



EMC® Watch4net®

Version 6.3u2

BroadSoft VoIP Collector Admin Guide

P/N: pending

Rev : 40032

EMC²

Copyright ©2010 - 2014 EMC Corporation. All rights reserved. Published in the USA.

Published April, 2014

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided as is. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC², EMC, and the EMC logo are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners.

For the most up-to-date regulatory document for your product line, go to EMC Online Support (<https://support.emc.com>).

Contents

Preface	iii
Purpose	iii
Intended Audience	iii
Prerequisites	iii
Document Organization	iv
Common Abbreviations and Acronyms	iv
Where to get help	v
1 Product Overview	1
1 CDR	1
2 Event	2
3 Timestamp	2
4 Fields	2
5 Errors	2
2 Installation	3
1 Dependencies	3
2 Frontend	4
3 Configuration	7
1 <i>Event Processing Manager</i> configuration	7
2 <i>BroadSoft VoIP Collector</i> configuration	8

Preface

Purpose

The *BroadSoft VoIP Collector* is a *Processing Element* designed to collect CDRs from the *BroadWorks Call Manager (CM)* in order to generate events. It is fully compatible with version 14.x of the *CM*.

Intended Audience

This guide is intended for advanced *APG* Administrators that have a good knowledge of the *APG Event Processing* framework.

Prerequisites

The *BroadSoft VoIP Collector* can only be used within an instance of the *Event Processing Manager*. Both components must be configured to work together.

Document Organization

1. Product Overview	Overview of the product.
2. Installation	Instillation instructions for the product.
3. Configuration	How to configure the <i>Event Processing Manager</i> and the <i>BroadSoft VoIP Collector</i> .

Common Abbreviations and Acronyms

CDR	Call Detail Record
CM	BroadWorks Call Manager
CSV	Comma-Separated Values
FTP	File Transfer Protocol
SFTP	SSH File Transfer Protocol
VoIP	Voice over Internet Protocol
XML	eXtensible Markup Language

Where to get help

EMC support, product, and licensing information can be obtained as follows:

PRODUCT INFORMATION - For documentation, release notes, software updates, or information about EMC products, go to EMC Online Support at:

<https://support.emc.com>

TECHNICAL SUPPORT - Go to EMC Online Support and click Service Center. You will see several options for contacting EMC Technical Support. Note that to open a service request, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

1

Product Overview

The *BroadSoft VoIP Collector* periodically downloads CDR files from a *Billing Server* which, in turn, receives these files from the CM. The collector then transforms the CDRs into *Events* which are sent to the next *Processing Element*.

1 CDR

A CDR is a CSV or XML file that contains data about VoIP calls. The data includes origin and destination phone numbers, the duration of calls, source and destination IP addresses, etc. With version 14.x of the CM, the CDR contains a maximum of 206 different data fields.

If a VoIP call occurs between two ends that are connected on the CM, then two CDRs are generated. One with the direction *terminating* and one with the direction *originating*. This could happen, for example, when a call is placed between two individuals at the same organization. To be able to link the two CDRs, the fields `localCallId` and `remoteCallId` must be used to match them.

CDRs are usually used for billing clients about their calls.

2 Event

Each *Event* is the result of one CDR. It is the events that are sent to the next *Processing Element*.

3 Timestamp

The timestamp field that is used by APG reports (frontend) is the `releaseTime`. Calls appearing for any particular day in a report are therefore calls that have been released that day. A call is released when it has been terminated.

For example, if a call starts at 11h50 PM on May 12 and ends at 12h05 AM on May 13, then the call appears on the May 13 report.

The duration of a call is the delta between its `answerTime` and `releaseTime`.

4 Fields

For a description of each field in a CDR file see the vendor's own documentation. In particular, it is recommended to see the *Accounting Call Detail Record* document corresponding to your version of the *BroadWorks Call Manager (CM)*.

5 Errors

When the *BroadSoft VoIP Collector* is unable to process a CDR, it will store it in the cache folder with the filename that the CDR originally came in, followed by a `.ignored` extension. All properly formed CDRs in the same file will be processed normally. When the *BroadSoft VoIP Collector* encounters a file that it cannot process for whatever reason (i.e. malformed header) it will store the entire file using the same procedure listed above.

Installation

1 Dependencies

The *BroadSoft VoIP Collector* can run on an APG-Datastore or on MySQL.

For the APG-Datastore, the following modules must be installed prior to installing the *BroadSoft VoIP Collector*:

- APG-Datastore
- APG-Datastore-Writer

For MySQL, the following modules must be installed prior to installing the *BroadSoft VoIP Collector*:

- MySQL
- Generic-Live-Writer

Note: The *BroadSoft VoIP Collector* copies some files into the directories of the prerequisite modules listed

above during its own installation, so it is important that these modules already be installed.

2 Frontend

After installing the collector, to be able to see the *CDR* in the frontend, you need to make some changes in the Tomcat configuration files.

Note: The syntax of the Tomcat files is documented fully in the Apache Tomcat documentation which you can refer to.

First you need to edit the **server.xml** file.

The path to this file is `<APG>/Web-Servers/Tomcat/Default/conf/server.xml`. You need to add the resource `FLOW-BROADSOFT-CDR` for your type of installation, just as indicated below.

```
...

<!-- For an installation with the APG Datastore. -->
<Resource name="jdbc/FLOW-BROADSOFT-CDR" auth="Container" type="javax.sql.DataSource"
  maxActive="30" maxIdle="5" validationQuery="SELECT_1" testOnBorrow="false"
  testWhileIdle="true" timeBetweenEvictionRunsMillis="10000"
  minEvictableIdleTimeMillis="60000" maxWait="10000" username="apg"
  password="watch4net" driverClassName="com.mysql.jdbc.Driver"
  removeAbandoned="true" removeAbandonedTimeout="60" logAbandoned="true"
  url="jdbc:mysql://localhost:53307/events?autoReconnect=true&tinyInt1isBit=false" />

<!-- For an installation with MySQL. -->
<Resource name="jdbc/FLOW-BROADSOFT-CDR" auth="Container" type="javax.sql.DataSource"
  maxActive="30" maxIdle="5" validationQuery="SELECT_1" testOnBorrow="false"
  testWhileIdle="true" timeBetweenEvictionRunsMillis="10000"
  minEvictableIdleTimeMillis="60000" maxWait="10000" username="apg"
  password="watch4net" driverClassName="com.mysql.jdbc.Driver"
  removeAbandoned="true" removeAbandonedTimeout="60" logAbandoned="true"
  url="jdbc:mysql://localhost:53306/events?autoReconnect=true&tinyInt1isBit=false" />
...
```

You then need to edit the **APG.xml** file.

The path to this file is: *<APG>/Web-Servers/Tomcat/Default/conf/Catalina/localhost/apg.xml*

You need to add the ResourceLink to the file just as indicated below.

```
...  
<ResourceLink name="jdbc/FLOW-BROADSOFT-CDR:{_mapping=FLOW-BROADSOFT-CDR.xml,_table=  
broadsoft_cdr_}"  
    global="jdbc/FLOW-BROADSOFT-CDR" type="javax.sql.DataSource" />  
...
```

Tip: *To view the data in a report, create a sub node with an 'Expand on every' pkid.*

Configuration

To configure the *BroadSoft VoIP Collector* modify two configuration files.

1 *Event Processing Manager* configuration

The *BroadSoft VoIP Collector* can be used with an *APG Datastore* or a *MySQL Database*.

In both cases, the **processing.xml** file is modified accordingly as illustrated in the examples in this section.

The path to the file is:

<APG>/Event-Processing/Event-Processing-Manager/Default/conf/processing.xml

APG Datastore example:

```
...
<processing-element enabled="true" name="DATASTORE-DB" config="APG-Datastore-Writer/Default/
conf/datastore-broadsoft-cdr.xml" />

<processing-element enabled="true" name="BROADSOFT-COLLECTOR" config="Broadsoft-VoIP-Collector/
Default/conf/broadsoft-voip-collector.xml" data="DATASTORE-DB" />
...
```

MySQL Database example:

```
...
<processing-element enabled="true" name="MYSQL-DB" config="Generic-Live-Writer/Default/conf/
mysql-broadsoft-cdr.xml" />

<processing-element enabled="true" name="BROADSOFT-COLLECTOR" config="Broadsoft-VoIP-Collector/
Default/conf/broadsoft-voip-collector.xml" data="MYSQL-DB" />
...
```

2 BroadSoft VoIP Collector configuration

The second configuration file to edit is **broadsoft-voip-collector.xml**.

The path to the file is:

<APG>/Event-Processing/BroadSoft-VoIP-Collector/Default/conf/broadsoft-voip-collector.xml

The code below is a complete example of a configuration file; an explanation of each parameter follows.

```
<?xml version="1.0" encoding="utf-8"?>
<configuration xmlns="http://www.watch4net.com/Events/BroadsoftVoIPCollector"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.watch4net.com/Events/BroadsoftVoIPCollector_broadsoft-voip-
collector.xsd">

  <cdr type="csv" pattern="^(.....)-(\d\d\d\d[01]\d[0123]\d\d\d\d\d\d)-([1234])-(.*)-(\d*)
\.(csv)\$"/>

  <fieldlist-file version="14.3" normal="conf/normal_fields.txt" failover="conf/
failover_fields.txt"/>
</configuration>
```

```
<cache-directory>cache</cache-directory>

<ftp-server>

  <host>hostname:port</host>
  <protocol>ftp</protocol>

  <mode>active</mode>
  <connection-timeout>60000</connection-timeout>
  <read-timeout>60000</read-timeout>
  <mtime-threshold>5000</mtime-threshold>

  <max-file>10000</max-file>

  <username>username</username>
  <password>password</password>

  <remote-directory>cdr</remote-directory>

</ftp-server>

<collecting-period>300</collecting-period>
<source>Broadsoft-CDR-Collector</source>
<retry-time-failure>7200</retry-time-failure>

<cdr-type>
  <type>Long Duration</type>
  <type>Interim</type>
  <type>Normal</type>
  <type>Failover</type>
</cdr-type>

</configuration>
```

Explanation of Parameters:

- **cdr**: This parameter determines the type of CDR files that will be downloaded and parsed. The attribute **type** can be either **csv** or **xml** and determines the type of CDR that is sent by the CM. The attribute **pattern** is any Java regex which is matched against filenames to determine which files to download.

- **fieldlist-file**: This parameter determines the fields list order of the *CSV CDRs*, the XML tag of the *XML CDRs* and the valid version of each *CDR*.

- **version** attribute: The version determines which version of *CDR* is parsed. If the version is left *EMPTY*, then no version check will be done and all *CDR* files are parsed. If it is set to 14.3 for example, then only the *CDR* with version 14.3 is parsed.

The version of a *CDR* is written in the header of each XML file and in the first line of each CSV file.

- **normal** attribute: References another configuration file which determines the order of the fields in a CSV *CDR* as well as each field's corresponding XML *CDR* field.
 - **failover** attribute: This attribute performs the same function as the normal attribute except it is only used for failover *CDRs*. It references a different configuration file specifically for failover *CDRs* since failover *CDRs* contain less fields and they are in a different order than normal *CDRs*.
- **cache-directory**: This is the cache directory where *CDR(s)* files are copied when using the FTP or SFTP protocol.
 - **ftp-server**: This is the *Billing server* where the *CDR* files will be downloaded. The Billing server is an FTP server where the CM is sending *CDRs* as they are generated.
 - **host**: This is the host address of the FTP server. It can be the name of the machine or the IP address. The *port* may also be specified. If the *port* is not specified, then the default port number is used. For the *FTP* protocol the default port number is **21** and for the *SFTP* protocol the default port number is **22**.
 - **protocol**: This is the protocol used to get the *CDR* files from the FTP server. Supported protocols are **ftp**, **sftp** and **local**. If the local protocol is used, the *host*, *username* and *password* parameter can be left blank. The local protocol means that the *BroadSoft VoIP Collector* will try to retrieve the *CDR* files on the same machine as the collector. It is useful if the *Billing server* is on the same machine as the *BroadSoft VoIP Collector*.
 - **mode**: This is the connection mode used by the FTP protocol. Supported modes are **active** and **passive**. This parameter has no effect if the *protocol* is not equal to *ftp*.
 - **connection-timeout**: This is the connection timeout (*in milliseconds*) used by the FTP and SFTP protocol.
 - **read-timeout**: This is the read timeout (*in milliseconds*) used by the FTP and SFTP protocol.

- **mtime-threshold**: This is the time (*in milliseconds*) that the collector waits between polling intervals to determine whether the file is still being written. If the file size is stable between intervals then it is no longer growing and will be downloaded. Files that are still growing are only retrieved at the next collect which occurs according to the collecting-period parameter.
- **max-file**: This is the maximum number of file that the collector will download on each collects.
- **username/password**: This is the credential to use for connecting to the FTP server.
- **remote-directory**: This is the remote directory (on the FTP server or local machine) where the *CDRs* are located.
- **collecting-period**: This is the collecting delay (*in seconds*). It is the time that the collector will wait between each collects.
- **source**: The value of the source property, as it will appear in APG. The source property is used to identify the collector that pushed the given metric in APG.
- **retry-time-failure**: Determines how much time (*in seconds*) the *BroadSoft VoIP Collector* tries sending an *Event* to the next *Processing Element*, in case of a failure. The most plausible cause of a failure is that the next *Processing Element* is the Database, which is down. After the retry period has expired, the *Event* will be dropped.

A value of 0 mean that the collector will drop the events in case of a failure.

- **cdr-type**: This is the *CDR* type that is parsed and sent to the next processing element. Supported types : **Start, End, Long Duration, Interim, Normal, Failover**.