Grandstream Networks, Inc.

UCM6xxx Series

CDR and REC API Guide

Table of Contents

Table of Figures
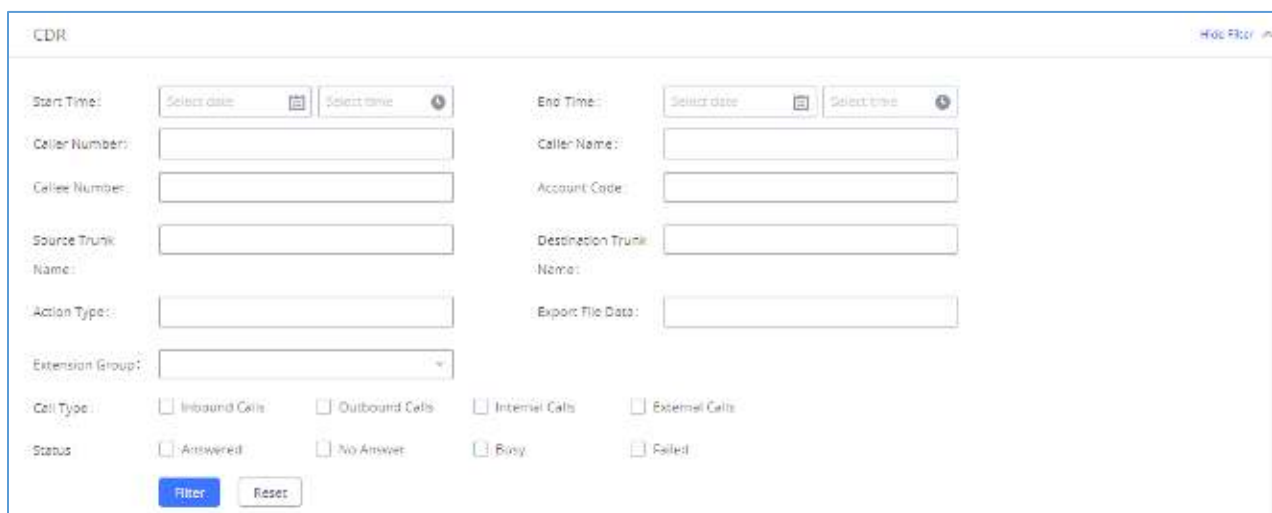
Table of Tables

UCM6XXX CDR and REC API Guide

CDR REPORT

CDR (Call Detail Record) is a data record generated by the PBX that contains attributes specific to a single instance of phone call handled by the PBX. It has several data fields to provide detailed description for the call, such as phone number of the calling party, phone number of the receiving party, start time, call duration…

CDR Filter

On the UCM6XXX, the CDR can be accessed under web GUI→**CDR**→**CDR**. Users could filter the call report by specifying the date range and criteria, depending on how the users would like to include the logs to the report. Click on **"Display Filter"** button to display the filter criteria, a second click will start the search.

Figure 1: CDR Filter

Table 1: CDR Filter Criteria

Call Type	Groups the following:
	• **Inbound calls**: Inbound calls are calls originated from a non-internal source (like a VoIP trunk) and sent to an internal extension.
	• **Outbound calls**: Outbound calls are calls sent to a non-internal source (like a VoIP trunk) from an internal extension.
	• **Internal calls**: Internal calls are calls from one internal extension to another extension, which are not sent over a trunk.
	• **External calls**: External calls are calls sent from one trunk to another trunk, which are not sent to any internal extension.
Status	Filter with the call status, the available statuses are the following: • Answered • No Answer • Busy • Failed

Source Trunk Name	Select source trunk(s) and the CDR of calls going through inbound the trunk(s) will be filtered out.
Destination Trunk Name	Select destination trunk(s) and the CDR of calls going outbound through the trunk(s) will be filtered out.
Action Type	Filter calls using the Action Type, the following actions are available: • Announce • Announcement page • Dial • Announcements • Callback • Call Forward • Conference • Disa • Fax • Follow Me • IVR • Page • Parked Call • Queue • Ring Group • Transfer • VFax • VM • VMG • Video Conference • VQ_Callback • Wakeup • Emergency Call • Emergency Notify • SCA
Extension Group	Specify the Extension Group name to filter with.
Export File Data	Select the fields that will be exported, the following fields are available: • Account Code • Session • Action owner • Action type • Source trunk name • Dest trunk name • Caller name • Caller number • Caller ID • Callee number • Answer by • Context • Start time

UCM6XXX CDR and REC API Guide

	• Answer time • End time • Call time • Talk time • Source channel • Dest channel • Last app • Dest channel extension • Disposition • Last data • AMAFLAGS • UIQUEID • USERFIELD
Account Code	Select the account Code to filter with. If pin group CDR is enabled, the call with pin group information will be displayed as part of the CDR under Account Code Field.
Start Time	Specify the start time to filter the CDR report. Click on the calendar icon on the right and the calendar will show for users to select the exact date and time.
End Time	Specify the end time to filter the CDR report. Click on the calendar icon on the right and the calendar will show for users to select the exact date and time.
Caller Number	Enter the caller number to filter the CDR report. CDR with the matching caller number will be filtered out. User could specify a particular caller number or enter a pattern. '.' matches zero or more characters, only appears in the end. 'X' matches any digit from 0 to 9, case-insensitive, repeatable, only appears in the end. For example: **3XXX**: It will filter out CDR that having caller number with leading digit 3 and of 4 digits' length. **3.**: It will filter out CDR that having caller number with leading digit 3 and of any length.
Caller Name	Enter the caller name to filter the CDR report. CDR with the matching caller name will be filtered out.
Callee Number	Enter the callee number to filter the CDR report. CDR with the matching callee number will be filtered out.

The call report will display as the following figure shows and since firmware version 1.0.15.16, the report will display only CDR records for the current month since it's split month by month.

In order to get records from previous months, users would need to use the filter search and specify start date and end date to get records from the required period of time.

Figure 2: Call Report

CDR Report Data Fields

The CDR report has the following data fields:

Field	Type	Description	Access
Account Code	String	An account code associated with the Party A channel.	r/w
Caller Number	String	The Caller ID Number.	r
Callee Number	String	The destination extension.	r
Context	String	The context of the call.	r
CallerID	String	The caller ID.	r
Source Channel	String	The name of the source channel.	r
Dest Channel	String	The name of the destination channel.	r
Lastapp	String	The last application the Party A channel executed.	r
Lastdata	String	The application data for the last application the Party A channel executed.	r
Start time	Date/time	The time the CDR was created.	r
Answer Time	Date/time	The time when Party A was answered, or when the bridge between a Party A and a Party B was created.	r
End time	Date/time	The time when the CDR was finished. This occurs when either party	r

		hangs up, or when the bridge between the parties is broken.	
Call time	Integer	The time in seconds from start time until end time.	r
Talk time	Integer	The time in seconds from answer time until end time.	r
Disposition	Enum	The final known disposition of the CDR record. The possible values are: "ANSWERED","NO ANSWER, CONGESTION, FAILED and BUSY.	r
Amaflags	Enum	A flag specified on the Party A channel. The possible values are: "OMIT, BILLING and DOCMENTATION.	r/w
Uniqueid	String	A unique identifier for the Party A channel	r
Userfield	String	A user defined field set on the channels. If set on both the Party A and Party B channel, the userfields of both are concatenated and separated by a comma.	r/w
Dest Channel Extension	String	The destination extension of the call	r
Caller Name	String	The name of the caller	r
Answer by	String	The extension to be called	r
Session	String	A numeric value that, combined with uniqueid and linkedid, can be used to uniquely identify a single CDR record	r
Action Owner	String	The extension that made the call	r
Action Type	String	The action type of the call. The supported Action Type values are: DIAL, ANNOUNCEMENTS, CALLBACK, CALLFORWARD, CONFERENCE, DISA, FAX, FALLOWME, IVR, PAGE, PARKEDCALL, QUEUE, RINGGROUP, TRANSFER, VFAX, VM, VMG, WAKEUP, EMERGENCYCALL, EMERGENCYNOIFY, SCA, VIDEOCONFERENCE, PRESENCE_STATUS, ANNOUNCEMENT PAGE.	r
Source Trunk Name	Sting	The inbound route trunk name	r
Dest Trunk Name	String	The outbound route trunk name	r

Example of a CDR report Data fields:

- **Account code: --**
- **Caller Number:** 1000
- **Callee number:** 7000
- **Context:** ext-did-1
- **Caller ID:** "1000" <1000>

- **Source Channel**: PJSIP/trunk_1-00000000

- **Dest Channel: --**

- **Lastapp:** ForkCDR

- **Lastdata:** ae

- **Start time**: 7/13/2018 12:00:12 PM

- **Answer time**: 7/13/2018 12:00:12 PM

- **End time**: 7/13/2018 12:00:32 PM

- **Call time**: 19 (in seconds)

- **Talk Time:** 19

- **Disposition:** ANSWERED

- **Amaflags**: DOCUMENTATION

- **UniqueID:** 1531497605

- **Userfield**: Inbound

- **Dest channel extension:** 7000

- **Caller name:** 1000

- **Answer by:** 7000

- **Session:** 1531497605793100-1000

- **Action owner:** 1000

- **Action type**: DIAL.

- **Source Trunk name:** test

- **Dest Trunk name: --**

CDR Report Operations

Users could perform the following operations on the CDR report.

- **Sort by "Start Time"**
  Click on the header of the column to sort the report by "Start Time". Clicking on "Start Time" again will reverse the order.

- **Download Searched Results**
  Click on "Download Search Result(s)" to export the records filtered out to a .csv file.

- **Download All Records**
  Click on "Download All Records" to export all the records to a .csv file.

- **Delete All**
  Click on "Delete All" button to remove all the call report information.

- **Recording File Options**

  Recording file is available to play or download if the PBX records the call, click on 📼 and the following options will be displayed.
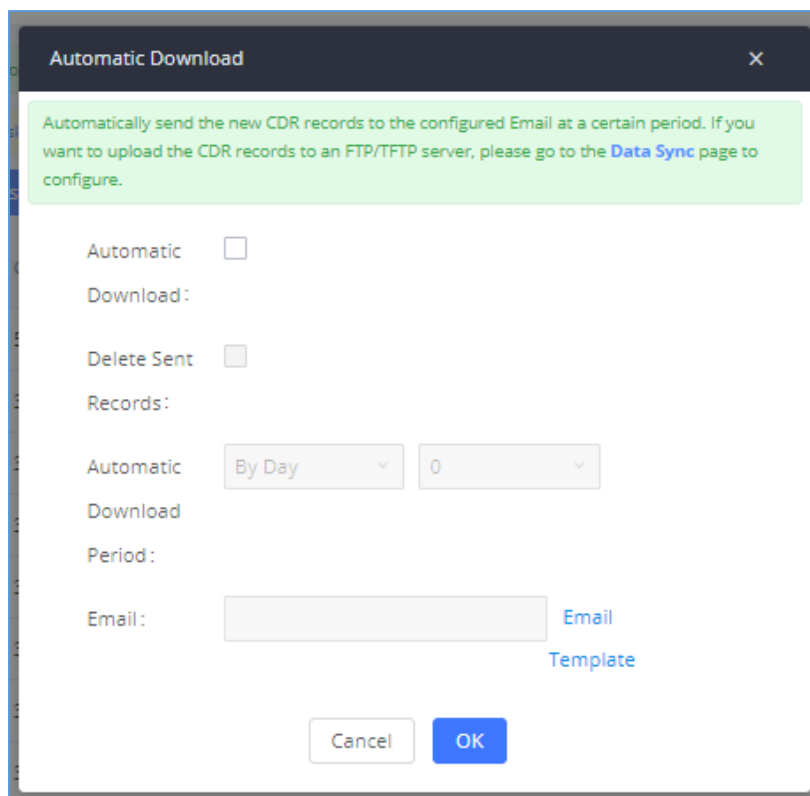
  ⬇️ : Download the voice recording for the call

  ▶️ : Play the voice recording for the call

  🗑️ : Delete the voice recording for the call

- **Automatic Download**

  User could configure the UCM6XXX to automatically download the CDR records and send the records to multiple email recipients in a specific hour. Click on "Automatic Download", and configure the parameters in the dialog below.

Figure 3: Automatic Download Settings

To receive CDR record automatically from Email, check "Enable" and select a time period "By Day" "By Week" or "By Month", select Hour of the day as well for the automatic download period. Make sure you have entered an Email or multiple email addresses where to receive the CDR records.

- **Options**

  Click on the icon ▸ to view more details about the call.

UCM6XXX CDR and REC API Guide

CDR CSV FILE

The downloaded CDR .csv file has different format from the web UI CDR. Here are some descriptions.

- **Caller number, Callee number**

"Caller number": the caller ID.
"Callee number": the callee ID.

If "Caller number" shows empty, "Callee number" shows "s" (see the below figure) and the "Source Channel" contains "DAHDI", this means the call is from FXO/PSTN line. For FXO/PSTN line, we only know there is an incoming request when there is incoming call but we don't know the number being called. So we are using "s" to match it where "s" means "start".

account code	caller number	callee number	context	callerid	source channel	dest channel	lastapp	lastdata
	1000	2000	from-internal	"1000" <1000>	PJSIP/1000-00000000	PJSIP/trunk_1-00000001	Dial	PJSIP/2000@trunk_1,,b(callee-handler^s^
	2001	1000	did-local	"Mario" <2001>	PJSIP/trunk_1-00000002	PJSIP/1000-00000003	Dial	PJSIP/1000/sip:1000@192.108.6.95:5066,6(
	1000	2001	from-internal	"1000" <1000>	PJSIP/1000-00000004	PJSIP/trunk_1-00000005	Dial	PJSIP/2001@trunk_1,,b(callee-handler^s^
	2000	1000	did-local	"Mark" <2000>	PJSIP/trunk_1-00000006	PJSIP/1000-00000007	Dial	PJSIP/1000/sip:1000@192.108.6.95:5066,6(
	1002	1000	from-internal	"1002" <1002>	PJSIP/1002-00000000	PJSIP/1000-00000001	Dial	PJSIP/1000/sip:1000@192.168.6.96:5066,6(
	2000	2001	from-internal	"aeae5d14" <2000>	PJSIP/2000-00000000	PJSIP/2001-00000001	Dial	PJSIP/2001/sip:2001@192.168.6.186:5060,(
	2000	2001		"aeae5d14" <2000>	PJSIP/2000-00000000		VoiceMail	2001@default,u
	2001	2000	from-internal	"2001" <2001>	PJSIP/2001-00000002		VoiceMail	2000@default,b
	2000	2001	from-internal	"aeae5d14" <2000>	PJSIP/2000-00000003	PJSIP/2001-00000004	Dial	PJSIP/2001/sip:2001@192.168.6.186:5060,(

Figure 4: Downloaded CDR File Sample

- **Context**

There are different context values that might show up in the downloaded CDR file. The actual value can vary case by case. Here are some sample values and their descriptions.

  - **from-internal**: internal extension makes outbound calls.
  - **ext-did-XXXXX:** inbound calls. It starts with "ext-did", and "XXXXX" content varies case by case, which also relate to the order when the trunk is created.
  - **ext-local**: internal calls between local extensions.

- **Source Channel, Dest Channel**

Sample 1:

callee number	context	callerid	source channel	dest channel	lastapp	lastdata
2000	from-internal	"1000" <1000>	PJSIP/1000-00000000	PJSIP/trunk_1-00000001	Dial	PJSIP/2000@trunk_1,,b(callee-handler^s^1)
1000	did-local	"Mario" <2001>	PJSIP/trunk_1-00000002	PJSIP/1000-00000003	Dial	PJSIP/1000/sip:1000@192.168.6.95:5066,60,zb(callee-handler^s^1)
2001	from-internal	"1000" <1000>	PJSIP/1000-00000004	PJSIP/trunk_1-00000005	Dial	PJSIP/2001@trunk_1,,b(callee-handler^s^1)
1000	did-local	"Mark" <2000>	PJSIP/trunk_1-00000006	PJSIP/1000-00000007	Dial	PJSIP/1000/sip:1000@192.168.6.95:5066,60,zb(callee-handler^s^1)
1000	from-internal	"1002" <1002>	PJSIP/1002-00000000	PJSIP/1000-00000001	Dial	PJSIP/1000/sip:1000@192.168.6.96:5066,60,zb(callee-handler^s^1)
2001	from-internal	"aeae5d14" <2000>	PJSIP/2000-00000000	PJSIP/2001-00000001	Dial	PJSIP/2001/sip:2001@192.168.6.186:5060,60,zM(record-check,StartMoni
2001		"aeae5d14" <2000>	PJSIP/2000-00000000		VoiceMail	2001@default,u
2000	from-internal	"2001" <2001>	PJSIP/2001-00000002		VoiceMail	2000@default,b
2001	from-internal	"aeae5d14" <2000>	PJSIP/2000-00000003	PJSIP/2001-00000004	Dial	PJSIP/2001/sip:2001@192.168.6.186:5060,60,zM(record-check,StartMoni

Figure 5: Downloaded CDR File Sample - Source Channel and Dest Channel 1

DAHDI means it is an analog call, FXO or FXS.

  - For UCM6102/6202, DAHDI/(1-2) are FXO ports, and DAHDI(3-4) are FXS ports.
  - For UCM6104/6204, DAHDI/(1-4) are FXO ports, and DAHDI(5-6) are FXS ports.

- For UCM6108/6208, DAHDI/(1-8) are FXO ports, and DAHDI(9-10) are FXS ports.
- For UCM6116, DAHDI/(1-16) are FXO ports, and DAHDI/(17-18) are FXS ports.
- For UCM6510, DAHDI/(1-2) are FXO ports, and DAHDI(3-4) are FXS ports.

Sample 2:

caller number	caller name	callee number	answer by	context	start time	answer time	end time	call time	talk time	source channel
1001		1002	1002	from-internal	11/23/2016 10:47	11/23/2016 10:47	11/23/2016 10:47	16	16	PJSIP/1001-00000000
1001		6400	6400	from-internal	11/23/2016 10:49		11/23/2016 10:49	0	0	PJSIP/1001-00000001

Figure 6: Downloaded CDR File Sample - Source Channel and Dest Channel 2

"PJSIP" means it's a SIP call. There are three possible formats:

1. **PJSIP/NUM-XXXXXX,** where NUM is the local SIP extension number. The last XXXXX is a random string and can be ignored.

2. **PJSIP/trunk_X/NUM,** where trunk_X is the internal trunk name, and NUM is the number to dial out through the trunk.

3. **PJSIP/trunk_X-XXXXXX,** where trunk_X is the internal trunk name and it is an inbound call from this trunk. The last XXXXX is a random string and can be ignored.

There are some other possible values, but these values are almost the application name which are used by the dialplan.

- **IAX2/NUM-XXXXXXX**: it means this is an IAX call.

- **Local/@from-internal-XXXXX**: it is used internally to do some special feature procedure. We can simply ignore it.

- **Hangup**: the call is hung up from the dialplan. This indicates there are some errors or it has run into abnormal cases.

- **Playback**: play some prompts to you, such as 183 response or run into an IVR.

- **ReadExten**: collect numbers from user. It may occur when you input PIN codes or run into DISA.

API CONFIGURATION

The UCM6XXX supports CDR API interface to interact with third party external billing software and collect recording details files from the PBX. The API uses HTTPS to request the CDR data and call recording data matching given parameters as configured on the third-party application, as well as a new API interface to query, edit PBX settings and implement multiple call functions on another server connected to it via API.

This guide will focus on the CDR API configuration and for more details about the new supported IP, please refer to the below link :
http://www.grandstream.com/sites/default/files/Resources/UCM_API_Guide.pdf

API Configuration

Before accessing the CDR API (to access call detail records) and REC API (to access call recording files), the administrators need to enable API and configure the access/authentication information on the UCM6XXX first under **Value-added Features→API Configuration**.

Table 2: API Configuration Parameters

CDR HTTP API Settings	
Basic Settings	
Enable	Enable/Disable API. The default setting is disabled.
TLS Bind Address	Configure the IP address for TLS server to bind to. "0.0.0.0" means binding to all interfaces. The port number is optional, and the default port number is 8443. The IP address must match the common name (host name) in the certificate so that the TLS socket won't bind to multiple IP addresses. The default setting is 0.0.0.0:8443.
User Name	Configure the user name for TLS authentication. If not configured, authentication will be skipped.
Password	Configure the password for TLS authentication. This is optional.
Permitted IP(s)	Specify a list of IP addresses permitted by CDR and REC API. This creates an API-specific access control list. Multiple entries are allowed. For example, "192.168.40.3/255.255.255.255" denies access from all IP addresses except 192.168.40.3. The default setting is blank, meaning all IP addresses will be denied.
Other Settings	
TLS Private Key	Upload TLS private key. The size of the key file must be under 2MB. This file will be renamed as 'private.pem' automatically.
TLS Cert	Upload TLS cert. The size of the certificate must be under 2MB. This is the certificate file (*.pem format only) for TLS connection. This file will be renamed as "certificate.pem" automatically. It contains private key for the client and signed certificate for the server.

API Module	
CDR API	Allows access to the CDR via HTTPS.
REC API	Allows access to call recordings via HTTPS.
PMS API	Allows access to PMS API via HTTPS.

Important note:

Starting from firmware 1.0.16.20, UCM6xxx supports only digest authentication.

CDR API – ACCESS CALL DETAIL RECORDS

CDR API URL Format

The format of the HTTPS request for the CDR API is shown as below. This is used to request the CDR data matching given parameters as set by the third-party application.

https://[UCM IP]:[Port]/cdrapi?[option1]=[value]&[option2]=[value]&...

By default, the port number for the API is **8443**.

CDR API URL Parameters

The options included in the above request URI control the record matching and output format. For CDR matching parameters, all non-empty parameters must have a match to return a record. Parameters can appear in the URI in any order. Multiple values given for caller or callee will be concatenated.

The following table shows the parameter list used in the CDR API.

Table 3: CDR API URL Parameters

Field	Value	Details
format	csv, xml, json	Determines the format for output of matching CDR rows. Default is csv (comma separated values).
numRecords	Number: 0-1000	Number of records to return. Default is 1000, which is also the maximum allowed value.
offset	Number	Number of matching records to skip. This will be combined with numRecords to receive all matches over multiple responses. Default is 0.
caller **Callee** **Answeredby**	Comma separated extensions, ranges of extensions, or regular expressions. Example: **caller=5300,5302-5304,_4@** -OR- **caller=5300&caller=5302-5304&caller=_4@** (Matches extensions 5300, 5302, 5303, 5304, and any extension containing 4 as the	Filters based on src (caller) or dst (callee) value, matching any extension contained in the parameter input string. Patterns containing one or more wildcards ('@' or '_') will match as a regular expression, and treat '-' as a literal hyphen rather than a range signifier. The '@' wildcard matches any number of characters (including zero), while '_' matches any single character. Otherwise, patterns containing a single hyphen will be matching a range of numerical extensions, with non-numerical characters ignored, while patterns containing multiple

	second digit/character).	hyphens will be ignored. (The pattern "0-0" will match all non-numerical and empty strings).
startTime	Date and/or time of day in any of the following formats: YYYY-MM-DDTHH:MM YYYY-MM-DDTHH:MM:SS YYYY-MM-DDTHH:MM:SS.SSS (literal 'T' character separator in above three formats) HH:MM HH:MM:SS HH:MM:SS.SSS now DDDDDDDDDD	Filters based on the start (call start time) value. Calls which start within this period (inclusive of boundaries) will match, regardless of the call answer or end time. An empty value for either field will be interpreted as range with no minimum or maximum respectively. Strings without a date have a default value of 2000-01-01. Strings without a time of day have a default value of 00:00 UTC, while strings with a time of day specified may also optionally specify a time zone offset - replace '+' in time zone offset with '%2B' (see http://www.w3.org/TR/NOTE-datetime).
endTime		
minDur	Number (duration in seconds)	Filters based on the billsec value, the duration between call answer and call end.
maxDur		

Example Queries

The following illustrates the format of queries to accomplish certain requests. In most cases, multiple different queries will accomplish the same goal, and these examples are not intended to be exhaustive, but rather to bring attention to particular features of the CDR API connector.

- **Query 1**: Request all records of calls placed on extension 5300 which last between 8 and 60 seconds (inclusive), with results in CSV format.

    https://192.168.254.200:8443/cdrapi?format=CSV&caller=5300&minDur=8&maxDur=60

    -OR-

    https://192.168.254.200:8443/cdrapi?caller=5300&minDur=8&maxDur=60

- **Query 2**: Request all records of calls placed on extension 5300 or in the range 6300-6399 to extensions starting with 5, with results in XML format.

UCM6XXX CDR and REC API Guide

https://192.168.254.200:8443/cdrapi?format=XML&caller=5300,6300-6399&callee=5@

-OR-

https://192.168.254.200:8443/cdrapi?cdrapi?format=XML&caller=5300&caller=6300-6399&callee=5@

- **Query 3:** Request 10 records of calls placed on extension 5300 with results in JSON format.

  https://192.168.254.200:8443/cdrapi?format=JSON&caller=5300&numRecords=10

- **Query 4**: Request all records of calls placed on extensions containing substring "53" prior to January 23, 2013 00:00:00 UTC to extensions 5300-5309, with results in CSV format.

  https://192.168.254.200:8443/cdrapi?caller=@53@&callee=5300-5309&endTime=2013-01-23

  -OR-

  https://192.168.254.200:8443/cdrapi?caller=@53@&callee=530_&endTime=2013-01-23T00:00:00

- **Query 5**: Request all records of calls placed by an Anonymous caller during July 2013 Central Standard Time to extensions starting with 2 or 34 or ending with 5, with results in CSV format.

  https://192.168.254.200:8443/cdrapi?caller=Anonymous&callee=2@,34@,@5&startTime=2013-07-01T00:00:00-06:00&endTime=2013-07-31T23:59:59-06:00

- **Query 6**: Request all records during July 2013 Central Standard Time, 200 at a time, with results in CSV format.

  https://192.168.254.200:8443/cdrapi?startTime=2013-07-01T00:00:00-06:00&endTime=2013-07-31T23:59:59-06:00&numRecords=200&offset=0

  -THEN-

  https://192.168.254.200:8443/cdrapi?sstartTime=2013-07-01T00:00:00-06:00&endTime=2013-07-31T23:59:59-06:00&numRecords=200&offset=200

  -THEN-

  https://192.168.254.200:8443/cdrapi?startTime=2013-07-01T00:00:00-06:00&endTime=2013-07-31T23:59:59-06:00&numRecords=200&offset=400

⚠ **Notes:**

- Disallowed characters in the caller, callee, startTime, or endTime strings, and non-digit characters in the values of numRecords, offset, minDur, or maxDur, will result in no records returned - the appropriate container/header for the output format will be the only output. If the format parameter is in error, the CSV header will be used. Error messages will appear in the Asterisk log (along with errors stemming from failed database connections, etc.).

- Other errors which return no records include:

  o Multiple hyphens in an extension range (e.g. caller=5300-5301-,6300)

  o Empty parameter value (e.g. caller=)

- o Extension values starting with comma, or with consecutive commas (e.g. caller=5300,,5303)

- o Unknown parameters (e.g. caller=5300) or URI ending with '&'

- o Except for caller and callee, multiple instances of the same parameter within the URI (e.g. minDur=5&minDur=10)

Output

Fields

From UCM6XXX firmware 1.0.11.x and higher, the CDR output has the following changes:

- New output fields are added for all format (CSV, XML and JSON):
    - o *session*
    - o *action_owner*
    - o *action_type*
    - o *src_trunk_name*
    - o *dst_trunk_name*

- For XML and JSON format output, sub record is added for certain call scenarios, e.g., call transfer. (CSV format doesn't have sub record added).

"session"

The new field "**session**" is the unique identifier for the call record and it exists in all CDR for this call. Same "**session**" value indicates this is the same call. The value of the "**session**" consists of the time when the channel is generated and the caller extension number, for example, 1461920017104140-1000.

"action_owner"

The new field "**action_owner**" indicates the owner of the CDR record, which is the initiator of this call. In a call transfer scenario, the sub record 1 has A as the **action_owner** and the sub record 2 has B as the **action_owner** who is performing the call transfer.

"action_type"

The new field "**action_type**" indicates the type for the call record. If this call type has its own extension number or name, it will be listed in *[ ]*. For example, if an extension calls conference number 6300, the **action_type** will be *CONFERENCE [6300]*. Here are the possible values of **action_type**:

- **DIAL**: basic call
- **TRANSFER**: call transfer
- **PAGE**: paging/intercom
- **CALLFORWARD**: call forward always, call forward no answer, call forward busy

- **IVR**: enter IVR or call from IVR
- **RINGGROUP**: ring group
- **VM**: voice mail
- **VMG**: voice mail group
- **CONFERENCE**: conference call by calling into UCM's conference extension
- **DISA**: enter DISA or call from DISA
- **FAX**: FAX
- **VFAX**: VFAX
- **Announcements**: Announcement center
- **FOLLOWME**: Follow Me
- **QUEUE**: call queue
- **CALLBACK**: call back

"src_trunk_name"

Source trunk name indicates the name of the trunk for the inbound call. For example, if the FOLLOWME call is from trunk, this call record will have source trunk name field.

"dst_trunk_name"

Destination trunk name indicates the name of the trunk for the outbound call. Only outbound call has dst_trunk_name field.

XML Output

On UCM6XXX, the XML output looks like below:

```
<cdr session=xxxxx-xxxx>
<main_cdr>
...
</main_cdr>
<sub_cdr>
<AcctID>1</AcctID>
...
</sub_cdr>
<sub_cdr>
<AcctID>2</AcctID>
...
</sub_cdr>
</cdr>
```

- One **<cdr>** element represents one call record. "**session**" is the unique identifier for this call.
- **<main_cdr>** is the primary record displayed in UCM web UI status->CDR page.
- **<sub_cdr>** is the sub record for this call. **<AcctID>** is the index for sub record.

UCM6XXX CDR and REC API Guide

Example:

Assuming the following scenario:

Extension A: 5000
Extension B: 5001
Extension C: 5002
A calls B->B answers->B Transfers the call to C.

- **The XML output for the scenario above looks like below:**

```
<root>
<cdr session="1461920017104140-1000">
<main_cdr>
<AcctId/>
<accountcode/>
<src>1000</src>
<dst>1001</dst>
<dcontext/>
<clid/>
<channel/>
<dstchannel/>
<lastapp/>
<lastdata/>
<start>2016-04-29 02:53:37</start>
<answer/>
<end/>
<duration>16</duration>
<billsec>9</billsec>
<disposition/>
<amaflags/>
<uniqueid/>
<userfield/>
<channel_ext/>
<dstchannel_ext/>
<service/>
<caller_name/>
<recordfiles/>
<dstanswer/>
<chanext/>
<dstchanext/>
<session>1461920017104140-1000</session>
<action_owner/>
<action_type>DIAL</action_type>
<src_trunk_name/>
<dst_trunk_name/>
</main_cdr>
<sub_cdr>
<AcctId>1</AcctId>
<accountcode/>
<src>1000</src>
```

```
<dst>1001</dst>
<dcontext>from-internal</dcontext>
<clid>"John Doe" <1000></clid>
<channel>PJSIP/1000-00000003</channel>
<dstchannel>PJSIP/1001-00000004</dstchannel>
<lastapp>Dial</lastapp>
<lastdata>PJSIP/1001/sip:1001@192.168.124.179:5066,60,ztTkKA(dialog-being-
recorded)M(reco</lastdata>
<start>2016-04-29 02:53:37</start>
<answer>2016-04-29 02:53:44</answer>
<end>2016-04-29 02:53:49</end>
<duration>12</duration>
<billsec>5</billsec>
<disposition>ANSWERED</disposition>
<amaflags>DOCUMENTATION</amaflags>
<uniqueid>1461920017.7</uniqueid>
<userfield>Internal</userfield>
<channel_ext>1000</channel_ext>
<dstchannel_ext>1001</dstchannel_ext>
<service>s</service>
<caller_name>John Doe</caller_name>
<recordfiles>auto-1461920017-1000-1001.wav@</recordfiles>
<dstanswer>1001</dstanswer>
<chanext/>
<dstchanext/>
<session>1461920017104140-1000</session>
<action_owner>1000</action_owner>
<action_type>DIAL</action_type>
<src_trunk_name/>
<dst_trunk_name/>
</sub_cdr>
<sub_cdr>
<AcctId>2</AcctId>
<accountcode/>
<src>1000</src>
<dst>1002</dst>
<dcontext>from-internal-xfer</dcontext>
<clid>"John Doe" <1000></clid>
<channel>PJSIP/1000-00000003</channel>
<dstchannel>PJSIP/1002-00000005</dstchannel>
<lastapp>Dial</lastapp>
<lastdata>PJSIP/1002/sip:1002@192.168.124.136:5066,60,ztTkKb(callee-
handler^s^1)</lastdata>
<start>2016-04-29 02:53:49</start>
<answer>2016-04-29 02:53:49</answer>
<end>2016-04-29 02:53:54</end>
<duration>4</duration>
<billsec>4</billsec>
<disposition>ANSWERED</disposition>
<amaflags>DOCUMENTATION</amaflags>
```

```
<uniqueid>1461920017.7</uniqueid>
<userfield>Internal</userfield>
<channel_ext>1000</channel_ext>
<dstchannel_ext>1002</dstchannel_ext>
<service>s</service>
<caller_name>John Doe</caller_name>
<recordfiles/>
<dstanswer>1002</dstanswer>
<chanext/>
<dstchanext/>
<session>1461920017104140-1000</session>
<action_owner>1001</action_owner>
<action_type>TRANSFER</action_type>
<src_trunk_name/>
<dst_trunk_name/>
</sub_cdr>
</cdr>
</root>
```

JSON Output

In general, the JSON output format looks like below:

```
{cdr_root:[
{
cdr: xxxx-xxxx,
main_cdr:{...},
sub_cdr_1:{...},
...
sub_cdr_n:{...}
},
...
{
cdr: xxxx-xxxx,
src: 2003,
...,
action_type:DIAL
}]
}
```

- **cdr_root** represents the JSON array that has all the call records in this query.
- The above result highlighted in red indicates the current call. The "**cdr**" value is the unique identifier for this call, similar to the <session> element in XML format output.
- **main_cdr** is the primary record displayed.
- **sub_cdr_x** is the sub record for this call.

Example:
Assuming the following scenario:
Extension A: 5000
Extension B: 5001
Extension C: 5002
A calls B→B answers→B Transfers the call to C.

- **The JSON output format for the above scenario looks like this:**

```
{
"cdr_root": [
{
"cdr": "1461920017104140-1000",
"main_cdr": {
"AcctId": "",
"accountcode": "",
"src": "1000",
"dst": "1001",
"dcontext": "",
"clid": "",
"channel": "",
"dstchannel": "",
"lastapp": "",
"lastdata": "",
"start": "2016-04-29 02:53:37",
"answer": "",
"end": "",
"duration": "16",
"billsec": "9",
"disposition": "",
"amaflags": "",
"uniqueid": "",
"userfield": "",
"channel_ext": "",
"dstchannel_ext": "",
"service": "",
"caller_name": "",
"recordfiles": "",
"dstanswer": "",
"chanext": "",
"dstchanext": "",
"session": "1461920017104140-1000",
"action_owner": "",
"action_type": "DIAL",
"src_trunk_name": "",
"dst_trunk_name": ""
},
"sub_cdr_1": {
"AcctId": "1",
"accountcode": "",
"src": "1000",
"dst": "1001",
"dcontext": "from-internal",
"clid": "\"John Doe\" <1000>",
"channel": "PJSIP/1000-00000003",
"dstchannel": "PJSIP/1001-00000004",
"lastapp": "Dial",
"lastdata": "PJSIP/1001/sip:1001@192.168.124.179:5066,60,ztTkKA(dialog-being-
recorded)M(reco",
"start": "2016-04-29 02:53:37",
"answer": "2016-04-29 02:53:44",
"end": "2016-04-29 02:53:49",
"duration": "12",
```

```
"billsec": "5",
"disposition": "ANSWERED",
"amaflags": "DOCUMENTATION",
"uniqueid": "1461920017.7",
"userfield": "Internal",
"channel_ext": "1000",
"dstchannel_ext": "1001",
"service": "s",
"caller_name": "John Doe",
"recordfiles": "auto-1461920017-1000-1001.wav@",
"dstanswer": "1001",
"chanext": "",
"dstchanext": "",
"session": "1461920017104140-1000",
"action_owner": "1000",
"action_type": "DIAL",
"src_trunk_name": "",
"dst_trunk_name": ""
},
"sub_cdr_2": {
"AcctId": "2",
"accountcode": "",
"src": "1000",
"dst": "1002",
"dcontext": "from-internal-xfer",
"clid": "\"John Doe\" <1000>",
"channel": "PJSIP/1000-00000003",
"dstchannel": "PJSIP/1002-00000005",
"lastapp": "Dial",
"lastdata": "PJSIP/1002/sip:1002@192.168.124.136:5066,60,ztTkKb(callee-
handler^s^1)",
"start": "2016-04-29 02:53:49",
"answer": "2016-04-29 02:53:49",
"end": "2016-04-29 02:53:54",
"duration": "4",
"billsec": "4",
"disposition": "ANSWERED",
"amaflags": "DOCUMENTATION",
"uniqueid": "1461920017.7",
"userfield": "Internal",
"channel_ext": "1000",
"dstchannel_ext": "1002",
"service": "s",
"caller_name": "John Doe",
"recordfiles": "",
"dstanswer": "1002",
"chanext": "",
"dstchanext": "",
"session": "1461920017104140-1000",
"action_owner": "1001",
"action_type": "TRANSFER",
"src_trunk_name": "",
"dst_trunk_name": ""
}
}
]
}
```

CSV Output

The following CSV output is for an internal call from extension 6039 to extension 6011.

```
AcctId,accountcode,src,dst,dcontext,clid,channel,dstchannel,lastapp,lastdata,start,answer,end,duration,billsec,disposition,amaflags,uniqueid,userfield,channel_ext,dstchannel_ext,service,caller_name,recordfiles,dstanswer,chanext,dstchanext,session,action_owner,action_type,src_trunk_name,dst_trunk_name
```

```
1201,,6039,6011,from-internal,"GVC Test" <6039>,PJSIP/6039-0000069e,PJSIP/6011-0000069f,Dial,"PJSIP/6011/sip:6011@218.17.227.162:5062&PJSIP/6011/sip:6011@218.17.227.163:38102,30",2015-11-10 17:58:47,,2015-11-10 17:58:51,4,0,NO ANSWER,DOCUMENTATION,1447207127.4182,Internal,6039,6011,s,GVC Test,,,,1447207127558984-6039,6039,DIAL,,
```

UCM6XXX CDR and REC API Guide

REC API – ACCESS CALL RECORDING FILES

REC API URL Format

The format of the HTTPS request for the REC API is shown as below. This is used to request call recordings, or a list of recording files within the Asterisk spool directories, matching given parameters as set by the third party application.

https://[UCM IP]:[Port]/recapi?[option1]=[value]&[option2]=[value]&...

By default, the port number for the API is 8443.

REC API URL Parameters

REC API takes two parameters: filedir and filename. Both parameters are optional, and the response depends on which parameters are included in the request.

- **<u>Case 1</u>**: Neither parameter is set.

  <u>Example Request:</u>

  https://192.168.254.200:8443/recapi

  <u>Response:</u>

  A CSV file listing all directories under ast_config_AST_SPOOL_DIR.

- **<u>Case 2</u>**: Only filedir is set.

  In this case, multiple file directories are supported, separated by '@'.

  <u>Example Request:</u>

  https://192.168.254.200:8443/recapi?filedir=monitor@meetme@voicemail/default

  <u>Response:</u>

  A CSV file listing the contents of each directory listed (relative to ast_config_AST_SPOOL_DIR).

- **<u>Case 3</u>**: Both filedir and filename are set.

  In this case, multiple file names are supported, separated by '@'; multiple file directories are not currently supported.

  <u>Example Request:</u>

  https://192.168.254.200:8443/recapi?filedir=meetme&filename=meetme-conf-rec-6300-

  1411501234.0-0.wav@meetme-conf-rec-6301-1411505678.9-0.wav

  <u>Response:</u>

  The matching WAV file (if only one valid file found, in WAV format).

  > -OR-

  A tarball containing all matching files, named [timestamp].tar, where the timestamp is set at the start of the API callback function.

-OR-

404: Not Found (if no matching files are found).

- **Case 4**: Only filename is set.

  This case is the same as **Case 3**, with the file directory defaulting to "monitor".

  Example Request:

  https://192.168.254.200:8443/recapi?filename=auto-1414771234-1000-1004.wav@auto-1414775678-1001-1003.wav

  Response:

  See **Case 3**.

⚠ **Notes:**

Requests formed incorrectly or with disallowed characters may result in an error, or cause the file to be skipped. Error cases include:

- Multiple occurrences of the same parameter (filedir or filename) in the POST variable list are not allowed. (404)

- Empty parameters in the POST variable list are not allowed. (404)

- File names longer than 256 characters will be truncated.

- File names containing ' ' (space) or '..' will be skipped.

UCM6XXX CDR and REC API Guide