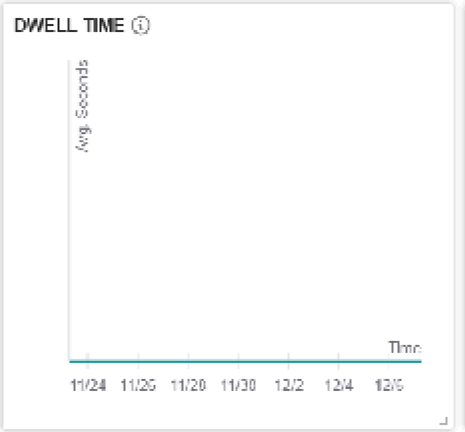
- Dwell Time Panel
- Endpoint Hygiene Panel
- Event Monitor Panel
- Query Duration Panel
- Resolution Time Panel
- Saved Searches Panel
- Sensors Panel
- Task Errors Panel

- Unresolved Alerts Panel

Dwell Time Panel

The **Dwell Time** panel shows the daily average for how long malware dwells on hosts that are reporting to this server over a 30-day period.

This average is based on the duration between when an undesirable binary first appears on one of the hosts and it is longer reported on any hosts. The following image shows zero dwell time activity; this is desirable.



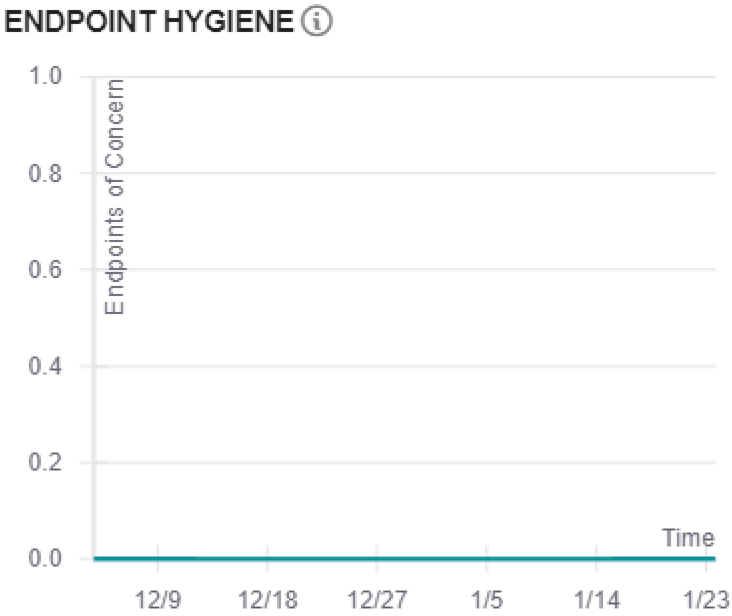
Endpoint Hygiene Panel

The **Endpoint Hygiene** panel shows the daily percentage of hosts that are reporting suspect processes over a 30-day period.

This percentage is based on two values that Carbon Black EDR records:

- The total number of active hosts in the network.
- The number of hosts that have one or more bad processes.

The following image shows zero endpoint activity, which is desirable.

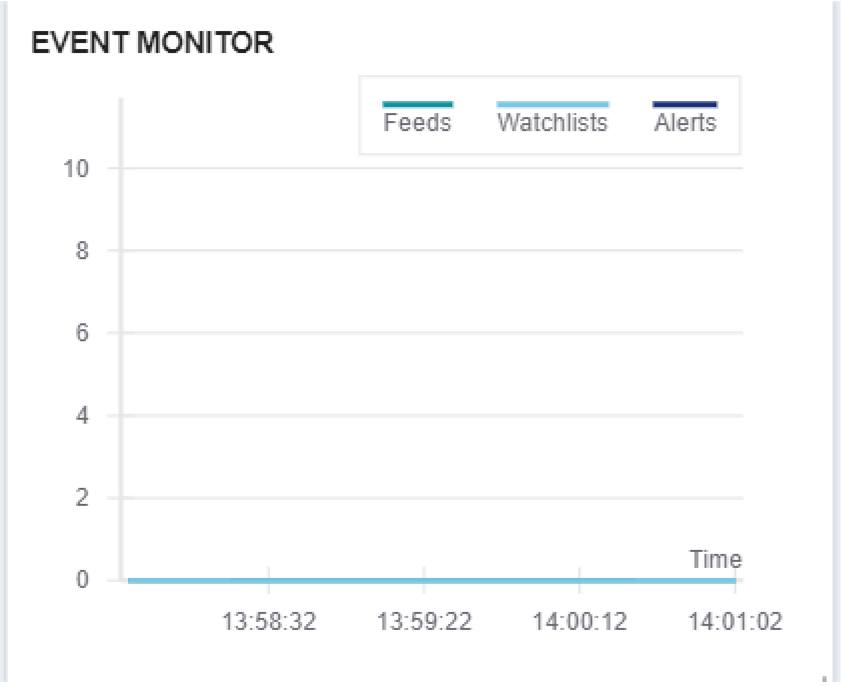


Event Monitor Panel

The **Event Monitor** panel provides a live feed of event activity. It updates every five seconds.

Vertical bars indicate alert activity, such as resolving an alert or incoming alerts.

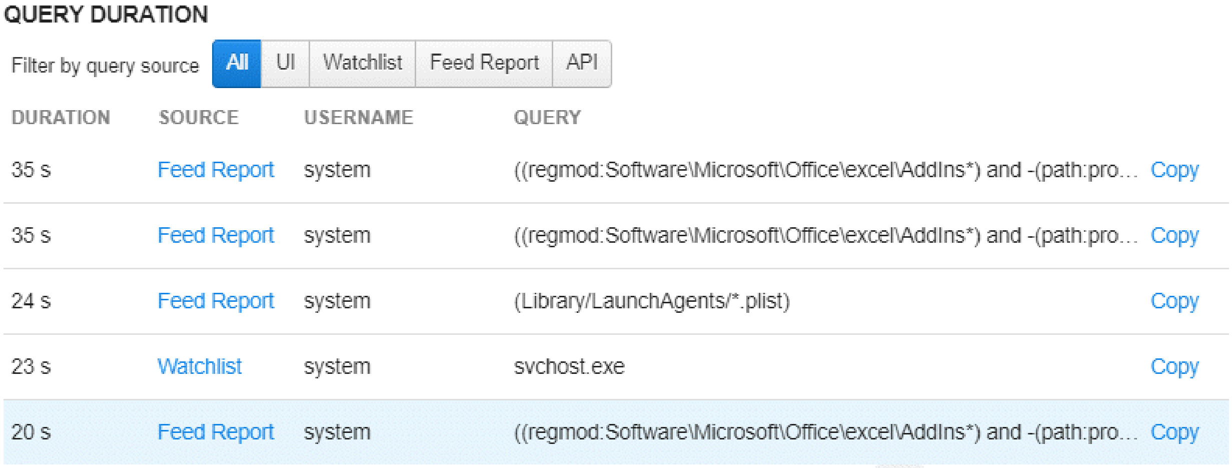
Horizontal lines indicate watchlist activity.



Query Duration Panel

The **Query Duration** panel presents queries that take longer than a second to complete.

At a glance, you can see which queries are taking a long time to complete, and take action to improve query structures and efficiency.



QUERY DURATION

DURATION	SOURCE	USERNAME	QUERY	
35 s	Feed Report	system	((regmod:Software\Microsoft\Office\excel\AddIns*) and -(path:pro…	Copy
35 s	Feed Report	system	((regmod:Software\Microsoft\Office\excel\AddIns*) and -(path:pro…	Copy
24 s	Feed Report	system	(Library/LaunchAgents/*.plist)	Copy
23 s	Watchlist	system	svchost.exe	Copy
20 s	Feed Report	system	((regmod:Software\Microsoft\Office\excel\AddIns*) and -(path:pro…	Copy

You can filter the displayed queries in the following ways:

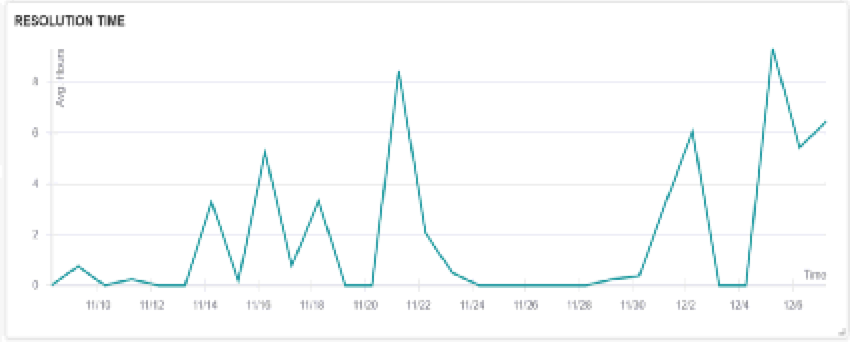
- **All** – Displays all queries that take longer than a second to complete.
- **UI** – These slow queries are generated at the user interface.
- **Watchlist** – Automated queries. Watchlist queries are created by Carbon Black EDR users and run every 10 minutes.
- **Feed Report** – Automated queries that the threat research team generates. You cannot edit the queries, but you can ignore them.
- **API** – These queries are run via an API.

A user or script can run UI- or API-generated queries many times. If any query takes long enough to appear in the **Query Duration** panel, multiple executions of that query add to the overall effect.

For queries that are too long to display in the panel, you can hover over the query to cause the entire query to display in the hover text. You can also click **Copy** to copy a query. This is useful for closely examining a complex slow-running query, and for editing a query to improve performance.

Resolution Time Panel

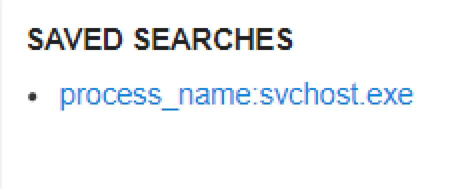
The **Resolution Time** panel contains a graph that displays the average time (in hours) between reporting and resolution of alerts on each day over a 30-day period.



Saved Searches Panel

The **Saved Searches** panel provides a convenient list of your saved process searches.

You can click a saved search to go to the Process Search page. See Save a Process Search.

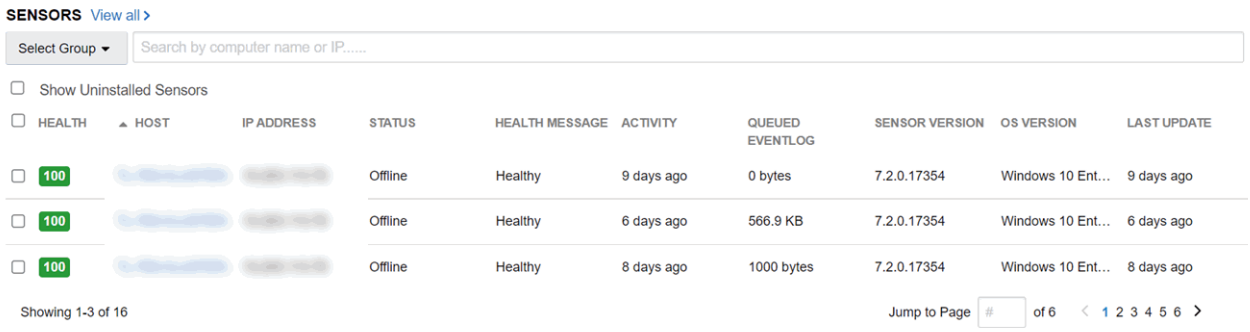


SAVED SEARCHES

- process_name:svchost.exe

Sensors Panel

Use the **Sensors** panel to monitor and manage sensor hosts.



SENSORS View all >

Select Group ▾ | Search by computer name or IP......

□ Show Uninstalled Sensors

□	HEALTH	▲ HOST	IP ADDRESS	STATUS	HEALTH MESSAGE	ACTIVITY	QUEUED EVENTLOG	SENSOR VERSION	OS VERSION	LAST UPDATE
□	100	[redacted]	[redacted]	Offline	Healthy	9 days ago	0 bytes	7.2.0.17354	Windows 10 Ent...	9 days ago
□	100	[redacted]	[redacted]	Offline	Healthy	6 days ago	566.9 KB	7.2.0.17354	Windows 10 Ent...	6 days ago
□	100	[redacted]	[redacted]	Offline	Healthy	8 days ago	1000 bytes	7.2.0.17354	Windows 10 Ent...	8 days ago

Showing 1-3 of 16

Jump to Page # of 6 < 1 2 3 4 5 6 >

By default, uninstalled sensors do not display in this panel. Select the **Show Uninstalled Sensors** checkbox to show uninstalled sensors.

By default, all sensor hosts in your organization display; in this case, you cannot perform any actions on the displayed sensor hosts. You can select a group for which you have permissions and then perform the following actions on the hosts in that group:

- Sync
- Restart
- Uninstall

- Isolate; see Isolating an Endpoint.
- Remove isolation; see Restore Connectivity to an Isolated Endpoint.

You can also search for a specific host by computer name or IP address.

The **Sensors** panel contains the following columns:

Pane	Description
Activity	Displays the time related to the sensor activity.
Health	Displays the sensor health score, where 100 is healthy. Lower numbers indicate problems. See Chapter 24 Sensor Health Score Messages.
Health Message	Displays a health message that relates to the sensor health score. See Chapter 24 Sensor Health Score Messages.
Host	Displays the name of the host on which the sensor is installed. Click the host name to go to the Sensors page for that host. See Viewing Sensor Details.
Sensor Version	Displays the sensor version.
Status	Indicates whether the sensor is online or offline, and whether the sensor is undergoing any activity. For example, if a sensor is online and syncing, the status displays syncing-online.

See Chapter 5 Managing Sensors.

Task Errors Panel

The **Task Errors** panel displays errors that are generated from backend processes (tasks or jobs). By default, all errors are displayed; however, you can filter to show only tasks or only jobs.

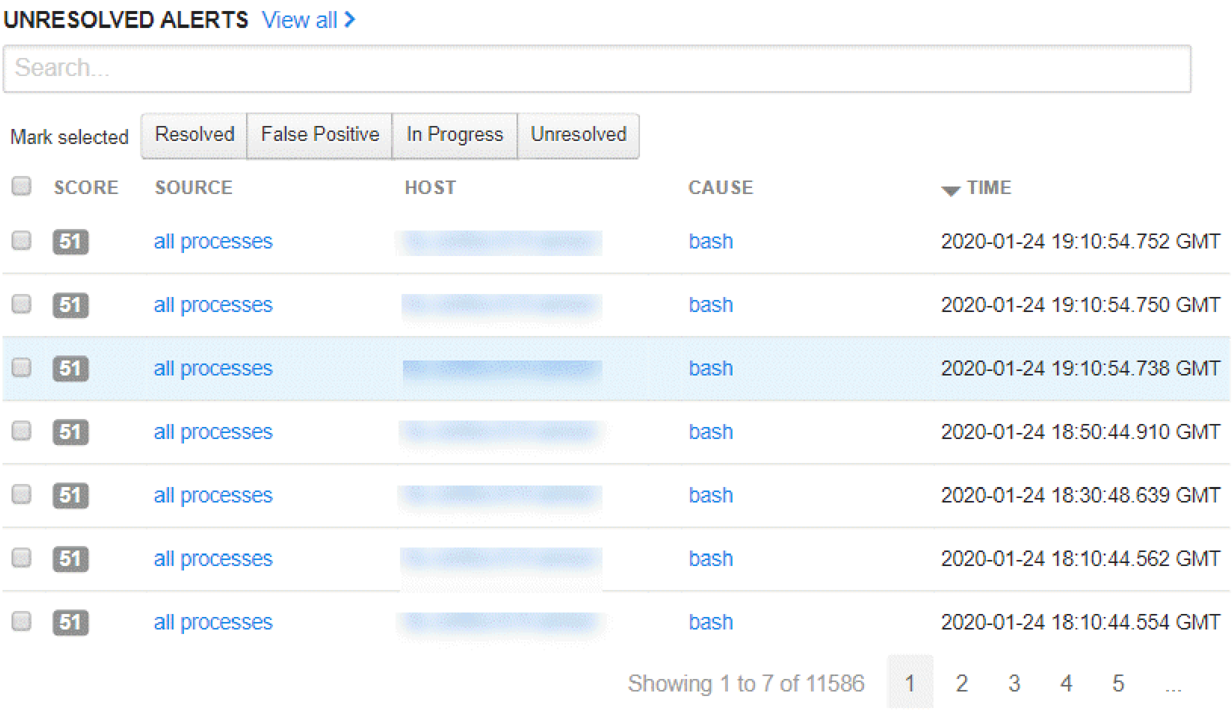
You must be an administrator to view this data.

TASK ERRORS ⓘ
Filter by task type
All
Task
Job
TYPE
NAME
TIMESTAMP
ERROR
Showing 0 of 0
Jump to Page
#
of 0

Most errors are truncated; to view the entire error, click **Copy** underneath the item of interest. The full entry is copied to your clipboard.

Unresolved Alerts Panel

By default, the **Unresolved Alerts** panel displays all unresolved alerts for a sensor. You can also display resolved, false positive, and in-progress alerts.



UNRESOLVED ALERTS View all >

Search...

Mark selected Resolved False Positive In Progress Unresolved

□	SCORE	SOURCE	HOST	CAUSE	TIME
□	51	all processes	[redacted]	bash	2020-01-24 19:10:54.752 GMT
□	51	all processes	[redacted]	bash	2020-01-24 19:10:54.750 GMT
□	51	all processes	[redacted]	bash	2020-01-24 19:10:54.738 GMT
□	51	all processes	[redacted]	bash	2020-01-24 18:50:44.910 GMT
□	51	all processes	[redacted]	bash	2020-01-24 18:30:48.639 GMT
□	51	all processes	[redacted]	bash	2020-01-24 18:10:44.562 GMT
□	51	all processes	[redacted]	bash	2020-01-24 18:10:44.554 GMT

Showing 1 to 7 of 11586 1 2 3 4 5 ...

See Viewing Alert Activity on the HUD Page.

Netconn Metadata 22

This section describes additional or recently added netconn metadata in Carbon Black EDR. It specifically describes TLS fingerprinting.

Overview of TLS Fingerprinting

Transport Layer Security (TLS) fingerprinting is a platform-independent method for creating TLS fingerprints that can easily be shared for improved threat intelligence. TLS fingerprints are properties of a netconn event for TCP connectivity only.

JA3 and JA3S are TLS fingerprinting methods. JA3 fingerprints how a client application communicates over TLS, and JA3S fingerprints the server response. Combined, they create a fingerprint of the cryptographic negotiation between client and server.

JA3, when used in combination with JA3S, is a useful method to fingerprint a TLS negotiation between client and server. When used in conjunction with a process hash, even greater fidelity can be achieved. For example, some Peer-to-Peer (P2P) client connections can be tracked via TLS fingerprinting. This can be used to correlate an application if the binary and/or process metadata has been changed to avoid more direct forms of identification. Additionally, commodity malware variants often re-use cryptographic information, resulting in a common JA3 hash across families.

Read the following topics next:

- How TLS Fingerprinting Works
- TLS Fingerprinting Implementation

How TLS Fingerprinting Works

To initiate a TLS session, a client sends a TLS `Client Hello` packet following the TCP handshake. This packet, and the way in which it is generated, is dependent on packages and methods that are used when building the client application.

The server responds with a TLS `Server Hello` packet that is based on server-side supported ciphers and configurations as well as details in the `Client Hello` .

Because TLS negotiations are transmitted in the clear, it is possible to fingerprint and potentially identify client applications using the details in the TLS `Client Hello` packet.

The JA3 method gathers the decimal values of the bytes for the following fields in the `Client Hello` packet:

- Version
- Accepted cipher suites
- List of extensions
- Elliptic curves
- Elliptic curve formats

It then concatenates those values together to create an MD5 hash (or unique fingerprint) that can enhance traditional cybersecurity approaches such as allow lists, deny lists, and searching for IOCs.

The JA3S method then gathers the decimal values of the bytes for the following fields in the `Server Hello` packet:

- Version
- Accepted cipher
- List of extensions

It concatenates these values in the same way as the `Client Hello` packet, resulting in an MD5 hash known as a JA3S fingerprint.

TLS Fingerprinting Implementation

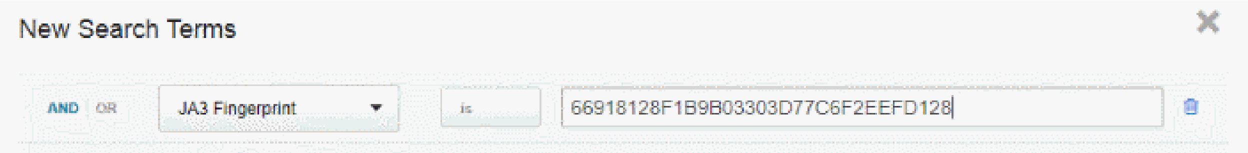
TLS fingerprinting is available with the 7.1.0 release of Carbon Black EDR (for Carbon Black EDR Windows 7.0.0 and higher sensors only). It provides additional endpoint telemetry that can be delivered to the Carbon Black EDR server, and used for narrowing investigations of known malware by identifying known TLS fingerprints.

TLS fingerprints can be specified as IOCs in custom threat feeds. See Chapter 14 Threat Intelligence Feeds.

TLS fingerprints can be used in the following ways.

Process Search

TLS fingerprints are searchable via Process Search. See Overview of Process Search. For example:



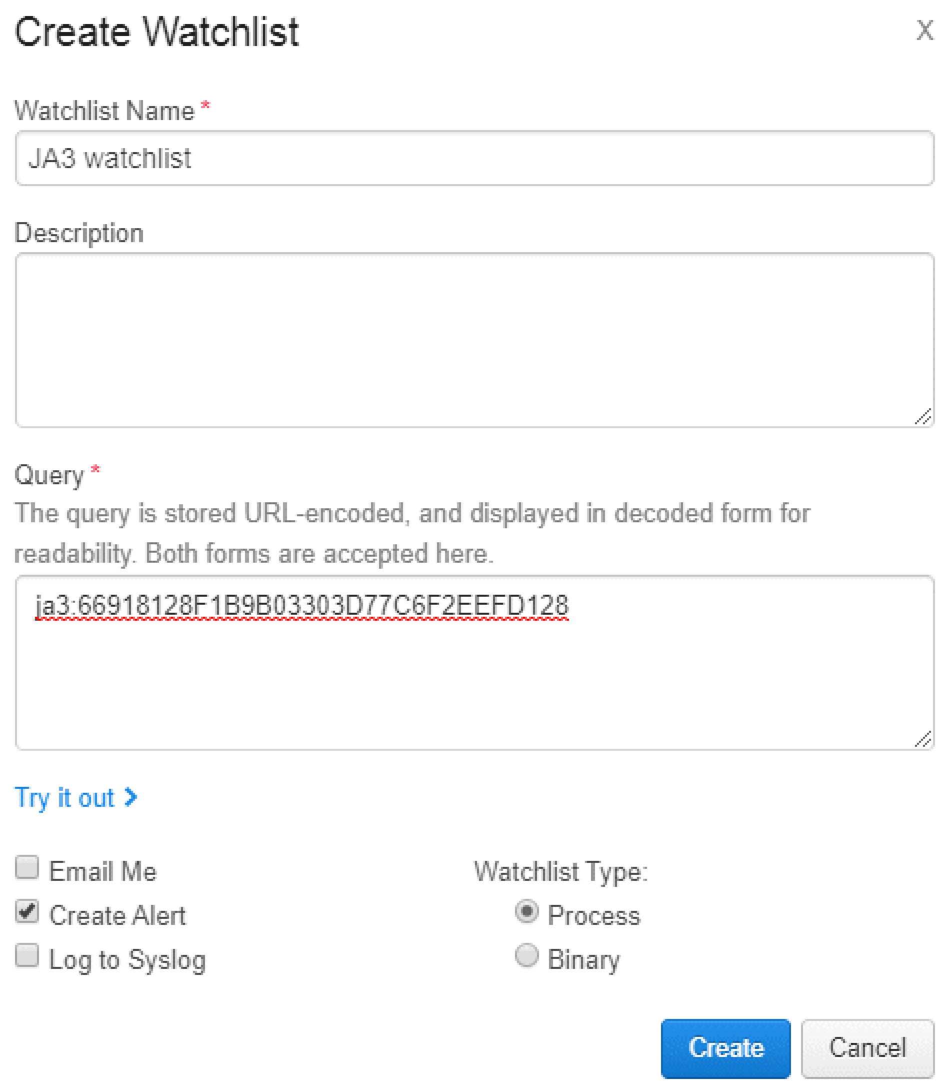
JA3 Fingerprint is 66918128F1B9B03303D77C6F2EEFD128

Process Analysis

TLS fingerprints display in the Process Analysis page (under netconn events), and as quick filters. See Chapter 10 Process Search and Analysis and Process Event Filters.

Watchlists

TLS fingerprints can be used in watchlists. See Chapter 19 Watchlists. For example, to create a TLS fingerprint watchlist:



Create Watchlist

Watchlist Name *: JA3 watchlist

Description:

Query *
The query is stored URL-encoded, and displayed in decoded form for readability. Both forms are accepted here.

```
ja3:66918128F1B9B03303D77C6F2EEFD128
```

☐ Email Me
☑ Create Alert
☐ Log to Syslog

Watchlist Type:
● Process
○ Binary

In addition, TLS fingerprints can trigger an alert, email or syslog event.

Sensor Parity

23

This section contains two tables that show which Carbon Black EDR features or configurations are available for Carbon Black EDR sensors on each operating system platform.

In addition:

- Sensors are discussed in Chapter 5 Managing Sensors
- Sensor groups are discussed in Chapter 6 Sensor Groups

Read the following topics next:

- Sensor Feature Support
- Sensor Group Feature Support

Sensor Feature Support

The following table describes whether certain features are available on Carbon Black EDR sensors.

Feature	Windows	Linux	macOS
Binaries (Collection)	Yes	Yes	Yes
Binary Info (Collection)	Yes	Yes	Yes
BinaryModule loads (Collection)	Yes	Yes	Yes
Carbon Black Live Response	Yes	Yes	Yes
Child Process events (Collection)	Yes	Yes	Yes
Compatibility Control	No	No	Yes
Cross Process events (Collection)	Yes	No	No
Retention Maximization	Yes	No	Yes
Diagnostics collection with SensorDiags	Yes	Yes	Yes
Disable sensor operation events	Yes	No	No
EMET events (Collection)	Yes	N/A	N/A

Feature	Windows	Linux	macOS
File modifications (Collection)	Yes	Yes	Yes-1
Global VDI Support	Yes	Yes	Yes
Hash Banning	Yes	Yes-2	Yes
Hash Banning Allow List (restrictions)	Yes	No	No
Improved proxy support: WPAD & PAC files	Yes	No	No
Known DLLs (Dylib/Mac) Filtering	Yes	No	Yes
Network Connections (Collection)	Yes	Yes	Yes
Network Connections for IPv6 (Collection)	Yes	Yes	Yes
Network Isolation	Yes	Yes-2	Yes
Non-Binary File Writes (Collection)	Yes	Yes	Yes
ODX Support	Yes	N/A	N/A
Process Information (Collection)	Yes	Yes	Yes
Process user context (Collection)	Yes	Yes	Yes
Proxy Support (unofficial support)	Yes	Yes	Yes
Registry modifications (Collection)	Yes	N/A	N/A
Server TLS certificate swapping	Yes-3	No	Yes-3
SHA256 hashes in events (Collection)	Yes-4	No	Yes-4
Support for FIPS	Yes	No	No
Tamper Detection	Yes	No	No
Tamper Protection	Yes	No	No
TLS JA3 and JA3S Fingerprinting	Yes	No	No

Note 1 - The macOS sensor reports a file write event at the time a process opens the file. This event is based on the requested access mask. It is not based on actual writes. Even if the process does not write anything in the file, a file write event occurs.

2 - Currently available eBPF-based sensors (for RHEL/CentOS 8.0 and SUSE 12&15) do not support isolation or banning.

3 - TLS cert swapping support is for sensor versions Windows 6.2.3-win and macOS 6.2.5-osx and above.

4 - SHA-256 sensor support begins with 6.2.x sensors for both Windows and macOS. Check with Broadcom Carbon Black Support for any updates about other sensors that can generate this hash type.

SHA-256 hashes are reported in addition to MD5 hashes. They can be used to report information to the Event Forwarder (v3.4.0 or later) and are also displayed on relevant pages in the console. See https://github.com/carbonblack/cb-event-forwarder for information on installing and configuring the Event Forwarder. See Chapter 16 Event Forwarder.

Sensor Group Feature Support

The following table describes whether certain features can be configured for Carbon Black EDR sensor groups.

Feature	Windows	Linux	macOS
Alerts Critical Level	Yes	Yes	Yes
Banning Settings	Yes	Yes	Yes
Binaries (Enable/Disable)	Yes	Yes	Yes
Binary Info (Enable/Disable)	Yes	Yes	Yes
BinaryModule loads (Enable/Disable)	Yes	Yes	Yes
Child Process events (Enable/Disable)	Yes	Yes	Yes
Cross Process events (Enable/Disable)	Yes	N/A	N/A
Retention Maximization (Enable/Disable)	Yes	No	Yes
EMET events (Enable/Disable)	Yes	N/A	N/A
File Modifications (Enable/Disable)	Yes	Yes	Yes
Known DLLs (Dylib/Mac) Filtering (Enable/Disable)	Yes	No	Yes
Network Connections (Enable/Disable)	Yes	Yes	Yes
Non-Binary File Writes (Enable/Disable)	Yes	No	No
Process Information (Enable/Disable)	Yes	Yes	Yes

Feature	Windows	Linux	macOS
Process user context (Enable/Disable)	Yes	Yes	Yes
Registry modifications (Enable/Disable)	Yes	N/A	N/A
Sensor Name	Yes	No	No
Sensor Network Throttling	Yes	No	Yes
Sensor Upgrade Policy	Yes	Yes	Yes
Sensor-side Max Disk Usage (%)	Yes	Yes	Yes
Sensor-side Max Disk Usage (MB)	Yes	Yes	Yes
Server TLS certificate swapping (choose cert)	Yes	No	Yes
Server TLS strict certificate validation	Yes	No	Yes
Tamper Level Settings	Yes	N/A	N/A
VDI Behavior Enabled	Yes	Yes	Yes

Sensor Health Score Messages 24

This section describes sensor health score messages that display on the Sensor Details page in the Carbon Black EDR console.

Sensors are discussed in Chapter 5 Managing Sensors. For information about installing, updating, troubleshooting, or uninstalling sensors, see the *Carbon Black EDR Sensor Installation Guide*.

Sensor health scores are generated by using a variety of inputs. The default score for a sensor that is running without any issues is 100. Carbon Black subtracts points from this score for events that fall outside of the “healthy range”, based on severity. Sensor health score messages are provided in the Carbon Black EDR console when the sensor is in an unhealthy state.

Health events are presented in priority order. If two events occur at the same time, the message for the higher priority event is presented, regardless of the severity. The sensor can only report one message at a time even when multiple messages occur. The last message type that is processed by the sensor is the one that is reported to the Carbon Black EDR server.

The priority order for each sensor type is listed in the following applicable sections.

Read the following topics next:

- Windows Health Events
- macOS Health Events
- Linux Health Events

Windows Health Events

This section describes Windows sensor health score messages that display on the Sensor Details page in the Carbon Black EDR console.

Driver and Component Failures

This topic describes Carbon Black EDR Windows sensor driver and component failures.

Cause

This alert occurs if Netmon, Svc component, or core drivers fail to load.

Impact

The sensor does not collect netconn events if the Netmon driver fails. The sensor can stop collecting one or more event types if Svc component fails. The sensor does not collect any events if the core driver fails.

Severity Scale

Driver failure	Health score	Message
Svc component	-25	Svc component failure
Netmon driver	-25	NetMon stream driver failure
Core driver	-100	Core driver failure

Remediation

Restart the failed service. For Netmon issues, a system reboot and re-installation of the network driver might be necessary if issues persist. Contact Carbon Black Technical Support if issues continue.

Memory Usage (Windows)

This topic describes Carbon Black EDR Windows sensor memory usage.

Cause

Carbon Black EDR sensor service memory usage has risen above expected values.

Impact

Excessive memory consumption can impact system and application performance.

Severity Scale

Memory (MB)	Health score	Message
> 50	-5	Elevated memory usage
> 100	-10	Elevated memory usage
> 200	-20	High memory usage
> 512	-25	Very high memory usage
> 1024	-50	Excessive high memory usage

Remediation

Restart service. Contact Carbon Black Technical Support if issues continue.

GDI Handle Count

This metric records GDI handles usage from the Carbon Black EDR Windows sensor service. GDI handles are used in module extraction only.

Cause

Carbon Black EDR sensor service GDI handle usage is above normal values.

Severity Scale

GDI handles	Health score	Message
> 100	-5	High GDI handle count
> 500	-10	Very high GDI handle count
> 1000	-20	Excessive GDI handle count

Remediation

Analyze event collection to see if a specific event type is generating an excessive count. If these are non-binary file writes, this collection type can be often be turned off. See Turn off Event Collection of Non-binary File Writes.

Handle Count

This metric records kernel handles usage from the sensor service. This metric does not include GDI (Graphics Device Interface) or user handles. Sensors that are running on Windows XP x86 do not report this metric.

Cause

Carbon Black EDR sensor service kernel handle usage is above normal values.

Severity Scale

Handles	Health score	Message
> 500	-5	Elevated handle count
> 1000	-10	High handle count
> 2000	-25	Very high handle count
> 4000	-50	Excessive handle count

Remediation

Analyze event collection to see if a specific event type is generating an excessive count. If these are non-binary file writes, this collection type can be often be turned off. See Turn off Event Collection of Non-binary File Writes.

Disk Space

This topic describes Carbon Black EDR Windows sensor disk space.

Cause

The free disk space on the volume where the Carbon Black EDR sensor is installed has dropped below normal values. This metric does not consider available disk space on other system disks.

Impact

Data collection/usability.

Severity Scale

Disk space (MB)	Health score	Message
< 1024	-5	Low disk space
< 100	-25	Very low disk space
< 10	-50	Excessively low disk space

Remediation

Run utilities to clear disk space.

Event Loss

This topic describes Carbon Black EDR Windows sensor event loss.

Cause

Events are dropped by the kernel driver due to high activity or component malfunction, which is calculated by the percentage of total kernel events that were dropped.

Impact

Potential impact to data collection.

Severity Scale

Event loss (%)	Health score	Message
1	-5	Elevated event loss
2	-10	High event loss
5	-20	Very high event loss
10	-50	Excessive event loss

Remediation

Restarting the service should resolve the issue.

Event Load

This topic describes Carbon Black EDR Windows sensor event loads.

Cause

The number of outstanding raw kernel events to be processed has exceeded a threshold.

Note Netconn events are handled in a separate driver.

Impact

Data collection/usability.

Severity Scale

Event queue depth	Health score	Message
> 512	-5	Elevated event load
>1024	-10	High event load
> 4096	-25	Excessive event load

Remediation

Analyze event collection to determine what is generating the event load. Consider disabling event collection on certain event types.

macOS Health Events

This section describes macOS sensor health score messages that display on the Sensor Details page in the Carbon Black EDR console.

Memory Usage (macOS)

This topic describes Carbon Black EDR macOS sensor memory usage.

Cause

Carbon Black EDR sensor service memory usage has risen above expected values.

Impact

System stability and performance can be impacted if abnormal memory usage persists.

Severity Scale

Memory (MB)	Health score	Message
> 100	-10	Elevated memory usage
> 250	-20	High memory usage
> 512	-25	Very high memory usage
> 1024	-50	Excessive memory usage

Remediation

Restart service. Contact Carbon Black Technical Support if issues continue.

Out of License (macOS)

This topic describes a situation where a Carbon Black EDR server is out of license.

Cause

Carbon Black EDR server license is expired.

Impact

The sensor is currently unable to push data to the server. Event data is cached on the endpoint. Attempts to send data can cause elevated bandwidth consumption.

Severity Scale

Condition	Health score	Message
Expired license	-25	Out of License

Remediation

Apply updated license to the Carbon Black EDR server.

Upgrade Issue

This topic describes a Carbon Black EDR macOS upgrade issue.

Cause

An upgrade issue exists that is probably due to an unapproved kext or a required restart at the endpoint to complete the upgrade.

Impact

Inoperable until the condition is resolved.

Severity Scale

Condition	Health score	Message
Upgrade incomplete	-75	Endpoint must be restarted to complete upgrade
Upgrade failed	-75	Carbon Black EDR kernel extensions are not approved for load

Remediation

Check kext status and approve if necessary for upgrade failure condition. Contact Carbon Black Technical Support if issues continue.

Proxy Driver Failure

This topic describes a Carbon Black EDR macOS proxy driver failure.

Cause

Probable cause is an OS kernel major version mismatch with the proxy driver.

Impact

Sensor does not collect process events correctly because the proxy driver is not connected to OS sys tables.

Severity Scale

Condition	Health score	Message
Driver fails to load	-25	Proxy driver failure

Remediation

Validate that the kernel version is supported by the sensor. If the OS version is supported, restart the service. Contact Carbon Black Technical Support if issues continue.

Procmon Driver

This topic describes a Carbon Black EDR macOS sensor procman driver issue.

Cause

Issue with loading procmon (process monitoring) driver, or version mismatch from a failed upgrade.

Impact

The sensor might stop collecting one or more data collection types.

Severity Scale

Condition	Health score	Message
Version does not match sensor version	-37	Procmon driver version mismatch
Driver fails to load	-37	Procmon driver failure

Remediation

Restart service. Contact Carbon Black Technical Support if issues continue.

Netmon Driver

This topic describes a Carbon Black EDR macOS sensor netmon driver issue.

Cause

There is an issue with loading netmon (network monitoring) driver, or a version mismatch from a failed upgrade.

Impact

The sensor might stop collecting netconn events.

Severity Scale

Condition	Health score	Message
Version does not match sensor version	-37	Netmon driver version mismatch
Driver fails to load	-37	Netmon driver failure

Remediation

Restart service. Contact Carbon Black Technical Support if issues continue.

Linux Health Events

This section describes Linux sensor health score messages that display on the Sensor Details page in the Carbon Black EDR console.

Out of License (Linux)

This topic describes a situation where a Carbon Black EDR server is out of license.

Cause

Carbon Black EDR server license is expired.

Impact

The sensor is currently unable to push data to the server. Event data is cached on the endpoint. Attempts to send data can cause elevated bandwidth consumption.

Severity Scale

Condition	Health score	Message
Expired license	-25	Out of License

Remediation

Apply updated license to the Carbon Black EDR server.

Failed to get Event log Stats

This topic describes a Carbon Black EDR Linux sensor procman driver issue.

Cause

There is an issue loading a procmon driver, or there is a version mismatch from a failed upgrade.

Impact

The sensor cannot track the current event log.

Severity Scale

Condition	Health score	Message
Cannot determine current event log stats	-50	Failed to get Event log stats

Remediation

Restart CB daemon. Contact Carbon Black Technical Support if issues continue.

Driver Failure

This topic describes a Carbon Black EDR Linux sensor driver failure.

Cause

An issue occurred when loading a driver.

Impact

The sensor might stop collecting one or more data collection types.

Severity Scale

Condition	Health score	Message
Driver failure	-50	Driver failure

Remediation

Restart CB daemon. Contact Carbon Black Technical Support if issues continue.

Memory Usage (Linux)

This topic describes Carbon Black EDR Linux sensor memory usage.

Cause

Carbon Black EDR sensor daemon memory usage memory usage is above normal values.

Impact

System stability and performance can be impacted if abnormal memory usage persists.

Severity Scale

Memory (MB)	Health score	Message
> 75	-5	Elevated memory usage
> 100	-10	Elevated memory usage
> 250	-20	High memory usage
> 300	-25	Very high memory usage
> 450	-50	Excessive memory usage

Remediation

Restart CB daemon. Contact Carbon Black Technical Support if issues continue.

Document History

25

The following changes were made to this document:

Date	Associated Software Version	Topic	Change Description
12 April 2024	N/A	All	Updated product names
02 October 2023	N/A	Copyrights and Notices	Removed broken link

Date	Associated Software Version	Topic	Change Description
16 July 2023	7.8.0	■ Exclusion Settings ■ Add Exclusion Settings to the Sensor Group Panel ■ Overview of Sensor Management ■ Server-Sensor Certificate Requirements ■ Chapter 19 Watchlists ■ Vitals and Configuration ■ Search for Events ■ Process Event Search and child topics ■ Chapter 15 Antimalware Scan Interface and child topics	■ Added note about inheritance ■ Removed macOS limitation ■ Added content regarding the importance of accurate time keeping through NTP ■ Explanation of why duplicate SANs are not allowed; FIPS certificate validations ■ Clarification of severity scoring ■ Corrected `Sensor Uptime` and `Host Uptime` descriptions ■ Added a topic about searching for events on the Process Search page ■ Added a section about searching for process event results on the Process Analysis page ■ Added Antimalware Scan Interface section. Removed *PowerShell script block logging must be enabled* requirement. Added Fileless script load events to events that can be forwarded.
23 September 2022	7.7.1	Copyrights and Notices	Fixed Product Version
18 September 2022	7.7.1	■ Event Forwarder Syslog Output ■ Chapter 3 Managing User Accounts (Carbon Black EDR) ■ Tamper Protection of Windows Sensors ■ Turn off Event Collection of Non-binary File Writes	■ Updated Syslog Destination format ■ Corrected Step 1 navigation for viewing user activity ■ Clarified note about disabling Carbon Black App Control Tamper Protection ■ Added topic

Date	Associated Software Version	Topic	Change Description
18 July 2022	N/A	■ Copyrights and Notices ■ Creating and Running a Query ■ Query Results	Fixed broken links
14 July 2022	7.7.0	All	Started the document history