A MITEL
PRODUCT
GUIDE

MiVoice MX-ONE

Data Interface (REST API) between the MiCollab Server and MiVoice MX-ONE, for Central Call History Interface - Interface Description

Release 7.6
66/15519-ANF 901 14 Uen B

December 2023

Mitel
Powering connections

Notices

The information contained in this document is believed to be accurate in all respects but is not warranted by **Mitel Networks Corporation (MITEL®).**The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes. No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

Trademarks

The trademarks, service marks, logos and graphics (collectively "Trademarks") appearing on Mitel's Internet sites or in its publications are registered and unregistered trademarks of Mitel Networks Corporation (MNC), its affiliates, parents, or subsidiaries (collectively "Mitel") or others. Use of the Trademarks is prohibited without the express consent from Mitel. Please contact our legal department at legal@mitel.com for additional information. For a list of the worldwide Mitel Networks Corporation registered trademarks, please refer to the website:http://www.mitel.com/trademarks.

Contents

Introduction 1

This interface description API document describes the message flows and data content for the Central Call History function. The document relates the data exchange between the Mitel MiCollab infrastructure, primarily the MiCollab Server, and the MiVoice MX-ONE PBXs.

This document is intended for internal use only.

The purpose of the Central Call History feature (abbreviated as CHLOG), is to provide every user of a SIP terminal/client with a log of calls received and placed, either answered or missed/failed.

By accessing the CHLOG feature, an end-user can:

- Browse the log
- Make calls to any stored number
- Delete entries that are no longer needed

The CHLOG feature comprises two different functions:

- CHLOG registration, which activates the central CHLOG feature for the extension.
    - The central CHLOG registration for SIP extension is done automatically after login or registration if the feature is enabled on the system. The extension will only accept call logs being pushed to the end-point if the MX-ONE has offered centralized CHLOG.
- CHLOG handling, which means accessing and/or manipulating the central CHLOG data.
    - The central CHLOG handling for SIP end-points is controlled by the terminal/client via keys or menus (depends on the end-point type/model). There are four types of logged calls; incoming, incoming-missed, outgoing, and outgoing-failed.

The system administrator can enable/disable this feature for a SIP terminal/client.

Overview, Reading Instructions, Limitation and Scope of the Document

2

This Interface description presented in this document is not complete and does not show all functions in detail; rather it describes the more important ones with some examples.

MiCollab Server (MCS) provides a REST API interface for adding the PBX data for Central Call History Log information to the MCS. The end-points might also have a local Call History function, but that must be turned off when the central Call History is active.

Authentication between PBX and MCS will use an authentication token. The token will be verified by the PBX and the MCS against the service.

Authentication And Security 3

This chapter contains the following sections:

3.1 Authentication Token

Authentication is provided by an authorization header.

Digest Authentication is proposed to be used for the authentication (same as for SIP and VDP). The recommended time interval for re-verification is proposed to be 30 minutes.

For more information, see https://en.wikipedia.org/wiki/Digest_access_authentication, and the RFC 7616.

Example:

HTTP header

"Authorization" : " Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJp
ZCI6IiIsIm5hbWUiOiIgIiwidHlwZSI6IlVzZXIiLCJlbWFpbCI6IiIsInBob3RvVXJsIjoiIiwiYXV0aFByb3ZpZGVyIjoidW5rbm93b
gPD-Q9KS2JhVYBm8"

The MCS and PBX will try to verify this token for validity. For performance reasons, the token is cached for 30 minutes, before it is re-verified.

The call history can be accessed by authenticating against the user's login credentials, giving only access to the user's call history. With the `command user_authentication`, the administrator can configure a user to access all generic extension call history.

3.1.1 Authentication, JSON Proposal

securitySchemes:

digestAuth:

type: http

scheme: digest

description: Authenticate with phone number and phone password

Preliminary JSON file:

```
{
  "openapi": "3.0.0",
  "info": {
    "title": "Call history",
    "description": "Requesting call history data from PBX",
    "version": "1.0.0"
  },
  "paths": {
    "/callHistory/{directoryNumber}": {
      "parameters": [
        {
          "name": "directoryNumber",
          "in": "path",
          "description": "Directory number to get call history for",
          "required": true,
          "schema": {
            "type": "string"
          }
        }
      ],
      "get": {
        "summary": "Get call history",
        "tags": [
          "Get call history"
        ],
        "parameters": [
          {
            "name": "startTime",
            "in": "query",
            "description": "optional start time, ISO date YYYY-MM-DD",
            "schema": {
              "type": "string"
            }
          },
          {
            "name": "stopTime",
            "in": "query",
            "description": "optional stop time, ISO date YYYY-MM-DD",
            "schema": {
              "type": "string"
            }
          }
        ],
        "responses": {
          "200": {
            "description": "successful operation",
            "content": {
              "application/json": {
                "schema": {
                  "type": "array",
                  "items": {
                    "$ref": "#/components/schemas/CallItem"
                  }
                }
              }
            }
          },
          "404": {
            "description": "Directory number not found"
          },
```

```
          "405": {
            "description": "Invalid input"
          },
          "407": {
            "description": "Authentication required"
          }
        },
        "security": [
          {
            "digestAuth": []
          }
        ]
      },
      "post": {
        "summary": "Enable/disable call history reporting",
        "tags": [
          "Subscribe to call history"
        ],
        "requestBody": {
          "required": true,
          "content": {
            "application/json": {
              "schema": {
                "type": "object",
                "properties": {
                  "action": {
                    "type": "string",
                    "enum": [
                      "startReporting",
                      "stopReporting"
                    ]
                  },
                  "callbackUrl": {
                    "type": "string",
                    "format": "uri",
                    "example": "https://server.com/callHistory/callEvent"
                  }
                },
                "required": [
                  "action",
                  "callbackUrl"
                ]
              }
            }
          }
        },
        "responses": {
          "201": {
            "description": "Subscription created"
          },
          "405": {
            "description": "Invalid input"
          },
          "407": {
            "description": "Authentication required"
          }
        },
        "callbacks": {
          "callEvent": {
            "{$request.body#/callbackUrl}": {
              "post": {
                "description": "Add call history item",
                "requestBody": {
                  "required": true,
```

```
                "content": {
                  "application/json": {
                    "schema": {
                      "type": "array",
                      "items": {
                        "$ref": "#/components/schemas/CallItem"
                      }
                    }
                  }
                }
              },
              "responses": {
                "200": {
                  "description": "Subscriber accepts the callback"
                }
              }
            },
            "delete": {
              "description": "Delete call history item",
              "parameters": [
                {
                  "name": "callIdentity",
                  "in": "query",
                  "description": "call identity of item to delete",
                  "required": true,
                  "schema": {
                    "type": "string"
                  }
                }
              ],
              "responses": {
                "200": {
                  "description": "Subscriber accepts the callback"
                }
              }
            }
          }
        },
        "security": [
          {
            "digestAuth": []
          }
        ]
      },
      "delete": {
        "summary": "Delete call history",
        "tags": [
          "Delete call history"
        ],
        "parameters": [
          {
            "name": "callIdentity",
            "in": "query",
            "description": "call identity of item to delete",
            "required": true,
            "schema": {
              "type": "string"
            }
          }
        ],
        "responses": {
          "200": {
            "description": "successful operation"
```

```
          },
          "404": {
            "description": "User number not found"
          },
          "405": {
            "description": "Invalid input"
          },
          "407": {
            "description": "Authentication required"
          }
        },
        "security": [
          {
            "digestAuth": []
          }
        ]
      }
    }
  },
  "components": {
    "schemas": {
      "CallItem": {
        "type": "object",
        "properties": {
          "directoryNumber": {
            "type": "string",
            "description": "Unique identity of a callee. Max 24 digits"
          },
          "dateTime": {
            "type": "string",
            "description": "Date and time of the logged call in ISO format.
Time is 24 h format only, in UTC."
          },
          "timeZone": {
            "type": "string",
            "description": "ISO time zone, \"GMT\""
          },
          "duration": {
            "type": "string",
            "description": "Length of the call in h:m:s"
          },
          "typeOfCall": {
            "type": "string",
            "enum": [
              "incoming-answered",
              "incoming-missed",
              "outgoing-answered",
              "outgoing-missed"
            ]
          },
          "transferredCall": {
            "type": "boolean",
            "default": false
          },
          "divertedCall": {
            "type": "boolean",
            "default": false
          }
        }
      }
    },
    "securitySchemes": {
      "digestAuth": {
        "type": "http",
```

```
        "scheme": "digest",
        "description": "Authenticate with phone number and phone passwd"
      }
    }
  }
}
```

3.2 PBX IP Address

A different approach could be to check whether the request is actually coming from the configured PBX IP address. The MCS has an active connection to the PBX that uses the same IP address. If token validation is not possible, we could use this approach. This is not the preferred approach.

3.3 HTTPS Server Certificate

HTTPS requires a server certificate to be installed on the PBX and the MCS. The same certificate must be uploaded to the MCS web portal and to the PBX portal. This enables the MCS to run HTTPS on port 22228 using the uploaded certificate. Only one certificate can be used for one specific port (22228) in the MCS.

Conventions and Terminology 4

The following conventions and terminology will be used in this document: Data will be added using JSON formatted objects and GET, PUT, POST and DELETE methods.

This is the base URL that is implemented in the REST API:https://<mxone-micollab>:22228/api/v1/mxoneCallHistoryApi/pbx/

GET method must be used for reading data. POST method must be used for creating data. PUT method must be used for changing data. DELETE method must be used for deleting data.

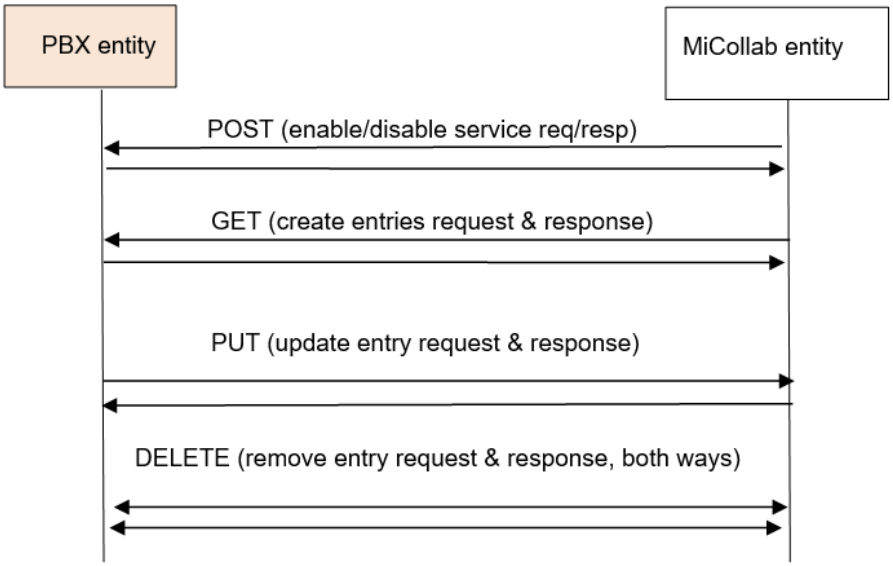
All requests will return a standard HTTP response, for example:

- 200 OK, the request was performed
- 400 Bad request; the data could not be parsed
- 401 Authentication failed (unauthorized)
- 403 Authentication rejected
- 404 User number not found
- 405 Invalid input
- 407 Authentication required

GET requests return additional data in the response body.

The messages shown in the following figure are used.

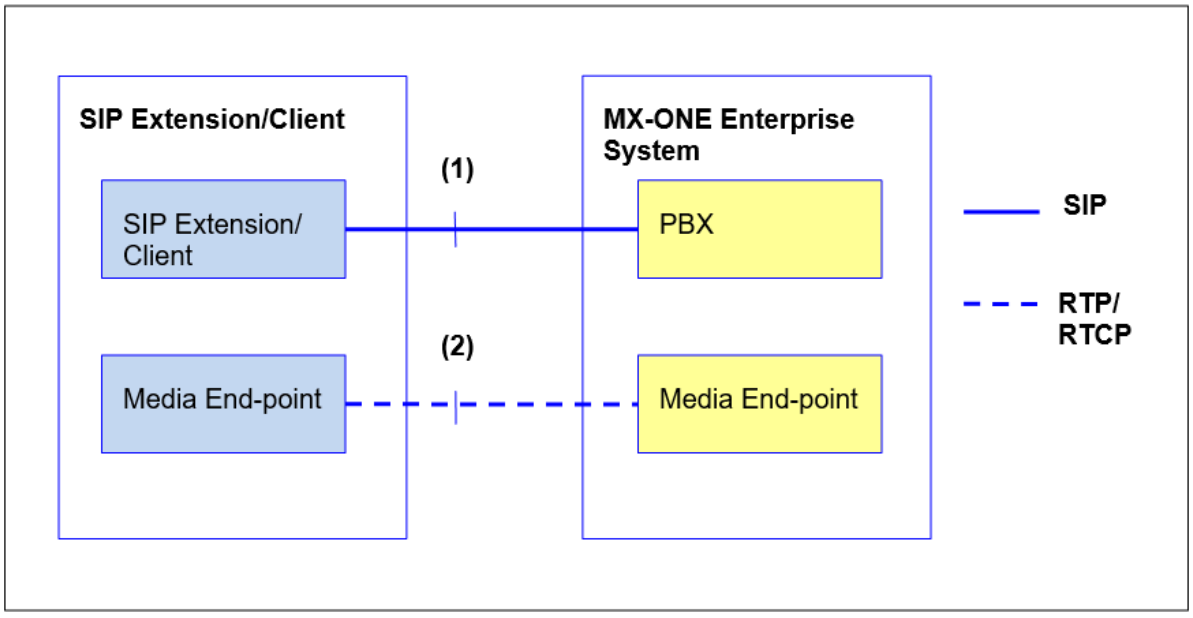
Figure 1: Central Call History messages



Reference Architecture 5

The reference architecture diagram in the following figure shows the common functional elements required to support the interface specification outlined by this document. The figure shows two reference points between the Enterprise system and the SIP based extension/client; reference point (1) and reference point (2). The Enterprise system has a SIP Proxy Server/Registrar and media gateways. The Central Call History API is part of reference point (1), and conveys data for the Central Call History log for extensions/clients.

Figure 2: Reference Architecture



Note that a single SIP-PBX may serve media endpoints in a number of geographically-distributed locations. One user (directory number) may have multiple endpoints (terminals/clients) registered.

Definitions 6

This chapter contains the following sections:

6.1 Glossary

SIP	Session Initiation Protocol, IETF standards for packet-based multimedia communication systems. See RFC 3261.
SIP Endpoint	SIP-EP, a term used in this document to refer to both SIP terminals/clients and SIP-PBXs (the media gateway).
SIP Extension	The SIP extension feature allows terminals that are compliant with SIP standards, IETF RFCs, to register in and connect to the MX-ONE system. These standards give recommendations for multimedia communications over IP networks. The term "IP extension" includes both H.323 and SIP extensions. The SIP extension is implemented as a generic extension. The SIP extension can be either single line access (have only one active call), or multi-line access (have several active calls). Which access type is valid depends on the terminal brand/model, and the configuration.
SIP-PBX	The Enterprise's point of SIP signaling inter-connection with the SIP extension/client. Here the MX-ONE System.

6.2 Acronyms

CHLog	Central Call History Log (in MX-ONE)
CSP	Common Service Profile (in MX-ONE)

EP	End Point (a terminal or client)
HTTP	HyperText Transfer Protocol
IWD	InterWorking Description (Interface Description)
MCS	MiCollab Server
ODN	Own Directory Number (main line)
PBX	Private Branch Exchange (Enterprise system)
RTP	Real-Time Protocol
SIP	Session Initiation Protocol
SRTP	Secure Real-Time Protocol

Call History Log 7

This chapter contains the following sections:

Call History Log (CHLog) is an end-user service, also known as "Call Log/Call List" and "Name and number log", which provides a list of calls received by or placed from the user's number. It might be valid for several EPs, if the user has multiple terminals/clients. The CHLog is stored centrally, in the PBX, and sent to the endpoints for presentation.

The service gives the ability for an extension/client to log calls; for example received, placed and missed calls. What is actually shown to the end-user is up to the EP.

7.1 Local Call History Log

If the log is local in the terminal/client, it is stored in the terminal/client based on the ordinary number and name information used in display functions. Local log is the default configuration, but must be turned off if central Call History is active. Any discussion about the Local Call History Log is outside the scope of this document. See the client/terminal documentation for details.

7.2 Central Call History Log

MX-ONE has the following central Call History Log limits with regard to the support for SIP extensions (currently support for Mitel 6800/6900 SIP phones, using a proprietary XML protocol and SIP signaling):

- Maximum 60 incoming calls (calls made to the EP, where missed calls are a subset of all incoming calls.)
- Maximum 30 outgoing calls (calls placed from the EP, where failed/rejected outgoing calls are a subset of all outgoing calls).

The same capacity is valid for this service (JSON-based CHLog).

7.2.1 Enabling/Disabling Central Call Log At Logon

When SIP REGISTER is received for an extension EP, the PBX will provision the terminal/client to use the Call History API service if the request comes from a supported EP (for example MiCollab), and is requested for a standard main line ("ODN"). The user registering must have a CSP profile with Central Call History Log (CHLog) enabled.

The PBX will expect an HTTPS:POST (or HTTP:POST) to activate/enable the CHLog service for the requesting user.

URL: https://<mxone-micollab>:22228/api/v1/mxoneCallHistoryApi/pbx/callHistoryLogs

Method: POST

Single user object:

Request body:

Table 1: POST (enable/disable CHLog service request) content

Field, information element	Type	Description
Service identity	string	Unique identity for the service CHLog.
Directory number	string	Unique identity of the requesting user. A number having a maximum of 20 digits for extensions.
Type of request	string	Enable/disable CHLog
DigestAuthentication type	string	Traditional Digest Authentication

Example 1, enable service:

{

"serviceIdentity":"1",

"extension": "20010",

"request": "Enable_CHLog",

"authenticationType": "TraditionalDigestAuthentication"

}

Example 2, disable service:

{

"serviceIdentity":"1",

"extension": "20010",

"request": "Disable_CHLog",

}

7.2.2 Creating Central Call Log (At Logon)

If the service is active and there are stored logs for the user, the stored logs are pushed via an HTTP: GET response, if a GET request is received by the PBX.

URL: https://<mxone-micollab>:22228/api/v1/mxoneCallHistoryApi/pbx/callHistoryLogs

Method: **GET (request)**

Single user object:

Request body:

Table 2: GET request (fetch CHLog) content

Field, information element	Type	Description
Directory number	string	Unique identity of the requesting us er. A number having a maximum of 20 digits for extensions.
Type of request	string	Enable/disable CHLog.
Scope of the request	string	All/Date range. Default is All.
Start date	string	ISO date YYYY-MM-DD. **Optional**
Stop date	string	ISO date YYY-MM-DD. **Optional**

Example:

{

"directoryNumber": "50011",

"requestType":"All"

}

Method: **GET (response)**

Response body:

Table 3: GET response (create CHLog) content

Field, information element	Type	Description
Item identity	string	Unique identity for the entry; an in teger number.
Directory number	string	Unique identity of the calling/called user. A number having a maximum of 20 digits for extensions; 40 digits for external numbers.

Field, information element	Type	Description
Name	string	Username of the Directory number. Maximum 40 characters. **Optional**
Call identity	string	Unique identity of a call. Max 24 di gits.
Date and time	string	Date and time of the logged call in ISO format. Time is in 24-hour forma t only, in UTC. (Receiver must calcu late local time for presentation to the user).
Time zone	string	ISO time zones, for the user of the CHlog.
Call duration	string	Length of the call in h:m:s
Type of call	string	Incoming-answered, incoming-miss ed, outgoing, outgoing-failed.
Transferred call	bool	true/false **Optional**
Diverted call	bool	true/false **Optional**
First dialed number	string	For example, at Diversion, Deflect. A number having a maximum of 20 digits. **Optional**
Remote number	string	Public subscriber number. A number which is max 20 digits. **Optional**
Directory number 2	string	Unique identity of e.g. the picking/ redirect-to user. A number having a maximum of 20 digits for extensions; 40 digits for external numbers. **Optional**
Name 2	string	Username of the Directory number 2. Maximum 40 characters. **Optional**

Field, information element	Type	Description
Info text 2	string	Information text/reason about directory number 2. number 2. Maximum 20 characters. **Optional**

Example, for a single log entry. Repeated for each entry:

{

"EntryIdentity":8

"directoryNumber": "50011",

"name":"John Smith",

"callIdentity": "126541134989005432104321",

"dateAndTime": "2021-10-21 13:23" ,

"timeZone": "GMT",

"callDuration": "00:03:12",

"typeOfCallLog":"incoming-answered",

"transferred":false

}

7.2.3 Updating Central Call History Log

When an extension that supports Central Call History Log receives or makes a call, the log must be updated; that is, the endpoint be informed of the new call.

When SIP INVITE is received/sent for an extension EP, the PBX will inform the extensions that support Central Call History Log receives or makes a call, the log is updated; that is, the endpoint be informed of the new call in the log.

URL: https://<mxone-micollab>:22228/api/v1/mxoneCallHistoryApi/pbx/callHistoryLogs

Method: **POST**

Single user object:

Request body:

Table 4: POST (update CHLog) content

Field, information element	Type	Description
Directory number	string	Unique identity of the calling/called user. A number having a maximum of 20 digits for extensions, 40 digits for external numbers.
Name	string	User name of the Directory number. Maximum 40 characters. **Optional**
Call identity	string	Unique identity of a call. Maximum 2 4 digits.
Date and time	string	Date and time of the logged call in ISO format. Time is 24-hour format o nly, in UTC. (Receiver must calculat e local time for presentation to the user).
Time zone	string	ISO time zone, for the user of the C HLog.
Call duration	string	Length of the call in h:m:s
Type of call	string	Incoming-answered, incoming-miss ed, outgoing, outgoing-failed.
Transferred call	bool	true/false **Optional**
Diverted call	bool	true/false **Optional**
First dialed number	string	For example, at Diversion, Deflect. A number having a maximum of 20 digits. **Optional**
Remote number	string	Public subscriber number. A number which is max 20 digits. **Optional**
Directory number 2	string	Unique identity of e.g. the picking/ redirect-to user. A number having a maximum of 20 digits for extensions; 40 digits for external numbers. **Optional**

Field, information element	Type	Description
Name 2	string	Username of the Directory number 2. Maximum 40 characters. **Optional**
Info text 2	string	Information text/reason about directory number 2. number 2. Maximum 20 characters. **Optional**

Example, only one entry:

{

"directoryNumber": "50011",

"name":"John Smith",

"callIdentity": "126541134989005432104338",

"dateAndTime": "2021-10-21 13:39" ,

"timeZone": "GMT",

"callDuration": "00:03:22",

"typeOfCallLog":"incoming-answered",

"transferred":true

}

7.2.4 Deleting A Central Call History Log Entry

When an extension that supports centralized Call History Log requests to erase one specific log entry, or erase all entries, manually or in certain use cases, that shall be possible. An example for all entries must be erased is at logoff or checkout for Hospitality users. The Delete method can be used both ways; sent from the PBX or sent from the MCS.

URL: https://<mxone-micollab>:22228/api/v1/mxoneCallHistoryApi/pbx/callHistoryLogs

Method: **DELETE**

Single user object:

Request body:

Table 5: DELETE (erase CHLog) content

Field, information element	Type	Description
Directory number	string	Unique identity of the calling/called user. A number having a maximum of 20 digits for extensions, 40 digits for external numbers.
Call identity	string	Unique identity of a call history item to delete. Maximum 24 digits/characters. Also, values all, all-outgoing, all-incoming and all-incoming-missed are supported.

Example:

{

"directoryNumber": "50013",

"callIdentity": "126541134989005432104341"

}

Some Use Case Examples 8

This chapter contains the following sections:

The use cases below are not complete, but cover some common cases, as seen from an extension/client user, who has the CHLog function active, and supported.

8.1 Registration (Logon) Of An Endpoint

When an EP (terminal or client) is registered (logged on), and its profile allows the CHLog feature, then that feature must be activated, and possible existing log entries shall be conveyed to the EP, for presentation to the end-user.

8.2 A Successfully Received Call Is Terminated

When an EP (terminal or client) receives an incoming call, and the CHLog feature is active, the CHLog must be updated with the new call's data when the call is terminated (disconnected). This will be logged as a received answered call.

8.3 A Received Call Is Never Answered (Missed)

When an EP (terminal or client) receives an incoming call, which is alerting or queueing, and the CHLog feature is active, the CHLog must be updated with the new call's data when the call is terminated (disconnected), by the caller or on timeout. This will be logged as a received missed call.

There are also other services, such as Diversion, Deflect, and Pickup, that can cause a call to be logged as "missed".

8.4 A Successful Outgoing Call Is Made, And Terminated

When an EP (terminal or client) makes an outbound call, and the CHLog feature is active, the CHLog must be updated with the new call's data when the call is terminated (disconnected). This will be logged as a successful answered outbound call.

8.5 A Made Outgoing Call Fails

When an EP (terminal or client) makes an outbound call, and the CHLog feature is active, the CHLog must be updated with the new call's data when the call is terminated (disconnected or rejected). This will be logged as an unsuccessful/failed outbound call.

8.6 De-registration (Logoff) Of An Endpoint

When an EP (terminal or client) is de-registered (logged off), and its profile allows the CHLog feature, that feature is in some cases, deactivated, and possible existing log entries must be erased for the user and its EPs. Normal extensions, or extensions with multiple EPs are not erased from the CHLog at de-registration. However, in some cases, for example, for Hospitality class extension users, the entire CHLog is erased.

SIP Ports Used 9

Table 6: SIP Ports Used

Protocol	Port number	Comments
JSON for CHLog	22228	HTTPS, for Central Call History, sec ure
JSON for CHLog	22227	HTTP, for Central Call History, non-secure

References 10

This chapter contains the following sections:

10.1 Internal and CPI Documents

- FS 'Name and Number Log', 449/15517-ANF 901 14 Uen (Centralized part, for SIP ext)
- IWD 'SIP extension interface', 64/15519-ANF 901 14 Uen
- MXO-4600, Jira ticket on Central Call History log for MiCollab in MX-ONE 7.4 SP1
- Name and Number Log, centralized Operational Directions, 38/154 31-ANF 901 14 Uen
- The extension_profile command, parameter -ext-cnnlog, 201/19082-ANF 901 14 Uen

10.2 Standards, RFCs

- RFC 3261, Session Initiation Protocol
- RFC 7230-7232 & 7234, Hypertext Transfer Protocol - HTTP/1.1 (replaces RFC 2616)
- RFC 7235, Hypertext Transfer Protocol (HTTP/1.1): Authentication (replaces RFC 2616)
- RFC 7616, HTTP Digest Access Authentication (SHA-256)
- RFC 7617, The 'Basic' HTTP Authentication Scheme (replaces RFC 2617, MD5)