



Carbon Black Server Configuration File (cb.conf)

Carbon Black Version 5.0.0

3 August 2015

Bit9, Inc.

1100 Winter Street, Waltham, MA 02451 USA

Tel: 617.393.7400 Fax: 617.393.7499

E-mail: support@bit9.com

Web: <http://www.bit9.com>

Copyright © 2004-2015 Bit9, Inc. All rights reserved. This product may be covered under one or more patents pending. Bit9 and Carbon Black are trademarks of Bit9, Inc. in the United States and other countries. Any other trademarks and product names used herein may be the trademarks of their respective owners.

Contents

Overview	5
Changes on Upgrade.....	5
Format Guidelines	5
Configuration Settings	6
Data Storage Settings	6
DatastoreRootDir	6
AllianceClientStorefilePurgeMax	6
AllianceClientNoStorefileDelete	6
EnableSolrBinaryInfoNotifications	7
EnableSolrFeedNotifications.....	7
CbSolrConnectionTimeout	7
CbSolrSocketTimeout.....	7
KeepAllModuleFiles	7
MaxEventStoreSizeInDocs	7
MaxEventStoreDays.....	8
MaxEventStoreSizeInMB.....	8
MaxEventStoreSizeInPercent.....	8
MinAvailableSizeInMB	8
ProcessDocumentSplitThreshold.....	8
WatchlistEndTimeOffset	8
WatchlistStartTimeOffset	9
Communication Settings	9
AllowNullSensorHostRegister	9
AllianceNoClientCert.....	9
AllianceVerifyServerCert.....	10
EnforceClientCerts	10
Network Settings	10
CoreServicesIP.....	10
CoreServicesPort.....	10
DatastorePort.....	11
DatastoreIP	11
NginxSensorHttpPort	11
NginxWebApiHttpPort	11
RedisHost	11
RedisPort.....	11
RedisStatsHost	11
RedisStatsPort.....	11
SolrIP	11
SolrPort	12
WebsocketPort	12
SSL Certificate Usage	12
SSLCertFile.....	12
SSLKeyFile.....	13

AllianceCert	13
AllianceCertKey	13
Cb Internal Settings	14
CbUser	14
CbGroup	14
CbFileDescriptorLimit	14
CbLicenseFile	14
CbServerTokenFile	14
CbJavaHome	15
CoreServicesSmallScaleSensorCount	15
CoreServicesMaxCheckinInterval	15
CoreServicesProcessSearchIntervalSeconds	15
CoreServicesEnableFuzzyProcessFacets	15
CoreServicesEnableFuzzyBinaryFacets	15
CoreServicesProcessSearchOrder	16
CoreServicesBinarySearchOrder	16
CoreServicesProcessPageSize	16
CoreServicesBinaryPageSize	16
CoreServicesProcessAutocomplete	16
CoreServicesBinaryAutocomplete	16
TimestampDeltaThreshold	17
CoreServicesPidFile	17
SensorInstallerDir	17
FlaskSecret	17
FailedLogonLockoutCount	17
AccountUnlockInterval	17
UserActivityQuota	17
UserActivityQuotaDelta	17
AllianceClientPidFile	18
AllianceSyncIntervalSecs	18
AllianceURL	18
DatastoreJvmMax	18
DatastoreAllowUnregisteredSensor	18
DatastoreShutdownTimeout	18
DatastoreDisableJMXRemote	18
DisableDatastoreCache	19
EventStoreSolrCore	19
ModInfoStoreSolrCore	19
ModInfoStoreFlushInterval	19
PgSqlDataDir	19
PgSqlPidFile	19
PgSqlLogfilePath	19
PgSqlPort	19
DatabaseURL	19
ModstorePath	20
RedisPidFile	20
Syslog Template Settings	20
WatchlistSyslogTemplateProcess	20

WatchlistSyslogTemplateBinary	20
BinaryInfoSyslogTemplateObserved.....	20
BinaryInfoSyslogTemplateGroupObserved.....	20
BinaryInfoSyslogTemplateHostObserved	21
Contacting Carbon Black Support	21

Overview

The primary configuration file for the Carbon Black Enterprise server is:

```
/etc/cb/cb.conf
```

In a standard production environment, there should be no need to modify this file directly – most configuration options are either set during installation or via the Carbon Black console. However, the configuration options described here may be useful for troubleshooting issues with the server, customizing the configuration for local integration, or making other customizations not available through the console interface.

Changes on Upgrade

When you install the Carbon Black Enterprise server the first time and run `cbinit`, this command auto-generates the `/etc/cb/cb.conf` file from a template, providing the standard parameters and default settings for that version of the server. As new server versions are released, `cb.conf` parameters may be added or the defaults changed. To avoid overwriting your own customizations, `cb.conf` is *not* programmatically updated when you upgrade the Carbon Black server. Instead, a PDF file of the current properties in `cb.conf` is included with the new release. This allows you to compare your version with the newest version, and copy over the new properties.

Format Guidelines

The configuration file is consumed by Carbon Black services as well as the Bash shell on your server. It includes two types of content: property settings and comments. If you edit the file, follow these formatting rules closely to avoid parsing errors:

- Any line starting with `#` is considered to be a comment.
- Place comments on their own line. Comments should never be added to the end of a line that contains a property setting.
- Define all properties strictly as `name=value` pairs.

Important

There must not be any whitespace (spaces or tabs) around the equals (=) sign in any of the `cb.conf` settings. For example:

Properly formed line:

```
name=value
```

Not properly formed and will not be processed correctly:

```
name =value
```

```
name= value
```

```
name = value
```

Configuration Settings

Data Storage Settings

DatastoreRootDir

Default: `/var/cb/data`

Sets the path to the root directory where *movable* runtime data for the Carbon Black Enterprise Server is stored. This data includes Solr, PostgreSQL, and flat-file storage of module files. Each of these storage types has additional parameters, which are described in this document.

Notes

- Consult Carbon Black Support if you want to move your data root directory from one volume to another.
- The parent directory `/var/cb` contains all runtime data for the server, but some of this data is not movable and therefore not in `DatastoreRootDir`.

AllianceClientStorefilePurgeMax

Default: 100

Specifies the maximum number of storage files (binaries uploaded from sensors to the server) that the Carbon Black Alliance client purges from a local hard drive if it determines that the files should not be stored locally (for example, if the files were already uploaded to a Carbon Black Alliance server and so shouldn't take up unnecessary space).

AllianceClientNoStorefileDelete

Default: 0

Specifies whether Carbon Black Alliance client keep binary files locally after they have been uploaded to the central Carbon Black Alliance server. If this is set to 0, Carbon Black Alliance clients delete binary storefiles after uploading them to preserve local hard drive space. If set to 1, binary modules are not deleted after they have been uploaded.

Warning

The Purge script still erases binary files to recover disk space unless `KeepAllModuleFiles` has been set to 1.

You can always download binary files from the Carbon Black Alliance server, even if the files have been deleted. Download attempts through the console will search the Alliance server if no local copy of a file is available, and the Carbon Black API also provides access to Alliance server downloads.

EnableSolrBinaryInfoNotifications

Default: False

When set to `True`, this parameter enables notifications for new binaries, new hosts, and sensor groups that are observing a particular binary. One notification event occurs for *each* of the following cases: the binary is completely new on sensors reporting to this server, it is new to the host that reported it, it is new to the sensor group to which the reporting host belongs – this means that one newly discovered binary can trigger up to three notifications. Notifications are sent to syslog as log messages.

EnableSolrFeedNotifications

Default: True

When set to `True`, this parameter enables notifications when documents that have feed hits are committed. Notifications are sent to the Carbon Black console UI as alerts, and to syslog as log messages.

CbSolrConnectionTimeout

Default: 0

Sets the connection timeout from the datastore to the Solr backend, in milliseconds. If the internal defaults for the Solr client are in use, this value is 0.

CbSolrSocketTimeout

Default: 0

Sets the socket read timeout from the datastore to the Solr backend, in milliseconds. If the internal defaults for the Solr client are in use, this value is 0.

KeepAllModuleFiles

Default: 0

Sets the timing for deleting the binary files uploaded from sensors. A default value of 0 indicates that these files are erased at these times:

- After they are uploaded to the Carbon Black Alliance server.
- When data is purged to free up storage volume.

Changing this value to 1 sets the server to **never** delete module files.

MaxEventStoreSizeInDocs

Default: 120

Sets the threshold of the process document count (in millions) that triggers clean-up. This parameter takes precedence over all other storage-size parameters.

MaxEventStoreDays

Default: n/a

By default, process data is purged automatically when disk space is required. If this value is set, any process with a `last_server_update` time older than the number of days defined in this value is deleted.

MaxEventStoreSizeInMB

Default: n/a

By default, process data is purged automatically when disk space is required. If this value is set, process data is deleted, starting from the earliest date, until the size of the process store is less than this value.

MaxEventStoreSizeInPercent

Default: 50

Sets the threshold size of disk usage for which cleanup is triggered as a percentage of the total disk space that is available to the event store. The total disk space that is available to the event store is calculated as the sum of the current event store size and free disk space.

MinAvailableSizeInMB

Default: n/a

This parameter is optional. It can be used to set a hard limit on the size of available disk space that has to be maintained on the mount point where the event store resides. This parameter takes precedence over all other storage-size parameters, except for `MaxEventStoreSizeInDocs`.

ProcessDocumentSplitThreshold

Default: 10000

Sets the total number of events after which a process document starts splitting into multiple segments.

WatchlistEndTimeOffset

Default: 0MINUTES

Changes the search window end-time offsets for watchlist search jobs.

Notes

- The Watchlist parameters are optimized based on the commit interval of the Solr backend. Contact Carbon Black Technical Support before you change these values.
- Available units are SECONDS, MINUTES, HOURS, DAYS, and YEARS.
- Note that there should be no space between the numeric value and the unit label, as shown above.

WatchlistStartTimeOffset

Default: 12MINUTES

Changes the search window start-time offsets for watchlist search jobs. The Notes for WatchlistEndTime Offset (above) also apply to this parameter.

Communication Settings

These settings adjust the communications between the Carbon Black Enterprise server and other components in the Carbon Black environment (i.e., sensors and the Carbon Black Alliance server).

AllowNullSensorHostRegister

Default: 0

During initial sensor registration, the Carbon Black Enterprise server requires the sensor computer's Security Identifier (SID). If this value is blank, the server rejects the registration. If the server rejects the sensor registration, the sensor re-attempts registration in a few minutes, which includes another attempt to get the sensor computer SID.

If the condition that prevents the sensor from getting the SID is temporary and has a short duration, the condition will fix itself. If the condition is chronic, set this value to 0 to allow the sensor to register with an empty SID. Sensors that the Carbon Black Enterprise server rejects for an empty SID are logged in `/var/log/cb/coreservices/debug.log`.

AllianceNoClientCert

Default: 0

The Carbon Black Alliance server uses SSL client certificates to authenticate communication with Carbon Black Enterprise servers. Most SSL inspection devices do not support client certificates and immediately end the connection when they receive a client certificate.

Set this parameter to 1 to prevent transmission of the SSL client certificate.

Note

Contact Bit9 + Carbon Black Technical Support for alternate authentication arrangements.

AllianceVerifyServerCert

Default: 1

Indicates that the Carbon Black Alliance server's SSL certificate must be validated with the Carbon Black Certificate Authority. If the server's SSL certificate was not signed by the Carbon Black Certificate Authority, the connection fails. If your network uses an SSL inspection device, this parameter must be disabled.

EnforceClientCerts

Default: True

Carbon Black sensors validate servers by using SSL server certificates. The Carbon Black Enterprise server also validates sensors by using SSL client certificates. This setting specifies whether the Carbon Black Enterprise server allows sensors that do not provide an SSL certificate to communicate with it.

This value should generally be set as `True`, but can be disabled for troubleshooting, addressing mismatched certificates or upgrading older sensors (pre v3.1.0) that did not support SSL client certificates.

Network Settings

Review, update, or modify these settings to adjust the Carbon Black Enterprise server listener IP addresses and ports.

CoreServicesIP

Default: [: :]

The `coreservices` daemon binds to this interface. This parameter allows you to specify an option that makes sense in your environment, such as:

- 127.0.0.1 – listen on local IPv4 loopback interface
- [::1] – listen on local IPv6 loopback interface
- 127.0.0.1|[::1] – listen on IPv4 AND IPv6 loopback interfaces

CoreServicesPort

Default: 5000

The `coreservices` daemon binds to the port whose number is set in this parameter.

DatastorePort

Default: 9000

`cbfs-http` binds to this port.

DatastoreIP

Default: 127.0.0.1

`cbfs-http` binds to this interface.

NginxSensorHttpPort

Default: 443

Nginx maintains its own configuration files. However, this property must be kept in sync with the configuration of the `listen` directive in `/etc/cb/nginx/conf.d/cb.conf` so that other components (such as firewall management) know which ports are used for HTTP communications.

NginxWebApiHttpPort

Default: 443

See notes for `NginxSensorHttpPort` for more details regarding this property.

RedisHost

Default: localhost

Sets the Redis general cache host.

RedisPort

Default: 6379

Sets the Redis general cache listener port (TCP).

RedisStatsHost

Default: localhost

Sets the Redis statistics cache host.

RedisStatsPort

Default: 6379

Sets the Redis statistics cache listener port (TCP).

SolrIP

Default: 127.0.0.1

Sets the network binding IP address for the `cb-solr` service.

SolrPort

Default: 8080

Sets the binding between the cb-solr service and the specified port. This identifies the HTTP port that is used for all external communications, which includes sensors as well as the Carbon Black console UI.

Note

If this value is modified, you must also update the file `/etc/nginx/conf.d/cb.conf` with the same modification.

WebsocketPort

Default: 5006

Sets the websocket daemon to bind with this port.

SSL Certificate Usage

Carbon Black uses SSL certificates in the following ways:

- Sensors use SSL server certificates to validate that they are communicating with the correct Carbon Black Enterprise server.
- The Carbon Black Enterprise server uses SSL client certificates to validate that it is communicating with an authentic sensor.
- The Carbon Black Enterprise server uses the SSL server certificate to validate that it is communicating with the correct Carbon Black Alliance server.
- The Carbon Black Alliance server uses SSL client certificates to validate that it is communicating with an authentic Carbon Black Enterprise server.

The following sections describe SSL certificate configuration.

SSLCertFile

Default: `/etc/cb/certs/cb-server.crt`

Sets the location of SSL certificate files that are used for HTTPS communications between sensors and the Carbon Black server.

Note

If these paths are modified, a corresponding change must also be made in the `/etc/nginx/conf.d/cb.conf` file.

These certificates are generated during `cbinit` and are unique for each Carbon Black server.

SSLKeyFile

Default: `/etc/cb/certs/cb-server.key`

Sets the location of SSL private key files that are used for HTTPS communications between sensors and the Carbon Black server.

Note

If these paths are modified, a corresponding change must also be made in the `/etc/nginx/conf.d/cb.conf` file.

These certificates are generated during `cbinit` and are unique for each Carbon Black server.

AllianceCert

Default: `/etc/cb/certs/carbonblack-alliance-client.crt`

Sets SSL certificate files that are used for client-side authentication when an HTTPS connection with a Carbon Black Alliance server is established. These files are loaded onto the machine when the Carbon Black Release RPM is installed. The files are used whenever the Carbon Black Enterprise server needs to communicate with central Carbon Black servers. This includes yum repositories for installing and upgrading the Carbon Black Enterprise server software as well as the Carbon Black Alliance client service.

Note

These Carbon Black certificates are specific to each customer organization and should be treated with care. Do not share them with other organizations or people outside your company.

AllianceCertKey

Default: `/etc/cb/certs/carbonblack-alliance-client.key`

Sets SSL private key files that are used for client-side authentication when an HTTPS connection with a Carbon Black Alliance server is established. These files are loaded onto the machine when the Carbon Black Release RPM is installed. The files are used whenever your Carbon Black Enterprise server needs to communicate with central servers at Carbon Black. This includes yum repositories for installing and upgrading the Carbon Black Enterprise server software as well as the Carbon Black Alliance client service.

Note

These Carbon Black certificates are specific to each customer organization and should be treated with care. Do not share them with other organizations or people outside your company.

Cb Internal Settings

You are unlikely to need to modify these settings, and in many cases *should not* modify them without specific instructions from Carbon Black Technical Support. However, they may provide valuable information for troubleshooting.

CbUser

Default: `cb`

This setting controls the user that the Carbon Black services run as. The `cb` user is created during RPM installation. To use another user, create the user, and then restart the Carbon Black Enterprise server (`cb-enterprise`).

CbGroup

Default: `cb`

This setting controls the Linux group that the Carbon Black services run as. The `cb` sensor group is created during RPM installation. To use another sensor group, create the group in Linux, update this value, and then restart the Carbon Black Enterprise server (`cb-enterprise`).

CbFileDescriptorLimit

Default: `80000`

By default, CentOS allows only 1024 file descriptors per process. This number is too low for Carbon Black. Carbon Black updates the process file descriptor limit in the `cb-enterprise` `init` script to the default value with `ulimit -n`.

CbLicenseFile

Default: `/etc/cb/server.lic`

The path to the Carbon Black Enterprise server license file. This path is provided by Carbon Black Technical Support and should not be modified unless done so in conjunction with a support representative.

CbServerTokenFile

Default: `/etc/cb/server.token`

A random hexadecimal string used to uniquely identify this Carbon Black server installation.

CbJavaHome

Default: /usr/lib/jvm/jre-1.7.0-openjdk.x86_64/

Carbon Black requires JRE version 1.7.0 or later. If the JRE is installed at a different location on your server, change this value to reflect the correct location.

CoreServicesSmallScaleSensorCount

Default: 5

If the number of sensors that are currently active is less than this value, the sensor check-in interval is always 30 seconds. If it is greater, Carbon Black calculates a dynamic check-in interval.

CoreServicesMaxCheckinInterval

Default: 1335

Configures the maximum interval, in seconds, between successive sensor check-ins from a single sensor. Raising this value decreases load on the server, as there are fewer sensor check-ins and fewer modifications to the event store.

CoreServicesProcessSearchIntervalSeconds

No Default

Limits the length of time for all process searches in the Carbon Black console UI to the most recent number of seconds specified in this setting. This applies to both process searches in the Search Processes page and process watchlists from the Watchlists page.

Note

This setting only applies to the Carbon Black console UI. Direct API queries do not honor this setting.

CoreServicesEnableFuzzyProcessFacets

Default: True

This setting enables the use of statistical sampling for calculating the terms in facets. This provides significantly improved runtime performance and reduced memory usage.

CoreServicesEnableFuzzyBinaryFacets

Default: True

This setting enables the use of statistical sampling for calculating the terms in facets. This provides significantly improved runtime performance and reduced memory usage.

CoreServicesProcessSearchOrder

Default: start desc

Sets the sort order of process search results as seen in the Carbon Black console UI. The format of this field is: `fieldnamedirection`, where `direction` is either `asc` (for ascending) or `desc` (for descending).

CoreServicesBinarySearchOrder

Default: server_added_timestamp desc

Sets the sort order of binary search results as seen in the Carbon Black console UI. The format of this field is: `fieldnamedirection` where `direction` is either `asc` (for ascending) or `desc` (for descending).

CoreServicesProcessPageSize

Default: 10

Sets the number of matching process documents that display on each page as seen in the Search Processes page in the Carbon Black console UI.

CoreServicesBinaryPageSize

Default: 10

Sets the number of matching binary documents that display on each page as seen in the Search Binaries page in the Carbon Black console UI.

CoreServicesProcessAutocomplete

Default: Suggester

Sets the backend method for the autocomplete function for search queries entered in the Search Processes page. Valid values are:

- `Suggester: Faster` This value does not include counts or infrequent terms.
- `Terms: Slower` This value includes counts and all terms.

CoreServicesBinaryAutocomplete

Default: Terms

Sets the backend method for the autocomplete function for search queries entered in the Search Binaries page. Valid values are:

- `Suggester: Faster` This value does not include counts or infrequent terms.
- `Terms: Slower` This value includes counts and all terms.

TimestampDeltaThreshold

Default: 5

Sets the time (in seconds) that is used as a threshold for identifying sensors with unsynchronized clocks.

CoreServicesPidFile

Default: /var/run/cb/coreservices.pid

Contains the current process ID of the `coreservices` daemon.

SensorInstallerDir

Default: /usr/share/cb/coreservices/installers

The main directory of sensor installers. The contents of this directory are loaded by `coreservices` at startup and are used for sensor versions, including the definition of `latest` if sensors are configured to be automatically upgraded to the latest version.

FlaskSecret

Default: none

This required value is a random string of ASCII-printable characters. It is unique for each server and auto-generated during `cbinit`. It is used to encrypt session cookies that are used after a user authenticates with the Carbon Black console UI.

FailedLogonLockoutCount

Default: 10

Sets the number of times a user can fail authentication before the account is locked.

AccountUnlockInterval

Default: 30

Sets the number of minutes after which a locked account unlocks.

UserActivityQuota

Default: 10000

Carbon Black logs all user authentication in the PostgreSQL database. This setting defines the *minimum* number of authentication records that are kept.

UserActivityQuotaDelta

Default: .1

Defines when to start trimming the number of user authentication records. It is a percentage of `UserActivityQuota`.

For example, if `UserActivityQuota` is set to 10000 and `UserActivityQuotaDelta` is set to .1, the database grows to 11000 user authentication records. When the number of records reaches 11000, it is reduced to 10000.

AllianceClientPidFile

Default: `/var/run/cb/allianceclient.pid`

Sets the path to the PID file that is used for `cb-allianceclient` service control.

AllianceSyncIntervalSecs

Default: 60

Sets the number of seconds between periodic connection attempts to the Carbon Black Alliance server.

AllianceURL

Default: `https://api.alliance.carbonblack.com`

Sets the URL of the Carbon Black Alliance server.

DatastoreJvmMax

Default: 20%

Sets the maximum amount of RAM to be used for JVM's memory heap. This parameter can be specified either as a number of megabytes (for example, 4096) or as a percentage of the host machine's physical RAM by appending '%' on the end (for example, 30%).

DatastoreAllowUnregisteredSensor

Default: 0

Controls whether the datastore accepts data from a sensor that has not been registered with a Carbon Black Enterprise server. The default of zero disables this capability, and there is generally no reason to enable it.

DatastoreShutdownTimeout

Default: 60

Sets the number of seconds to wait, when the datastore is being stopped, for all buffers and cached data to be cleanly written to disk. After this time, if the service is still running, it is forcibly stopped.

DatastoreDisableJMXRemote

Default: 0

JMXRemote allows an external Java management or debugging process on local machine to communicate with the datastore. If this setting is not 0, the datastore process is launched without JMXRemote.

DisableDatastoreCache

Default: `False`

Disables Ehcache in the datastore and forces all process events to be pushed immediately to the Solr engine.

EventStoreSolrCore

Default: `cbevents`

Sets the name of the Solr core to be used for process data.

ModInfoStoreSolrCore

Default: `cbmodules`

Sets the name of the Solr core to be used for module information storage.

ModInfoStoreFlushInterval

Default: `1000`

Sets the time interval, in milliseconds, with which buffered module information events are pushed to the module information Solr core.

PgSqlDataDir

Default: `/var/cb/pgsql`

Sets the location of the PostgreSQL data directory.

PgSqlPidFile

Default: `/var/run/cb/cb-pgsql.pid`

Sets the path to the PID file, which is used for `cb-pgsql` service control.

PgSqlLogfilePath

Default: `/var/log/cb/pgsql/startup.log`

Sets the path to the `cb-pgsql` startup log file. This file captures output that is generated prior to the initialization of the logging framework.

PgSqlPort

Default: `5002`

Sets the port on which `cb-pgsql` listens.

DatabaseURL

Default: `postgresql+psycopg2://cb:(passwd)@localhost:5002/cb`

Sets the SQLAlchemy database URL that is used to connect with PostgreSQL.

ModstorePath

Default: /var/cb/modulestore

Sets the flat-file storage location for module file storage.

RedisPidFile

Default: /var/run/cb/cb-redis.pid

Sets the path to the PID file that is used for `cb-redis` service control. This file must be writable by `CbUser`.

Syslog Template Settings

WatchlistSyslogTemplateProcess

Sets the path to the [Jinja2 Template](#) that is used to format process watchlist hits before sending the data to syslog. Use `/usr/share/cb/cbsyslog` to modify and test this path. For information about templates and syslog, see Appendix F, “Syslog Output for Carbon Black Events,” in the *Carbon Black User Guide*.

If this option is not specified, the system default template is used. Use the `cbsyslog` tool to retrieve the system default template.

WatchlistSyslogTemplateBinary

Sets the path to the [Jinja2 Template](#) that is used to format binary watchlist hits before sending the data to syslog. Use `/usr/share/cb/cbsyslog` to modify and test this path. For information about templates and syslog, see Appendix F, “Syslog Output for Carbon Black Events,” in the *Carbon Black User Guide*.

If this option is not specified, the system default template is used. Use the `cbsyslog` tool to retrieve the system default template.

BinaryInfoSyslogTemplateObserved

Sets the path to the [Jinja2 Template](#) that is used to format binary information events before sending the data to syslog. These events are created the first time a binary, as identified by its MD5 hash value, is observed on any sensor that is associated with the Carbon Black Enterprise server. For more information, see Appendix F, “Syslog Output for Carbon Black Events,” in the *Carbon Black User Guide*, and Appendix E, “Carbon Black APIs,” in the *Carbon Black User Guide*.

If this option is not specified, the system default template is used. Use the `cbsyslog` tool to retrieve the system default template.

BinaryInfoSyslogTemplateGroupObserved

Sets the path to the [Jinja2 Template](#) that is used to format binary information for new sensor group events before sending the data to syslog. These events are created the first time a binary, as identified by its MD5 hash value, is observed by a new sensor group. For more information, see Appendix F, “Syslog Output for Carbon Black Events,” in the *Carbon Black User Guide*, and Appendix E, “Carbon Black APIs,” in the *Carbon Black User Guide*.

If this option is not specified, the system default template is used. Use the `cbsyslog` tool to retrieve the system default template.

BinaryInfoSyslogTemplateHostObserved

Sets the path to the [Jinja2 Template](#) that is used to format binary information for new host events before sending the data to syslog. These events are created the first time a binary, as identified by its MD5 hash value, is observed by a new sensor. For more information, see Appendix F, “Syslog Output for Carbon Black Events,” in the *Carbon Black User Guide*, and Appendix E, “Carbon Black APIs,” in the *Carbon Black User Guide*.

If this option is not specified, the system default template is used. Use the `cbsyslog` tool to retrieve the system default template.

Contacting Carbon Black Support

For your convenience, Bit9 + Carbon Black Technical Support offers several means of contact:

Technical Support Contact Options
Web: www.bit9.com
E-mail: support@bit9.com
Phone: 877.248.9098 (877.BIT9.098)
Fax: 617.393.7499
Hours: 8 a.m. to 8 p.m. EST

When you call or e-mail Bit9 + Carbon Black technical support, please provide the following information to the support representative:

Required Information	Description
Contact	Your name, company name, telephone number, and e-mail address
Product version	Product name (Bit9 Server, Bit9 Agent, or Bit9 Software Reputation Service) and version number
Hardware configuration	Hardware configuration of the Bit9 Server or computer (processor, memory, and RAM)
Document version	For documentation issues, specify the version of the manual you are using. The date and version of the document appear after the copyright section of each manual.
Problem	Action causing the problem, error message returned, and event log output (as appropriate)
Problem severity	Critical, serious, minor, or enhancement