



Avaya Experience Portal 8.1 Automation Web Services Reference Guide

Release 8.1.1
Issue 1
Nov 2021

Notice

While reasonable efforts have been made to ensure that the information in this document is complete and accurate at the time of printing, Avaya assumes no liability for any errors. Avaya reserves the right to make changes and corrections to the information in this document without the obligation to notify any person or organization of such changes.

Documentation disclaimer

"Documentation" means information published in varying mediums which may include product information, operating instructions and performance specifications that are generally made available to users of products. Documentation does not include marketing materials. Avaya shall not be responsible for any modifications, additions, or deletions to the original published version of Documentation unless such modifications, additions, or deletions were performed by or on the express behalf of Avaya. End User agrees to indemnify and hold harmless Avaya, Avaya's agents, servants and employees against all claims, lawsuits, demands and judgments arising out of, or in connection with, subsequent modifications, additions or deletions to this documentation, to the extent made by End User.

Link disclaimer

Avaya is not responsible for the contents or reliability of any linked websites referenced within this site or Documentation provided by Avaya. Avaya is not responsible for the accuracy of any information, statement or content provided on these sites and does not necessarily endorse the products, services, or information described or offered within them. Avaya does not guarantee that these links will work all the time and has no control over the availability of the linked pages.

Warranty

Avaya provides a limited warranty on Avaya hardware and software. Refer to your sales agreement to establish the terms of the limited warranty. In addition, Avaya's standard warranty language, as well as information regarding support for this product while under warranty is available to Avaya customers and other parties through the Avaya Support website: <https://support.avaya.com/helpcenter/getGenericDetails?detailId=C20091120112456651010> under the link "Warranty & Product Lifecycle" or such successor site as designated by Avaya. Please note that if You acquired the product(s) from an authorized Avaya Channel Partner outside of the United States and Canada, the warranty is provided to You by said Avaya Channel Partner and not by Avaya.

Licenses

THE SOFTWARE LICENSE TERMS AVAILABLE ON THE AVAYA WEBSITE, [HTTPS://SUPPORT.AVAYA.COM/LICENSEINFO](https://support.avaya.com/licenseinfo), UNDER THE LINK "AVAYA SOFTWARE LICENSE TERMS (Avaya Products)" OR SUCH SUCCESSOR SITE AS DESIGNATED BY AVAYA, ARE APPLICABLE TO ANYONE WHO DOWNLOADS, USES AND/OR INSTALLS AVAYA SOFTWARE, PURCHASED FROM AVAYA INC., ANY AVAYA AFFILIATE, OR AN AVAYA CHANNEL PARTNER (AS APPLICABLE) UNDER A COMMERCIAL AGREEMENT WITH AVAYA OR AN AVAYA CHANNEL PARTNER. UNLESS OTHERWISE AGREED TO BY AVAYA IN WRITING, AVAYA DOES NOT EXTEND THIS LICENSE IF THE SOFTWARE WAS OBTAINED FROM ANYONE OTHER THAN AVAYA, AN AVAYA AFFILIATE OR AN AVAYA CHANNEL PARTNER; AVAYA RESERVES THE RIGHT TO TAKE LEGAL ACTION AGAINST YOU AND ANYONE ELSE USING OR SELLING THE SOFTWARE WITHOUT A LICENSE. BY INSTALLING, DOWNLOADING OR USING THE SOFTWARE, OR AUTHORIZING OTHERS TO DO SO, YOU, ON BEHALF OF YOURSELF AND THE ENTITY FOR WHOM YOU ARE INSTALLING, DOWNLOADING OR USING THE SOFTWARE (HEREINAFTER REFERRED TO INTERCHANGEABLY AS "YOU" AND "END USER"), AGREE TO THESE TERMS AND CONDITIONS AND CREATE A BINDING CONTRACT BETWEEN YOU AND AVAYA INC. OR THE APPLICABLE AVAYA AFFILIATE ("AVAYA").

Avaya grants You a license within the scope of the license types described below, with the exception of Heritage Nortel Software, for which the scope of the license is detailed below. Where the order

documentation does not expressly identify a license type, the applicable license will be a Designated System License as set forth below in the Designated System(s) License (DS) section as applicable. The applicable number of licenses and units of capacity for which the license is granted will be one (1), unless a different number of licenses or units of capacity is specified in the documentation or other materials available to You. "Software" means computer programs in object code, provided by Avaya or an Avaya Channel Partner, whether as stand-alone products, pre-installed on hardware products, and any upgrades, updates, patches, bug fixes, or modified versions thereto. "Designated Processor" means a single stand-alone computing device. "Server" means a set of Designated Processors that hosts (physically or virtually) a software application to be accessed by multiple users. "Instance" means a single copy of the Software executing at a particular time: (i) on one physical machine; or (ii) on one deployed software virtual machine ("VM") or similar deployment.

License types

Concurrent User License (CU). End User may install and use the Software on multiple Designated Processors or one or more Servers, so long as only the licensed number of Units are accessing and using the Software at any given time. A "Unit" means the unit on which Avaya, at its sole discretion, bases the pricing of its licenses and can be, without limitation, an agent, port or user, an e-mail or voice mail account in the name of a person or corporate function (e.g., webmaster or helpdesk), or a directory entry in the administrative database utilized by the Software that permits one user to interface with the Software. Units may be linked to a specific, identified Server or an Instance of the Software.

Shrinkwrap License (SR). You may install and use the Software in accordance with the terms and conditions of the applicable license agreements, such as "shrinkwrap" or "clickthrough" license accompanying or applicable to the Software ("Shrinkwrap License").

Copyright

Except where expressly stated otherwise, no use should be made of materials on this site, the Documentation, Software, Hosted Service, or hardware provided by Avaya. All content on this site, the documentation, Hosted Service, and the product provided by Avaya including the selection, arrangement and design of the content is owned either by Avaya or its licensors and is protected by copyright and other intellectual property laws including the sui generis rights relating to the protection of databases. You may not modify, copy, reproduce, republish, upload, post, transmit or distribute in any way any content, in whole or in part, including any code and software unless expressly authorized by Avaya. Unauthorized reproduction, transmission, dissemination, storage, and or use without the express written consent of Avaya can be a criminal, as well as a civil offense under the applicable law.

Virtualization

The following applies if the product is deployed on a virtual machine. Each product has its own ordering code and license types. Unless otherwise stated, each Instance of a product must be separately licensed and ordered. For example, if the end user customer or Avaya Channel Partner would like to install two Instances of the same type of products, then two products of that type must be ordered.

Third Party Components

"Third Party Components" mean certain software programs or portions thereof included in the Software or Hosted Service may contain software (including open source software) distributed under third party agreements ("Third Party Components"), which contain terms regarding the rights to use certain portions of the Software ("Third Party Terms"). As required, information regarding distributed Linux OS source code (for those products that have distributed Linux OS source code) and identifying the copyright holders of the Third Party Components and the Third Party Terms that apply is available in the products, Documentation or on Avaya's website at: <https://support.avaya.com/Copyright> or such successor site as designated by Avaya. The open source software license terms provided as Third Party Terms are consistent with the license rights granted in these Software License Terms, and may contain additional rights benefiting You, such as modification and distribution of the open source software. The Third Party Terms shall take precedence over these

Software License Terms, solely with respect to the applicable Third Party Components to the extent that these Software License Terms impose greater restrictions on You than the applicable Third Party Terms.

Preventing Toll Fraud

"Toll Fraud" is the unauthorized use of your telecommunications system by an unauthorized party (for example, a person who is not a corporate employee, agent, subcontractor, or is not working on your company's behalf). Be aware that there can be a risk of Toll Fraud associated with your system and that, if Toll Fraud occurs, it can result in substantial additional charges for your telecommunications services.

Avaya Toll Fraud intervention

If You suspect that You are being victimized by Toll Fraud and You need technical assistance or support, call Technical Service Center Toll Fraud Intervention Hotline at +1-800-643-2353 for the United States and Canada. For additional support telephone numbers, see the Avaya Support website: <https://support.avaya.com> or such successor site as designated by Avaya.

Trademarks

Avaya, the Avaya logo, Avaya Experience Portal, Avaya Aura® Communication Manager, and Avaya Orchestration Designer are either registered trademarks or trademarks of Avaya Inc. in the United States of America and/or other jurisdictions.

All non-Avaya trademarks are the property of their respective owners. Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Downloading Documentation

For the most current versions of Documentation, see the Avaya Support website: <https://support.avaya.com>, or such successor site as designated by Avaya.

Contact Avaya Support

See the Avaya Support website: <https://support.avaya.com> for product or Hosted Service notices and articles, or to report a problem with your Avaya product or Hosted Service. For a list of support telephone numbers and contact addresses, go to the Avaya Support website: <https://support.avaya.com> (or such successor site as designated by Avaya), scroll to the bottom of the page, and select Contact Avaya Support.

Contents

Chapter 1: Introduction	7
Purpose.....	7
Audience	7
Related Documents.....	7
Chapter 2: Avaya Experience Portal Web Services overview.....	8
Overview	8
Prerequisites	8
SOAP Web Services	8
Getting Started	9
Swagger UI	9
Client URL (curl).....	9
Chapter 3: Avaya Experience Portal Automation REST Web Services definitions	10
Overview	10
Configuring ASR Servers	10
Overview	10
GET - Retrieve the number of configured ASR servers	10
GET - Retrieve ASR Server configuration	11
POST - Add a new ASR server.....	11
PUT - Update ASR server configuration	12
DELETE ASR server.....	13
Configuring TTS Servers.....	16
Overview	16
GET - Retrieve the number of configured TTS servers.....	16
GET - Retrieve TTS Server configuration	16
POST - Add a new TTS server.....	17
PUT - Update TTS server configuration	18
DELETE TTS server	19
Input Values	19
Add and Configure an MPP Server	22
Overview	22
POST - Add MPP.....	22
PUT - Update MPP	25
Input Values	27
Deleting an MPP	30
Overview	30
DELETE – Delete an MPP server by name	30
Change the mode of an MPP server	31
Overview	31
PUT – Set the mode of an MPP server	31
GET – Get the mode of an MPP server	32
Change the state of an MPP Server.....	33
Overview	33
PUT – Change the state of an MPP server	33
GET – Get MPP server state	34
Configuring SIP Connections	35
Overview	35
POST – Create a SIP connection.....	35
PUT - Update SIP connection	36
DELETE – Delete a SIP connection.....	36
GET – Get SIP connection parameters by name	37
GET - Get list of all SIP connections.....	38
GET - Get SIP connections count	38
Identity Certificates.....	41
Overview	41
Pre-conditions	41
POST – Uploading an Identify Certificate.....	41
GET - Get Identity Certificate	42
Example	42

Configuring an External DB on the Primary EPM.....	44
Overview	44
POST - Set up External Report Database.....	44
PUT - Update External Reporting Database	45
GET - Get External Report Database.....	46
DELETE - Delete External Report Database	46
Configuring System Backup on Primary EPM	48
Overview	48
GET - Get System Backup Configuration.....	48
PUT - Update System Backup Configuration	49
GET - Get System Backup Customization Folders and Files.....	49
PUT - Update System Backup Customization Folders and Files	50
DELETE - Delete System Backup Customization Folders and Files.....	51
GET - Get System Backup Schedule	51
PUT - Update System Backup Schedule	52
DELETE - Delete System Backup Schedule.....	53
GET - Get System Backup History.....	54
POST - Start an On-Demand System Backup	55
Input Values	55
Trusted Certificates	57
Overview	57
POST - Import Trusted Certificate.....	57
POST - Upload Trusted Certificate - PEM.....	58
POST - Upload Trusted Certificate - PKCS#12.....	59
GET - Get Trusted Certificate - Count.....	59
GET - Get Trusted Certificate(s) - Names.....	60
DELETE - Delete Trusted Certificate	61
Configure SNMP Agent settings.....	63
Overview	63
PUT - Configure SNMP Agent.....	63
Request URL.....	63
Curl command.....	64
Known issues.....	64
Update the Operation Grace Period.....	66
Overview	66
Configuring EPM Conversation Store.....	66
Overview	66
PUT - Update Conversation Store.....	67
GET – Retrieve Conversation Store Parameters	67
Configuring LDAP Settings.....	69
Overview	69
GET.....	69
PUT.....	70
Configuring SSO Keycloak Settings.....	75
Overview	75
GET.....	75
PUT.....	76
Configuring MPP Settings – Resource & Trace Levels	80
Overview	80
GET – Resource Alert Threshold	80
PUT – Resource Alert Threshold	80
GET – Trace Logger	81
PUT – Trace Logger.....	82
GET – Transcription Retention Period	82
PUT – Transcription Retention Period.....	82
GET – Record Handling.....	83
PUT – Record Handling	83
GET – Category Trace Level	84
PUT – Category Trace Level.....	85
Retrieve list of EPMs.....	87
Overview	87
GET.....	87

Chapter 4: Resources88

Documentation	88
---------------------	----

Finding documents on the Avaya Support website	89
Avaya Documentation Center navigation	90
Viewing Avaya Mentor videos	91
Support.....	91

Chapter 1: Introduction

Purpose

This document describes the web services available with Avaya Experience Portal 8.1.

The audience includes and is not limited to application developers, administrators, business partners and solution providers.

Audience

The audience of the Avaya Experience Portal 8.1 web services reference guide includes and is not limited to:

1. Application developers
2. Administrators
3. Business partners
4. Solution Providers

Related Documents

The following documents, relevant to your Avaya Experience Portal deployment, must be consulted before the invocation of the Avaya Experience Portal 8.1 web services. Download the documents from the Avaya Support website at <http://support.avaya.com>.

1. Administering Avaya Experience Portal
2. Implementing Avaya Experience Portal on multiple servers
3. Implementing Avaya Experience Portal on a single server
4. Troubleshooting Avaya Experience Portal

Chapter 2: Avaya Experience Portal Web Services overview

Overview

The Avaya Experience Portal web services provide the ability to retrieve, add, modify and delete Avaya Experience Portal configuration elements.

Web services allow customers and integrators to configure and administer the Avaya Experience Portal configuration using third party tools and applications. It also allows integrators to develop tools for performing bulk operations.

Experience Portal supports upgrades without service interruption and without loss of data. To support an upgrade without service interruption the system must have more than a single MPP. There will be no loss of configuration or report data due to the upgrade. With a single MPP, there will be service interruption but there will not be any loss of data. With multiple MPPs, it will be possible to perform the upgrade with no interruption of the following services:

Important:

Invocation of the web services is asynchronous. The response of the web service request is sent asynchronously indicating that the request has been received and the instructions requested from the Avaya Experience Portal. It's important to note that the Avaya Experience Portal may get delayed in implementing the web service request depending on the complexity of commands, system load and network latency. It is recommended to utilize the available retrieve web services to validate any add, modify or delete requests. Please note that the available retrieve web services should not be invoked as a polling or heartbeat mechanism as this can detrimentally impact the database.

Prerequisites

The order in which the Avaya Experience Portal web services are invoked must adhere to the sequencing as defined in the Avaya Experience Portal installation, configuration and administration documents referenced in the **Related Documents** section of this document.

SOAP Web Services

The complete Avaya Experience Portal web services functionality includes a set of SOAP web services which are documented in the Administering Avaya Experience Portal document as referenced in the **Related Documents** section of this document.

Important:

For details on the existing Avaya Experience Portal SOAP web services refer to the following sections in the Administering Experience Portal document available at <http://support.avaya.com>

1. The Application Interface web service
2. The Management Interface web service

Getting Started

Swagger UI

The ability to visualize and interact with the available Avaya Experience Portal RESTful web services is available by navigating to the Swagger UI of the primary EPM and providing the username and password of the EPM UI e.g. [https://\[EPM IP\]/EPWebServices/rest-api/](https://[EPM IP]/EPWebServices/rest-api/).

Client URL (curl)

Each of the web services can be invoked via Client URL (curl). The curl examples for each of the web services can be expanded to enforce a secure connection by using the keyword `--cacert` and passing the location of a certificate, as well as the associated username and password. The following is one example of how this can be achieved. This example is a secure curl command for adding a TTS Server, where the `/root/cacert.pem` is the location of the Certificate Authority Certificate specific to your environment. The username and password fields are specific to the certificate being used.

```
curl -X GET "https://<EPM IP
Address>/EPWebServices/rest/management/ttsServers/count"-H
"accept: */*" --cacert /root/cacert.pem -u [username]:[password]
```

Chapter 3: Avaya Experience Portal Automation REST Web Services definitions

Overview

The following sections define the Avaya Experience Portal REST web services.

Configuring ASR Servers

Overview

The purpose of this feature is to provide a REST API where ASR servers can be managed using Experience Portal's web services. The ASR Servers web service can be found at `https://<EPM IP address>/EPWebServices/rest-api/#/ASR%20Servers`

Users can manage ASR servers in the following ways:

1. GET request to return the number of configured ASR servers.
2. GET request to retrieve a specific ASR server's configuration using its name.
3. POST request to add a new ASR server.
4. PUT request to update an existing ASR server's configuration using its name.
5. DELETE request to delete an ASR server using its name.

GET - Retrieve the number of configured ASR servers

This web service can be found at: `https://<EPM IP address>/EPWebServices/rest-api/#/ASR%20Servers/getCountOfSpeechServers`

This web service has no parameters, the user selects 'Try it out' then execute and the total number of configured ASR servers is returned.

Example:

Curl

```
curl -X GET "https://<EPM IP address>/EPWebServices/rest/management/asrServers/count"-H "accept: */*"
```

Request URL

```
https:// <EPM IP address>/EPWebServices/rest/management/asrServers/count
```

GET - Retrieve ASR Server configuration

This web service can be found at: `https://<EPM IP address>EPWebServices/rest-api/#/ASR%20Servers/getSpeechServerByName`

This web service takes one parameter, the user must enter a configured ASR server's name and select execute. The ASR server's configuration is returned.

Example:

User input:

Name: ASR1

Curl

```
curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/asrServers/?name=ASR1" -H "accept: */*"
```

Request URL

```
https:// <EPM IP Address>/EPWebServices/rest/management/asrServers/?name=ASR1
```

POST - Add a new ASR server

This web service can be found at: `https://<EPM IP address>/EPWebServices/rest-api/#/ASR%20Servers/createSpeechServer`

This web service provides a template which the user must edit with their desired ASR server configuration.

After the user has edited this template with valid values and submitted them using execute, a new ASR server will appear in EPM's Speech Server section.

Example:

```
{
  "name": "ASR2",
  "zone": "0",
  "enable": true,
  "engineType": "Loquendo",
  "data": {
    "networkAddress": "1.2.3.4",
    "basePort": 554,
    "port": 1,
    "perPort": false,
    "MRCP": {
      "activityTimer": 15,
      "responseTimer": 4,
      "mrcpProtocol": "mrcpv1",
      "mrcpProtocolData": {
        "rtspUrl": "1.2.3.4/media/speechrecognizer"
      }
    }
  },
  "selectedLang": [
```

```
"en-us English(USA) "  
]  
}  
}
```

Curl

```
curl -X POST "https://<EPM IP  
address>/EPWebServices/rest/management/asrServers/" -H "accept: */*" -H "Content-Type: application/json" -d  
"{\"name\": \"ASR2\", \"zone\": \"0\", \"enable\": true, \"engineType\": \"Loquendo\", \"data\": {\"networkAddress\": \"1.2.3.4\", \"basePort\": 554, \"port\": 1, \"perPort\": false, \"MRCP\": {\"activityTimer\": 15, \"responseTimer\": 4, \"mrcpProtocol\": \"mrcpv1\", \"mrcpProtocolData\": {\"rtspUrl\": \"1.2.3.4/media/speechrecognizer\"}}, \"selectedLang\": [\"en-us English(USA)\"]}"
```

Request URL

```
https://<EPM IP address>/EPWebServices/rest/management/asrServers/
```

PUT - Update ASR server configuration

This web service can be found at: <https://<EPM IP address>/EPWebServices/rest-api/#/ASR%20Servers/updateSpeechServerByName>

This web service takes one parameter and provides a template for the user to update with their desired ASR server configuration.

The user will input the configured ASR server name and use the template to update fields.

After the user has edited this template and submitted it using execute, the ASR server's configuration will be updated.

Example:

```
{  
  "name": "ASR2",  
  "zone": "0",  
  "enable": true,  
  "engineType": "Loquendo",  
  "data": {  
    "networkAddress": "127.0.0.1",  
    "basePort": 4900,  
    "port": 1,  
    "perPort": false,  
    "MRCP": {  
      "activityTimer": 15,  
      "responseTimer": 4,  
      "mrcpProtocol": "mrcpv1",  
      "mrcpProtocolData": {  
        "rtspUrl": "127.0.0.1/media/speechrecognizer"  
      }  
    },  
    "selectedLang": [  
      "en-us English(USA) "  
    ]  
  }  
}
```

```
}  
}
```

Curl

```
curl -X PUT "https://<EPM IP  
address>/EPWebServices/rest/management/asrServers/?name=ASR2" -H  
"accept: */*" -H "Content-Type: application/json" -d  
"{\"name\": \"ASR2\", \"zone\": \"0\", \"enable\": true, \"engineType\": \"Lo  
quendo\", \"data\": {\"networkAddress\": \"127.0.0.1\", \"basePort\": 4900,  
\"port\": 1, \"perPort\": false, \"MRCP\": {\"activityTimer\": 15, \"response  
Timer\": 4, \"mrcpProtocol\": \"mrcpv1\", \"mrcpProtocolData\": {\"rtspUrl\  
\": \"127.0.0.1/media/speechrecognizer\"}}, \"selectedLang\": [\"en-us  
English(USA)\"]}"
```

Request URL

```
https://<EPM IP  
Address>/EPWebServices/rest/management/asrServers/?name=ASR2
```

DELETE ASR server

This web service can be found at: <https://<EPM IP address>/EPWebServices/rest-api/#/ASR%20Servers/deleteSpeechServerByName>

This web service takes one parameter, the configured ASR server's name. After entering the name and selecting execute, the selected ASR server will be deleted.

Example:

Curl

```
curl -X DELETE "https://<EPM IP  
Address>/EPWebServices/rest/management/asrServers/?name=ASR1"-H  
"accept: */*"
```

Request URL

```
https://<EPM IP  
Address>/EPWebServices/rest/management/asrServers/?name=ASR1
```

Input Values

The ASR DTO (Data Transfer Object) properties:

DTO	Property	Description	Type/Format	Default
ASR Server	Name	Name	String	ASR1
	Zone	Zone	String	0
	Enable	Enable	Boolean	True
	EngineType	Engine Type	Enumeration (String: Nuance/Loquendo/DialogFlow/GoogleSpeech)	Nuance
	NetworkAddress	Network Address	String	127.0.0.1
	BasePort	Base Port	int32	4900

	Port	Total Number of Licensed TTS Resources	int32	1
	perPort	New Connection per Session	Boolean	False
	MRCP	See table below		
	selectedVoices	Selected Voices	(String) Array	en-US, Jennifer, F English(USA)

Other DTOs used by the ASR web service:

DTO	Property	Description	Type/Format	Default
MRCP	activityTimer	Ping Interval	int32	15
	responseTimer	Response Timeout	int32	4
	mrCPProtocol	Protocol	Enumeration (DTO: mrCPv1/mrCPv2)	mrCPv1
MRCPv1	rtspURI	RTSP URL	String	<networkAddress>/media/speechrecognizer
MRCPv2	sessionXML	Enable Session XML	Boolean	False
	transport	Transport Protocol	Enumeration (DTO: TCP/TLS)	TCP
TCP	localPort	Listener Port	int32	5060
TLS	localPort	Listener Port	int32	5061
	S RTP		DTO	
S RTP	Priority	Priority	int32	0

	srtpEnable	Enable	Boolean	true
	srtpEncryptAlgorithm	Encryption Algorithm	Enumeration (String: AES_CM_128/None)	AES_CM_128
	srtpAuthAlgorithm	Authentication Algorithm	Enumeration (String: HMAC_SHA1_80 / HMAC_SHA1_32)	HMAC_SHA1_80
	rtcpEncryptEnable	RTCP Encryption Enabled	Boolean	False
	rtpAuthEnable	RTP Authentication Enabled	Boolean	True
ASRDialogFlow	credentialFileName		String	N/A
	credentials		String	N/A
	audioChunkSize		Integer (int32)	8
ASRGoogleSpeech	credentialFileName		String	N/A
	credentials		String	N/A
	profanityFilter		Boolean	True
	audioChunkSize		Integer (int32)	8

Configuring TTS Servers

Overview

The purpose of this feature is to provide a REST API where TTS servers can be managed using Experience Portal's web services. The TTS Servers web service can be found at `https://<EPM IP address>/EPWebServices/rest-api/#/TTS%20Servers`

Users can manage TTS servers in the following ways:

1. GET request to return the number of configured TTS servers.
2. GET request to retrieve a specific TTS server's configuration using its name.
3. POST request to add a new TTS server.
4. PUT request to update an existing TTS server's configuration using its name.
5. DELETE request to delete an TTS server using its name.

GET - Retrieve the number of configured TTS servers

This web service can be found at: `https://<EPM IP address>/EPWebServices/rest-api/#/TTS%20Servers/getCountOfSpeechServers`

This web service has no parameters, the user selects 'Try it out' then execute and the total number of configured TTS servers is returned.

Example:

Curl

```
curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/ttsServers/count"-H "accept: */*"
```

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/ttsServers/count
```

GET - Retrieve TTS Server configuration

This web service can be found at: `https://<EPM IP address>EPWebServices/rest-api/#/TTS%20Servers/getSpeechServerByName`

This web service takes one parameter, the user must enter a configured TTS server's name and select execute. The TTS server's configuration is returned.

Example:

User input:

Name: TTS1

Curl

```
curl -X GET "https://<EPM IP address>/EPWebServices/rest/management/ttsServers/?name=TTS1" -H "accept: */*"
```

Request URL

```
https://<EPM IP address>/EPWebServices/rest/management/ttsServers/?name=TTS1
```

POST - Add a new TTS server

This web service can be found at: `https://<EPM IP address>/EPWebServices/rest-api/#/TTS%20Servers/createSpeechServer`

This web service provides a template which the user must edit with their desired TTS server configuration.

After the user has edited this template with valid values and submitted them using execute, a new TTS server will appear in EPM's Speech Server section.

Example:

```
{
  "name": "TTS2",
  "zone": "Default",
  "enable": true,
  "engineType": "Loquendo",
  "data": {
    "networkAddress": "1.2.3.4",
    "basePort": 554,
    "port": 1,
    "perPort": false,
    "MRCP": {
      "activityTimer": 15,
      "responseTimer": 4,
      "mrCPProtocol": "mrCPv1",
      "mrCPProtocolData": {
        "rtspUrl": "1.2.3.4/media/speechrecognizer"
      }
    }
  },
  "selectedLang": [
    "en-us English(USA)"
  ]
}
```

Curl

```
curl -X POST "https://<EPM IP address>/EPWebServices/rest/management/ttsServers/" -H "accept: */*" -H "Content-Type: application/json" -d '{"name":"TTS2","zone":"","enable":true,"engineType":"Loquendo","data":{"networkAddress":"1.2.3.4","basePort":554,"port":1,"perPort":false,"MRCP":{"activityTimer":15,"responseTimer":4,"mrCPProtocol":"mrCPv1","mrCPProtocolData":{"rtspUrl":\
```

```
"1.2.3.4/media/speechrecognizer\"}},\"selectedLang\": [\"en-us  
English(USA)\"]}}"
```

Request URL

`https://<EPM IP address>/EPWebServices/rest/management/ttsServers/`

PUT - Update TTS server configuration

This web service can be found at: `https://<EPM IP address>/EPWebServices/rest-api/#/TTS%20Servers/updateSpeechServerByName`

This web service takes one parameter and provides a template for the user to update with their desired TTS server configuration.

The user will input the configured TTS server name and use the template to update fields.

After the user has edited this template and submitted it using execute, the TTS server's configuration will be updated.

Example:

User Input:

Name: TTS1

```
{  
  "name": "TTS1",  
  "zone": "Default",  
  "enable": true,  
  "engineType": "Loquendo",  
  "data": {  
    "networkAddress": "127.0.0.1",  
    "basePort": 4900,  
    "port": 1,  
    "perPort": false,  
    "MRCP": {  
      "activityTimer": 15,  
      "responseTimer": 4,  
      "mrcpProtocol": "mrcpv1",  
      "mrcpProtocolData": {  
        "rtspUrl": "127.0.0.1/media/speechrecognizer"  
      }  
    },  
    "selectedLang": [  
      "en-us English(USA)"  
    ]  
  }  
}
```

Curl

```
curl -X PUT "https://<EPM IP  
address>/EPWebServices/rest/management/ttsServers/?name=TTS1" -H  
"accept: */*" -H "Content-Type: application/json" -d  
"{\"name\": \"TTS1\", \"zone\": \"0\", \"enable\": true, \"engineType\": \"Lo
```

```
quendo\", \"data\": {\"networkAddress\": \"127.0.0.1\", \"basePort\": 4900,
\"port\": 1, \"perPort\": false, \"MRCP\": {\"activityTimer\": 15, \"response
Timer\": 4, \"mrcpProtocol\": \"mrcpv1\", \"mrcpProtocolData\": {\"rtspUrl\":
\"127.0.0.1/media/speechrecognizer\"}}, \"selectedLang\": [\"en-us
English(USA)\"]}}
```

Request URL

```
https://<EPM IP
address>/EPWebServices/rest/management/ttsServers/?name=TTS1
```

DELETE TTS server

This web service can be found at: `https://<EPM IP address>/EPWebServices/rest-api/#/TTS%20Servers/deleteSpeechServerByName`

This web service takes one parameter, the configured TTS server's name. After entering the name and selecting execute, the selected TTS server will be deleted.

Example:

Curl

```
curl -X DELETE "https://<EPM IP
address>/EPWebServices/rest/management/ttsServers/?name=TTS1"-H
"accept: */*"
```

Request URL

```
https://<EPM IP
address>/EPWebServices/rest/management/ttsServers/?name=TTS1
Input Values
```

Input Values

The TTS Server DTO (Data Transfer Object) properties:

DTO	Property	Description	Format/Type	Default
TTS Server	Name	Name	String	TTS1
	Zone	Zone	String	Default
	Enable	Enable	Boolean	True
	EngineType	Engine Type	Enumeration (String: Nuance/Loquendo)	Nuance
	NetworkAddress	Network Address	String	127.0.0.1
	BasePort	Base Port	int32	4900
	Port	Total Number of Licensed TTS Resources	int32	1
	perPort	New Connection per Session	Boolean	False

	MRCP	See table below		
	selectedVoices	Selected Voices	(String) Array	en-US,Jennifer,F English(USA)

Other DTOs part of the TTS DTO:

DTO	Property	Description	Format/Type	Default
MRCP	activityTimer	Ping Interval	int32	15
	responseTimer	Response Timeout	int32	4
	mrcpProtocol	Protocol	Enumeration (DTO: mrcpv1/mrcpv2)	mrcpv1
MRCPv1	rtspURI	RTSP URL	String	<networkAddress>/media/speechrecognizer
MRCPv2	sessionXML	Enable Session XML	Boolean	False
	transport	Transport Protocol	Enumeration (DTO: TCP/TLS)	TCP
TCP	localPort	Listener Port	int32	5060
TLS	localPort	Listener Port	int32	5061
	SRTP		DTO	
SRTP	Priority	Priority	int32	0
	srtpEnable	Enable	Boolean	true
	srtpEncryptAlgorithm	Encryption Algorithm	Enumeration (String: AES_CM_128/None)	AES_CM_128
	srtpAuthAlgorithm	Authentication Algorithm	Enumeration (String: HMAC_SHA1_80 / HMAC_SHA1_32)	HMAC_SHA1_80
	rtcpEncryptEnable	RTCP Encryption Enabled	Boolean	False

	rtpAuthEnable	RTP Authentication Enabled	Boolean	True
--	---------------	----------------------------	---------	------

Add and Configure an MPP Server

Overview

The purpose of this feature is to add and update an MPP server using Experience Portal's web services.

The web service can be found at: `https://<EPM_IP_Address>/EPWebServices/rest-api/#/MPP%20Servers`

The web service implements the same functionality as the EPM Web UI page "Change MPP Server".

POST - Add MPP

To add an MPP the POST request is used. This web service can be found at: `https://<EPM_IP_Address>/EPWebServices/rest-api/#/MPP%20Servers/createMpp`

The only required parameters in a request body are "name" and "hostAddress". All other parameters can be left in their defaults.

Important:

After the MPP server has been added via the POST web service you must ensure that the MPP state is in *Stopped* state before attempting to change the state of the MPP e.g. to start the MPP. If you do not validate the state before attempting to start the MPP the MPP server will fail to start successfully. See *PUT – Change the state of an MPP server*.

Example:

```
{
  "name": "MPP122",
  "zone": "Default",
  "hostAddr": "192.168.123.123",
  "netAddrVoip": "",
  "netAddrMrpc": "",
  "netAddrAppSvr": "",
  "maxCalls": 5,
  "autoRestart": true,
  "useCustomTraces": false,
  "traceLevels": {
    "evtMgrTracelogLevel": "off",
    "avbObjLogLevel": "off",
    "avbIntLogLevel": "off",
    "avbInetLogLevel": "off",
    "ttsTracelogLevel": "off",
    "avbTelLogLevel": "off",
    "avbJsiLogLevel": "off",
    "cdrTracelogLevel": "off",
    "sipMessagingTracelogLevel": "off",
    "vxmlTracelogLevel": "off",
    "asrTracelogLevel": "off",
  }
}
```

```

    "avbClientLogLevel": "off",
    "webServiceTraceTracelogLevel": "off",
    "ccxmlTracelogLevel": "off",
    "sessionTracelogLevel": "off",
    "mediaMgrTracelogLevel": "off",
    "mgtTracelogLevel": "off",
    "avbPromptLogLevel": "off",
    "avbRecLogLevel": "off",
    "mediaVideoMgrTracelogLevel": "off",
    "teleTracelogLevel": "off",
    "mediaEndpointMgrTracelogLevel": "off",
    "mrcpTracelogLevel": "off"
  }
}

```

- You should set "useCustomTraces" to true to take "traceLevels" parameters into effect.
- Valid values for each of trace levels are "off", "fine", "finer", "finest".
- The parameter "zone" should have valid name of any zone configured on EPM, or "Default" if no zones are configured on EPM.

Curl command

```

curl -X POST
  "https://192.168.123.120/EPWebServices/rest/management/mppServers" -
  H "accept: */*" -H "Content-Type: application/json" -d
  "{\"name\":\"MPP122\",\"hostAddr\":\"192.168.123.122\",\"zone\":\"Default\",
  \"netAddrVoip\":\"\", \"netAddrMrcp\":\"\", \"netAddrAppSvr\":\"\",
  \"maxCalls\":10, \"autoRestart\":true, \"useCustomTraces\":false, \"traceLevels\":{
  \"evtMgrTracelogLevel\":\"off\", \"avbObjLogLevel\":\"off\",
  \"avbIntLogLevel\":\"off\", \"avbInetLogLevel\":\"off\", \"ttsTracelogLevel\":
  \"off\", \"avbTelLogLevel\":\"off\", \"avbJsiLogLevel\":\"off\", \"cdrTracelogLevel\":
  \"off\", \"sipMessagingTracelogLevel\":\"off\", \"vxm1TracelogLevel\":\"off\",
  \"asrTracelogLevel\":\"off\", \"avbClientLogLevel\":\"off\", \"webServiceTraceTracelogLevel\":
  \"off\", \"ccxmlTracelogLevel\":\"off\", \"sessionTracelogLevel\":\"off\", \"mediaMgrTracelogLevel\":
  \"off\", \"mgtTracelogLevel\":\"off\", \"avbPromptLogLevel\":\"off\",
  \"avbRecLogLevel\":\"off\", \"mediaVideoMgrTracelogLevel\":\"off\", \"teleTracelogLevel\":
  \"off\", \"mediaEndpointMgrTracelogLevel\":\"off\", \"mrcpTracelogLevel\":\"off\"}}\"

```

Request URL

https://<EPM address>/EPWebServices/rest/management/mppServers

Response code on success

201 (Created)

Response body

```

{
  "name": "MPP122",
  "hostAddr": "192.168.123.122",
  "netAddrVoip": "192.168.123.122",
  "netAddrMrcp": "192.168.123.122",
  "netAddrAppSvr": "192.168.123.122",
  "zone": "Default",

```

```

"maxCalls": 10,
"autoRestart": true,
"useCustomTraces": false,
"traceLevels": {
"mediaVideoMgrTracelogLevel": "off",
"vxmlTracelogLevel": "off",
"evtMgrTracelogLevel": "off",
"avbPromptLogLevel": "off",
"cdrTracelogLevel": "off",
"sessionTracelogLevel": "off",
"defaultLogLevel": "default",
"sipMessagingTracelogLevel": "off",
"mgtTracelogLevel": "off",
"avbClientLogLevel": "off",
"avbJsiLogLevel": "off",
"mediaEndpointMgrTracelogLevel": "off",
"webServiceTraceTracelogLevel": "off",
"avbIntLogLevel": "off",
"teleTracelogLevel": "off",
"avbObjLogLevel": "off",
"avbTelLogLevel": "off",
"ttsTracelogLevel": "off",
"asrTracelogLevel": "off",
"mediaMgrTracelogLevel": "off",
"avbRecLogLevel": "off",
"mrcpTracelogLevel": "off",
"ccxmlTracelogLevel": "off",
"avbInetLogLevel": "off"
}
}

```

Response headers

```

cache-control: private,no-cache,no-store,no-transform,must-
revalidate,proxy-revalidate,max-age=0,s-maxage=0
connection: Keep-Alive
content-length: 889
content-type: application/json
date: Fri,05 Mar 2021 14:12:05 GMT
keep-alive: timeout=5,max=100
server: Apache
strict-transport-security: max-age=31536000; includeSubdomains;
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block

```

Error cases

In all cases of unsuccess the code 500 (Internal Server Error) is returned. Possible errors:

1. The parameter "name" is empty. Error message: "Name cannot be null or empty".
2. The parameter "name" has invalid characters. Error message: "error-result=Exception occurred: Invalid MPP name: <name>"
3. MPP with such name already added. Error message: "error-result=Exception occurred:

Invalid MPP name: <name>"

4. The parameter "hostAddr" is empty or invalid. Error message: "error-result=Exception occurred: Invalid MPP address: <address>".
5. MPP does not exist or run. Error message: "error-result=Exception occurred: MPP host invalid".

PUT - Update MPP

To update an MPP the PUT request is used at `https://<EPM IP Address>/EPWebServices/rest-api/#/MPP%20Servers/updateMppByName`

Only required parameter is "name". You can change any parameter in a request body. If you change "hostAddr", an MPP with new address should exist and be run.

Example

```
{
  "zone": "Default",
  "hostAddr": "192.168.123.122",
  "netAddrVoip": "",
  "netAddrMrcp": "",
  "netAddrAppSvr": "",
  "maxCalls": 5,
  "autoRestart": true,
  "useCustomTraces": false,
  "traceLevels": {
    "evtMgrTracelogLevel": "off",
    "avbObjLogLevel": "off",
    "avbIntLogLevel": "off",
    "avbInetLogLevel": "off",
    "ttsTracelogLevel": "off",
    "avbTelLogLevel": "off",
    "avbJsiLogLevel": "off",
    "cdrTracelogLevel": "off",
    "sipMessagingTracelogLevel": "off",
    "vxmlTracelogLevel": "off",
    "asrTracelogLevel": "off",
    "avbClientLogLevel": "off",
    "webServiceTraceTracelogLevel": "off",
    "ccxmlTracelogLevel": "off",
    "sessionTracelogLevel": "off",
    "mediaMgrTracelogLevel": "off",
    "mgtTracelogLevel": "off",
    "avbPromptLogLevel": "off",
    "avbRecLogLevel": "off",
    "mediaVideoMgrTracelogLevel": "off",
    "teleTracelogLevel": "off",
    "mediaEndpointMgrTracelogLevel": "off",
    "mrpcTracelogLevel": "off"
  }
}
```

Request URL

```
https://<EPM
Address>/EPWebServices/rest/management/mppServers?name=MPP122
```

Curl command

```
curl -X PUT
"https://192.168.123.120/EPWebServices/rest/management/mppServers?name
=MPP122" -H "accept: */*" -H "Content-Type: application/json" -d
{"zone\":\"Default\", \"hostAddr\":\"192.168.123.122\", \"netAddrVoip\
\": \"\", \"netAddrMrcp\": \"\", \"netAddrAppSvr\": \"\", \"maxCalls\":5, \"au
toRestart\":true, \"useCustomTraces\":false, \"traceLevels\":{\"evtMgrTr
acelogLevel\":\"off\", \"avbObjLogLevel\":\"off\", \"avbIntLogLevel\":\"
off\", \"avbInetLogLevel\":\"off\", \"ttsTracelogLevel\":\"off\", \"avbTe
llogLevel\":\"off\", \"avbJsiLogLevel\":\"off\", \"cdrTracelogLevel\":\"
off\", \"sipMessagingTracelogLevel\":\"off\", \"vxmlTracelogLevel\":\"of
f\", \"asrTracelogLevel\":\"off\", \"avbClientLogLevel\":\"off\", \"webSe
rviceTraceTracelogLevel\":\"off\", \"ccxmlTracelogLevel\":\"off\", \"ses
sionTracelogLevel\":\"off\", \"mediaMgrTracelogLevel\":\"off\", \"mgtTra
celogLevel\":\"off\", \"avbPromptLogLevel\":\"off\", \"avbRecLogLevel\":
\"off\", \"mediaVideoMgrTracelogLevel\":\"off\", \"teleTracelogLevel\":
\"off\", \"mediaEndpointMgrTracelogLevel\":\"off\", \"mrpcTracelogLevel\
\": \"off\"}}"
```

Response code on success

200 (OK)

Response body

```
{
  "name": "MPP122",
  "hostAddr": "192.168.123.122",
  "netAddrVoip": "192.168.123.122",
  "netAddrMrcp": "192.168.123.122",
  "netAddrAppSvr": "192.168.123.122",
  "zone": "Default",
  "maxCalls": 5,
  "autoRestart": true,
  "useCustomTraces": false,
  "traceLevels": {
    "mediaVideoMgrTracelogLevel": "off",
    "vxmlTracelogLevel": "off",
    "evtMgrTracelogLevel": "off",
    "avbPromptLogLevel": "off",
    "cdrTracelogLevel": "off",
    "sessionTracelogLevel": "off",
    "defaultLogLevel": "default",
    "sipMessagingTracelogLevel": "off",
    "mgtTracelogLevel": "off",
    "avbClientLogLevel": "off",
    "avbJsiLogLevel": "off",
    "mediaEndpointMgrTracelogLevel": "off",
    "webServiceTraceTracelogLevel": "off",
    "avbIntLogLevel": "off",
    "teleTracelogLevel": "off",
    "avbObjLogLevel": "off",
    "avbTelLogLevel": "off",
```

```

    "ttsTracelogLevel": "off",
    "asrTracelogLevel": "off",
    "mediaMgrTracelogLevel": "off",
    "avbRecLogLevel": "off",
    "mrcpTracelogLevel": "off",
    "ccxmlTracelogLevel": "off",
    "avbInetLogLevel": "off"
  }
}

```

Response headers

```

cache-control: private,no-cache,no-store,no-transform,must-revalidate,proxy-revalidate,max-age=0,s-maxage=0
connection: Keep-Alive
content-length: 888
content-type: application/json
date: Fri,05 Mar 2021 14:40:15 GMT
expires: Thu,01 Jan 1970 00:00:00 GMT
keep-alive: timeout=5,max=100
server: Apache
strict-transport-security: max-age=31536000; includeSubdomains;
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block

```

Error cases

In all cases of unsuccess the code 500 (Internal Server Error) is returned. Possible errors:

- MPP doesn't exist. Error message: "error-result=Exception occurred: MPP with the name <name> not found".
- Other errors as on MPP adding.

Input Values

The MPP DTO (Data Transfer Object) properties:

DTO	Property	Description	Type	Default
MPP	name	MPP Name	String	N/A
	hostAddr	Host Address	String	N/A
	zone	MPP Zone	String	Default
	netAddrVoip	Network Address (VOIP)	String	N/A
	netAddrMrcp	Network Address (MRCP)	String	N/A

	netAddrAppSvr	Network Address (AppSVR)	String	N/A
	maxCalls		Integer (int32)	10
	autoRestart		Boolean	True
	useCustomTraces	Enable Trace Levels	Boolean	False
	traceLevels	Trace Levels	Object (MppTraceLevelsDTO)	
MppTraceLevelsDTO	evtMgrTracelogLevel	Event Manager	String	off
	avbObjLogLevel	Voice Browser Object	String	off
	avbIntLogLevel	Voice Browser Interpreter	String	off
	avbInetLogLevel	Voice Browser INET	String	off
	ttsTracelogLevel	TTS	String	off
	avbTelLogLevel	Voice Browser Telephony	String	off
	avbJsiLogLevel	Voice Browser Java Script Interface	String	off
	cdrTracelogLevel	Reporting	String	off
	sipMessagingTracelogLevel	SIP Messages Tracing	String	off
	vxmITracelogLevel	Voice Browser Platform	String	off
	asrTracelogLevel	ASR	String	off
	avbClientLogLevel	Voice Browser Client	String	off

	webServiceTraceTracelogLevel	Trace Logger	String	off
	ccxmlTracelogLevel	CCXML Browser	String	off
	sessionTracelogLevel	Session Manager	String	off
	mediaMgrTracelogLevel	Media Manager	String	off
	mgtTracelogLevel	MPP System Manager	String	off
	avbPromptLogLevel	Voice Browser Prompt	String	off
	avbRecLogLevel	Voice Browser Recognition	String	off
	mediaVideoMgrTracelogLevel	Media Video Manager	String	off
	teleTracelogLevel	Telephony	String	off
	mediaEndpointMgrTracelogLevel	Media Endpoint Manager	String	off
	mrcpTracelogLevel	MCRP	String	off

Deleting an MPP

Overview

The purpose of this feature is to delete an MPP server using Experience Portal's web services. The MPP web services can be found at: `https://<EPM_IP_Address>/EPWebServices/rest-api/#/MPP%20Servers`

DELETE – Delete an MPP server by name

To delete an MPP the DELETE request is used. The only required parameter of a request is "name". The web service can be found at:

`https://<EPM_IP_Address>/EPWebServices/rest-api/#/MPP%20Servers/deleteMppByName`

Example:

User input:

name: [mppNameToDelete]

curl

```
curl -X DELETE "https://<EPM IP Address>/EPWebServices/rest/management/mppServers?name=mppNameToDelete "
-H "accept: */*"
```

Request URL

`https:// <EPM IP Address>/EPWebServices/rest/management/mppServers?name=mppNameToDelete`

Response Code

204 (No Content)

Response headers

```
cache-control: private,no-cache,no-store,no-transform,must-revalidate,proxy-revalidate,max-age=0,s-maxage=0
connection: Keep-Alive
date: Fri,05 Mar 2021 13:59:35 GMT
expires: Thu,01 Jan 1970 00:00:00 GMT
keep-alive: timeout=5,max=99
server: Apache
strict-transport-security: max-age=31536000; includeSubdomains;
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Error codes

In all cases of unsuccess the code 500 (Internal Server Error) is returned. Possible errors:

1. The parameter "name" is empty. Error message: "Name cannot be null or empty".

Change the mode of an MPP server

Overview

The purpose of this feature is to change the mode of an MPP server using Experience Portal's web services. The MPP Manager change mode web service can be found at: `https://<EPM_IP_address>/EPWebServices/rest-api/#/MPP%20Manager/changeMPPmode`

The MPP Manager web service allows users to perform the following mode changes:

1. Online
2. Offline
3. Test

PUT – Set the mode of an MPP server

This web service can be found at: `https://<EPM_IP_Address>/EPWebServices/rest-api/#/MPP%20Manager/changeMPPmode`

It takes two parameters (a String and an enumeration value) from the user, the MPP's name and the desired mode.

The user will type the MPP's name into the input box and select the desired mode from a drop-down menu and select Execute.

If the PUT request is successful a 200-response code will be returned and the MPP will change into the desired mode.

The MPP Manager web service contains a validation class that will check if the MPP's name is valid and if the MPP can be changed into the requested mode.

If a user inputs an invalid MPP name, they will receive an Error: 400, code 1042 with the message "MPP name is invalid".

If a user selects a mode which an MPP cannot change into due to its current state/mode, the user will receive an Error: 400, code 1042 with the message "*Mode requested cannot be performed. MPP in Current state: <currentState> and Current Mode: <currentMode>. Possible state changes: <list of possible mode changes>*".

Example:

Curl

```
curl -X PUT "https://<EPM_IP_Address>/EPWebServices/rest/management/manageMPPServers/mode?mppName=mppTest01&mppMode=offline" -H "accept: */*"
```

Request URL

```
https:// <EPM_IP_Address>/EPWebServices/rest/management/manageMPPServers/mode?mppName=mppTest01&mppMode=online
```

Error cases

Unsupported mode changes:

The user will receive a 400 code (400 Bad Request response status code indicates that the server cannot or will not process the request due to something that is perceived to be a client error (e.g., malformed request syntax, invalid request message framing, or deceptive request routing).

The user will also receive a code: 1042 error (VPEXception.REQUEST_NOT_VALID) with a message stating the MPP's current mode as well as the supported mode changes, they can make.

MPP name not found:

The user will receive a 400 code from swagger (400 code is defined above). The user will also receive a 1042 error code (VPEXception.REQUEST_NOT_VALID) with a message to say the MPP name is invalid.

GET – Get the mode of an MPP server

This web service can be found at: `https://<EPM IP Address>/EPWebServices/rest-api/#/MPP%20Manager/getMPPmode`

This allows the user to retrieve the MPP's current mode before submitting a mode change.

If the GET request is successful, a 200 response code will appear along with the current mode of the MPP.

Similar to the PUT request, if a user enters an invalid MPP name they will receive an Error: 400, code 1042 with the message "MPP name is invalid".

Example

User input:

mppName: MPP

Curl

```
curl -X PUT "https://<EPM IP Address>/EPWebServices/rest/management/manageMPPServers/mode?mppName=MPP&mppMode=test" -H "accept: */*" --cacert /root/cacert.pem -u [username]:[password]
```

Note: The `/root/cacert.pem` is the location of `ca.crt` and `username` and `password` is specific to the `ca.crt` file.

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/manageMPPServers/mode?mppName=MPP&mppMode=test
```

Error cases

MPP name not found:

The user will receive a 400 code from swagger (400 code is defined above). The user will also receive a 1042 error code (VPEXception.REQUEST_NOT_VALID) with a message to say the MPP name is invalid.

Change the state of an MPP Server

Overview

The purpose of this feature is to change the state of an MPP server using Experience Portal's web services.

The MPP Manager web service allows users to perform the following state changes:

1. Start
2. Stop
3. Restart
4. Reboot
5. Halt
6. Cancel

PUT – Change the state of an MPP server

The MPP Manager change state web service can be found at: `https:// <EPM IP Address>/EPWebServices/rest-api/#/MPP%20Manager/changeMPPstate`. It takes two parameters (a String and an enumeration value) from the user, the MPP's name and the desired state. The MPP Manager web service contains a validation class that will check if the MPP's name is valid and if the MPP can be changed into the requested state.

Important:

If the MPP state change web service is being used to start an MPP server immediately after it has been added using the *POST - Add MPP* web service, the MPP server must be in a *Stopped* state before attempting to *Start* the MPP server. This can be achieved by utilizing the

GET – Get MPP server state web service and verifying that the returned state is *Stopped* before issuing the *Start* state change. It's important to check for the *Stopped* state and not *Stopping* state. For a newly added MPP server it is expected that it will transition to a stopped state within 30 seconds of being added.

Example:

User input:

mppName: [mppNameToChangeState]

mppState: [start, stop, restart, reboot, halt, cancel]

curl

```
curl -X PUT "https://<EPM IP Address>/EPWebServices/rest/management/manageMPPServers/state?mppName=mppNameToChangeState&mppState=start" -H "accept: */*"
```

Request URL

```
https://<EPM IP
Address>/EPWebServices/rest/management/manageMPPServers/state?mppName=
mppNameToChangeState&mppState=start
```

Response Code

200 (OK)

Error codes

1. If a user inputs an invalid MPP name, the following error will be received:
 - Error: 400, code 1042 with the message "MPP name is invalid".
2. If a request includes a state which an MPP cannot change into due to its current state/mode, the following error will be received:
 - Error: 400, code 1042 with the message "*Command requested cannot be performed. MPP in Current state: <currentState> and Current Mode: <currentMode>. Possible state changes: <list of possible state changes>.*"

GET – Get MPP server state

The MPP Manager get state web service can be found at: `https://<EPM IP Address>/EPWebServices/rest-api/#/MPP%20Manager/getMPPstate`. This web service allows the user to retrieve the MPP's current state before submitting a state change. It takes one parameter (a String) from the user, the MPP's name. The MPP Manager web service contains a validation class that will check if the MPP's name is valid.

Example

User input:

mppName: MPP

Curl:

```
curl -X GET "https://<EPM IP
Address>/EPWebServices/rest/management/manageMPPServers/currentState?mp
pName=mpp1" -H "accept: */*"
```

Request URL

```
https://<EPM IP
Address>/EPWebServices/rest/management/manageMPPServers/state?mppName=M
PP&mppState=stop
```

Response Code

If the GET request is successful, a 200 response code will appear along with the current state of the MPP.

200 (OK)

Error cases

If a user inputs an invalid MPP name, the following error will be received:

- Error: 400, code 1042 with the message "MPP name is invalid".

Configuring SIP Connections

Overview

The purpose of this feature is to create/read/update/delete SIP Connections using Experience Portal's web services.

The web service can be found at: `https://<EPM_IP_Address>/EPWebServices/rest-api/#/Sip%20Connections`

The web service corresponds to the 'System Configuration > VoIP Connections → SIP' WEB UI section.

POST – Create a SIP connection

To create a SIP Connection the POST request is used. This web service can be found at: `https://<EPM_IP_Address>/EPWebServices/rest-api/#/SIP%20Connections/createSipConnection`

The mandatory parameters are name, enable, proxyTransport, proxyServers, listenerPort, sipDomain, maximumCalls. The other parameters are optional.

Note 1. If "zone" parameter is not specified explicitly the system default zone with name "Default" will be used.

Note 2. Proxy servers and DNS SRV Domain:

- Per internal code structure "proxyServers" field is used for both proxy servers and DNS SRV domain. The both IP address and FQDN are valid values for proxyServers=>address.
- If DNS SRV Domain, should have data as following:
 - address (domain): IP or FQDN
 - port: 0 (port 0 is to indicate DNS SRV Domain is entered)
 - priority: 0
 - weight: 0

Example:

```
{
  "name": "testSipConn",
  "enable": false,
  "proxyTransport": "tcp",
  "proxyServers": [
    {
      "address": "192.168.111.222",
      "port": 5060,
      "priority": 0,
      "weight": 0
    }
  ],
  "listenerPort": 5060,
```

```
"sipDomain": "192.168.111.333",
"maximumCalls": 5
}
```

Response code on success

201 (Created)

Response body example:

```
{
  "name": "testSipConn",
  "enable": false,
  "proxyTransport": "tcp",
  "proxyServers": [
    {
      "address": "192.168.111.222",
      "port": 5060,
      "priority": 0,
      "weight": 0
    }
  ],
  "listenerPort": 5060,
  "sipDomain": "192.168.111.333",
  "maximumCalls": 5
}
```

Error cases

In case of one or more parameters have invalid values or a mandatory parameter is not set the related error message is displayed.

PUT - Update SIP connection

To update a SIP connections parameter's the PUT request is used. This web service can be found at: <https://<EPM IP Address>/EPWebServices/rest-api/#/SIP%20Connections/updateSipConnectionByName>

It takes 'name' parameter as name of SIP Connection to be updated. One or more parameters can be updated per a request.

Example

```
{
  "sipDomain": "sip.local.domain",
  "maximumCalls": 10
}
```

Response code on success

200 (OK)

Error cases

In case of one or more parameters have invalid values, the related error message is displayed.

DELETE – Delete a SIP connection

To delete a SIP Connection the DELETE request is used. This web service can be found at: <https://<EPM IP Address>/EPWebServices/rest-api/#/SIP%20Connections/deleteSipConnectionByName>

api/#/SIP%20Connections/deleteSipConnectionByName

It takes 'name' parameter as name of SIP Connection to be deleted.

Example:

User input:

name: [sipConnectionName]

curl

```
curl -X DELETE "https://<EPM IP Address>/EPWebServices/rest/management/sipConnections/sipConnectionName" -H "accept: */*"
```

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/sipConnections/sipConnectionName
```

Response code on success

204 (DELETED)

GET – Get SIP connection parameters by name

To retrieve the parameters of a SIP connection the GET request is used. This web service can be found at: <https://<EPM IP Address>/EPWebServices/rest-api/#/SIP%20Connections/getSipConnectionByName>

It takes 'name' parameter as name of SIP Connection to be retrieved.

Example

User input:

name: <sipConnectionName>

Curl

```
curl -X GET "https://10.134.142.178/EPWebServices/rest/management/sipConnections/sipConnectionName" -H "accept: */*"
```

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/sipConnections/sipConnectionName
```

Response code on success

200 (OK)

Response body example:

```
{
  "name": "sipconn11",
  "zone": "Default",
  "enable": true,
  "proxyTransport": "tcp",
  "proxyServers": [
    {
```

```

    "address": "192.168.122.22", "port": 5060, "priority": 0, "weight": 0
  }
],
"listenerPort": 5060,
"sipDomain": "accdev.lab",
"maximumCalls": 5,
"configureCallCapacity": "default",
"outboundAllowed": 2,
"inboundAllowed": 2,
"maximumRedirectionAttempts": 0,
"consultativeTransfer": "InviteWithReplaces",
"timerT1": 320,
"timerT2": 2000,
"timerBF": 4000,
"sipRejectResponseCode": "ASM",
"sipRejectResponseCodeValue": 503,
"passertedId": ""
}

```

GET - Get list of all SIP connections

To retrieve a list of all SIP connections the GET request is used. This web service can be found at: <https://<EPM IP Address>/EPWebServices/rest-api/#/SIP%20Connections/getAllSipConnections>

Example

curl

```
curl -X GET "https://10.134.142.178/EPWebServices/rest/management/sipConnections" -H "accept: */*"
```

Request URL

<https://<EPM IP Address>/EPWebServices/rest/management/sipConnections>

Response code on success

200 (OK)

Response body example:

```

[
  { "name": "cc-sm" },
  { "name": "sipconn11" },
  { "name": "yk_SipConn2" },
  { "name": "yk_SipConn3" }
]

```

GET - Get SIP connections count

To retrieve a count of all SIP connections the GET request is used. This web service can be found at: <https://<EPM IP Address>/EPWebServices/rest-api/#/SIP%20Connections/getCountOfSipConnections>

Example:

curl

```
curl -X GET "https://<EPM IP
Address>/EPWebServices/rest/management/sipConnections/count" -H "accept:
*/*"
```

Request URL

```
https://<EPM IP
Address>/EPWebServices/rest/management/sipConnections/count
```

Response code on success

200 (OK)

Response body example:

```
{ "count": 4 }
```

Input Values

The SIP DTO properties:

DTO	Property	Description	Type/Format	Default
SIP	Name	Name	String	N/A
	Zone	SIP Zone	String	N/A
	enable		Boolean	True
	proxyTransport		String	N/A
	proxyServers		Array (ProxyServerDTO)	N/A
	listenerPort		Integer (int32)	0
	sipDomain		String	N/A
	maximumCalls		Integer (int32)	0
	passertedId		String	N/A
	configureCallCapacity		String	N/A
	outboundAllowed		Integer (int32)	0
	inboundAllowed		Integer (int32)	0
	maximumRedirectionAttempts		Integer (int32)	0
	timerT1		Integer (int32)	0
	timerT2		Integer (int32)	0
	timerBF		Integer (int32)	0
	consultativeTransfer		Enumeration (String: InviteWithReplaces/Refer)	InviteWithReplaces
	sipRejectResponseCode		String	N/A
	sipRejectResponseCodeValue		Integer (int32)	0
	srtplibSelect		array (SRTPdto)	(See table below)

Other DTOs used by the SIP Web Service:

DTO	Property	Description	Type/Format	Default
SRTP	Priority	Priority	int32	0
	srtpEnable	Enable	Boolean	true
	srtpEncryptAlgorithm	Encryption Algorithm	Enumeration (String: AES_CM_128/None)	AES_CM_128
	srtpAuthAlgorithm	Authentication Algorithm	Enumeration (String: HMAC_SHA1_80 / HMAC_SHA1_32)	HMAC_SHA1_80
	rtcpEncryptEnable	RTCP Encryption Enabled	Boolean	False
	rtpAuthEnable	RTP Authentication Enabled	Boolean	True
ProxyServerDTO	address	Proxy Server Address	String	N/A
	Port	Port Number	Integer (int32)	0
	priority		Integer (int32)	0
	weight		Integer (int32)	0

Identity Certificates

Overview

The purpose of this feature is to configure Identity certificates for EPM and MPP servers using Experience Portal's web services. The web services can be found

at: `https://<EPM_IP_Address>/EPWebServices/rest-api/#/Certificates`

The web service implements the same functionality as the EPM Web UI page "Certificates".

Pre-conditions

Before uploading externally signed identity certificates the following steps should be performed:

- Execute "Disable Signing" on the "EP Signing Certificate" tab of the "Certificate" page.
- Upload a Certification Authority certificate on the "Trusted Certificate" tab of the "Certificate" page, as a type "Platform".

POST – Uploading an Identify Certificate

To upload an identity certificate the POST request is used. This web service can be found at:

`https://<EPM IP Address>/EPWebServices/rest-api/#/Certificates/uploadIdentityCertificate`

The certificate should be in PKCS#12 format. A server name, which the certificate is being installed for, and a certificate file password are required.

Example

User input

serverName: [EPMServerName]

password: [passwordOfPKCS12file]

file: [PKCS12fileToUpload]

curl

```
curl -X POST "https://<EPM IP Address>/EPWebServices/rest/management/certificates/identity/upload?serverName=EPM&password=<password>" -H "accept: */*" -H "Content-Type: multipart/form-data" -F "file=@PKCS12fileToUpload.pl2;type=application/x-pkcs12"
```

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/certificates/identity/upload?serverName=EPM&password=<password>
```

Response code on success

201 (Created)

Response headers

```
cache-control: private,no-cache,no-store,no-transform,must-revalidate,proxy-revalidate,max-age=0,s-maxage=0
connection: close
content-length: 40
content-type: application/json
date: Wed,14 Apr 2021 14:02:59 GMT
server: Apache
strict-transport-security: max-age=31536000; includeSubdomains;
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Error codes

In cases of unsuccess the code 400 (Bad Request) or 500 (Internal Server Error) is returned. Possible errors:

1. The parameter "serverName" is empty. Error message: "Identity Certificate Server Name cannot be empty".
2. The parameter "serverName" has invalid characters. Error message: "Identity Certificate Server Name is invalid <name>".
3. The certificate file password is invalid. Error message: "Invalid security certificate file (<file name>) - keystore password was incorrect".

Input Values

The Identity certificate DTO (Data Transfer Object) properties:

Property	Description	Type/Format	Default
Name	Name of certificate	String	N/A
File	A .p12 certificate file to upload	String (Binary)	N/A
Password	A password for,p12 certificate	String (Password)	N/A

GET - Get Identity Certificate

To retrieve an identity certificate the GET request is used. The only required parameter is Server Name. This web service can be found at: <https://<EPM IP Address>/EPWebServices/rest-api/#/Certificates/Certificates/getIdentityCertificate>

Example

User input

```
serverName: [serverNameToGetIdentityCertFrom]
```

curl

```
curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/certificates/identity?serverName=MPP121" -H "accept: */*"
```

Request URL

```
https://<EPM address>/EPWebServices/rest/management/certificates/identity?serverName=serverNameToGetIdentityCertFrom
```

Response code on success

200 (OK)

Response headers

```
cache-control: private,no-cache,no-store,no-transform,must-revalidate,proxy-revalidate,max-age=0,s-maxage=0
connection: Keep-Alive
content-length: 1281
content-type: application/json
date: Fri,30 Apr 2021 15:16:42 GMT
expires: Thu,01 Jan 1970 00:00:00 GMT
keep-alive: timeout=5,max=100
server: Apache
strict-transport-security: max-age=31536000; includeSubdomains;
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Error codes

In cases of unsuccess the code 500 (Internal Server Error) is returned. Possible errors:

1. The parameter "serverName" is invalid. Error message: "errorCode: 1000, errorMessage: Server does not exist <name>".

Input Values

The Identity certificate DTO (Data Transfer Object) properties:

Property	Description	Type/Format	Default
Name	Name of certificate	String	N/A

Configuring an External DB on the Primary EPM

Overview

The purpose of this feature is to create/read/update/delete an External Report Database using Experience Portal's web services.

The External Report Database web service test Swagger page can be found

at: `https://<EPM_IP_Address>/EPWebServices/rest-api/#/External%20Report%20Database`

The web service corresponds to the 'EPM Servers → Data Storage Settings → Report Database' WEB UI section.

POST - Set up External Report Database

To set up an External Report Database the POST request is used. EPM reporting is switched to the configured External Database.

The mandatory parameters are **url**, **jdbcDriver**, **username** and **password**. Parameters **dbEncryptionType** and **dbChecksumType** are applicable for Oracle database only.

Example:

```
{
  "url": "jdbc:postgresql://192.168.161.102:5432/postgres",
  "jdbcDriver": "org.postgresql.Driver",
  "username": "postgres",
  "password": "Password1",
  "dbEncryptionType": "",
  "dbChecksumType": ""
}
```

Curl

```
curl -X PUT "https://<EPM IP
address>/EPWebServices/rest/management/externalReportDB" -H "accept:
/*/*" -H "Content-Type: application/json" -d
"{\"url\": \"http://external.database.com\", \"jdbcDriver\": \"Oracle\", \"
username\": \"my_username\", \"password\": \"my_password\", \"dbEncryptio
nType\": \"None\", \"dbChecksumType\": \"None\"}"
```

Request URL

```
https://<EPM IP
address>/EPWebServices/rest/management/externalReportDB
```

Response code on success

```
201 (Created)
```

Response body example:

```
{
  "url": "jdbc:postgresql://192.168.161.102:5432/postgres",
  "jdbcDriver": "org.postgresql.Driver",
```

```

"username": "postgres",
"password": "Password1",
"dbEncryptionType": "",
"dbChecksumType": ""
}

```

Error cases

- One or more mandatory parameters are empty. Error message: "<Parameter name> cannot be null or empty".
- EPM Reporting Database is already set to External. Error message: "The operation is not allowed since Report DB is already set to External. Please use UPDATE operation instead."
- External Database is not available or cannot be created with the provided parameters. The related error message is displayed.
- External Database parameters cannot be retrieved or saved to the local database. The related error message is displayed.

PUT - Update External Reporting Database

To update an External Report Database the PUT request is used.

NOTE! In case of empty request body passed → **External Database availability** will be checked using the existing parameters.

Example:

```

{
  "url": "http://external.database.com",
  "jdbcDriver": "Oracle",
  "username": "my_username",
  "password": "my_password",
  "dbEncryptionType": "None",
  "dbChecksumType": "None"
}

```

Curl

```

curl -X PUT "https://<EPM IP
address>/EPWebServices/rest/management/externalReportDB" -H "accept:
*/*" -H "Content-Type: application/json" -d
"{\"url\": \"http://external.database.com\", \"jdbcDriver\": \"Oracle\", \"
username\": \"my_username\", \"password\": \"my_password\", \"dbEncryptio
nType\": \"None\", \"dbChecksumType\": \"None\"}"

```

Request URL

```

https://<EPM IP
address>/EPWebServices/rest/management/externalReportDB

```

Response code on success

200 (OK)

Error cases

The same as for set up request except there is no mandatory parameters in update request.

GET - Get External Report Database

To retrieve the parameters of an existing External Report Database the GET request is used.

Example:

Curl:

```
curl -X GET "https://<EPM IP  
Address>/EPWebServices/rest/management/externalReportDB" -H "accept:  
*/*"
```

Request URL

```
https://<EPM IP  
Address>/EPWebServices/rest/management/externalReportDB
```

Response code on success

200 (OK)

Response body example:

```
{  
  "url": "jdbc:postgresql://192.168.1.2:5432/postgres",  
  "jdbcDriver": "org.postgresql.Driver",  
  "username": "postgres",  
  "password": "Password1",  
  "dbEncryptionType": "",  
  "dbChecksumType": ""  
}
```

Error cases

EPM Reporting Database is not set to External. Error message: "The operation is not allowed since Report DB is not set to External."

DELETE - Delete External Report Database

To switch EPM reporting from External Database to Local the DELETE request is used.

Example:

Curl

```
curl -X DELETE "https://<EPM IP  
address>/EPWebServices/rest/management/externalReportDB" -H "accept:  
*/*"
```

Request URL

```
https://<EPM IP  
address>/EPWebServices/rest/management/externalReportDB
```

Response code on success

204

Error cases

EPM Reporting Database is not set to External. Error message: "The operation is not allowed

since Report DB is not set to External."

Input Values

The ExternalReportDB DTO (Data Transfer Object) properties:

Property	Description	Type	Default
URL	URL	String	http://external.database.com
jdbcDriver	JDBC Driver	String	Oracle
username	User name	String	my_username
password	User password	String	my_password
dbEncryptionType	Oracle Encryption Type	String	None
dbChecksumType	Oracle Checksum Type	String	None

Configuring System Backup on Primary EPM

Overview

The System Backup feature in the Experience Portal Manager (EPM) can be used to regularly back up the data in a local Experience Portal database and the associated properties files.

The purpose of this feature is to perform an on-demand backup and configure the backup server, backup schedule, and the files and folders for the backup operation using Experience Portal's Web Services. This web service corresponds to the 'System Management > System Backup' WEB UI section. The System Backup web service test Swagger page can be found at:

https://<epm_ip_address>/EPWebServices/rest-api/#/System%20Backup

GET - Get System Backup Configuration

To retrieve the currently configured backup server location and authentication information the GET request is used at

https://<EPM_IP_Address>/EPWebServices/rest/management/systemBackup/config. The package name and date/time of the last successful backup is also returned.

Example:

Curl

```
-curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/config" -H "accept: */*"
```

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/config
```

Response code on success:

200 (OK)

Response body example:

```
{
  "servertype": "rhLinux",
  "serveraddress": "localhost",
  "backupfolder": "/opt/Avaya/backup",
  "numberbackups": "3",
  "username": "",
  "password": "*****"
}
```

Parameter notes:

Password will always be returned as *****.

PUT - Update System Backup Configuration

To update the backup server location and authentication information the PUT request is used at
`https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/config`

Example

```
{
  "servertype": "",
  "serveraddress": "",
  "backupfolder": "",
  "numberbackups": "3",
  "username": "",
  "password": ""
}
```

Curl

```
curl -X PUT "https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/config" -H "accept: */*" -H "Content-Type: application/json" -d '{"servertype":"rhLinux","serveraddress":"localhost","backupfolder":"/opt/Avaya/backup","numberbackups":"3","username":"","password":""}'
```

Request URL

`https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/config`

Response code on success

200 (OK)

Error cases:

If ServerType is not empty, it must be "pcWindows" or "rhLinux":

```
{"code": 500, "message": "servertype is invalid"}
```

ServerType is required if ServerAddress or BackupFolder are not empty:

```
{"code": 500, "message": "servertype cannot be null or empty"}
```

NumberBackups, must be a numeric value between 1 and 5. Default 0 = ignore:

```
{"code": 500, "message": "numberbackups is invalid"}
```

```
{"code": 500, "message": "numberbackups is not between 1 and 5"}
```

GET - Get System Backup Customization Folders and Files

To retrieve the custom folders and files the GET request is used at
`https://<epm_ip_address>/EPWebServices/rest/management/systemBackup/customization`

Example

curl

```
curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/customization" -H "accept: */*"
```

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/customization
```

Response code on success

200 (OK)

Response body example

```
{
  "customfolders": "/folder1,/folder2",
  "customfiles": "1*.xml,2*.xml"
}
```

Parameter notes:

1. "customfolders": is a string of folders for custom components (comma separated) e.g. "/folder1,/folder2"
2. "customfiles": Name and the path of the file that you want to backup. One file specifier must exist for each folder specified in customfolders. You can specify the file name with file extension or with asterisk (*). For example, you can specify file names like *.xml or common.*.

PUT - Update System Backup Customization Folders and Files

To update the custom folders and files the PUT request is used at `https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/customization`

Example

```
{
  "customfolders": "/folder1,/folder2",
  "customfiles": "1*.xml,*"
}
```

Response code on success

200 (OK)

Response body example

```
{
  "customfolders": "/folder1,/folder2",
  "customfiles": "1*.xml,*"
}
```

Parameter notes

1. "customfolders": is a string of folders for custom components (comma separated). eg "/folder1,/folder2"

2. "customfiles": Name and the path of the file that you want to backup. One file specifier must exist for each folder specified in customfolders. You can specify the file name with file extension or with asterisk (*). For example, you can specify file names like *.xml or common.*.

Error cases

When the number of comma separated items in customfiles does not match the number of comma separated items in customfolders:

```
{"code": 500,"message": "error-result=Exception occurred: Cannot update System Backup settings: The number of custom file specifications does not equal the number of folders"}
```

DELETE - Delete System Backup Customization Folders and Files

To remove all of the custom folders and files from the backup configuration the DELETE request is used at `https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/customization`

Example

curl

```
curl -X DELETE "https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/customization" -H "accept: */*"
```

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/customization
```

Response code on success

204 (Deleted)

GET - Get System Backup Schedule

To retrieve the currently configured backup schedule the GET request is used at `https:// <EPM IP Address>/EPWebServices/rest/management/systemBackup/schedule`

Example

curl

```
curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/schedule" -H "accept: */*"
```

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/customization  
https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/schedule
```

Response code on success

200

Response body example

```
{
  "oncedate": 0,
  "dailytime": "12:34",
  "weeklyday": 0,
  "weeklytime": "",
  "monthlyday": 0,
  "monthlytime": ""
}
```

Parameter notes

- oncedate, represents the requested one-time scheduled backup start time in UNIX Epoch time (i.e. milliseconds since January 1, 1970 00:00:00 UTC).
- dailytime, weeklytime and monthlytime, are in 24 hr time with format HH:mm.
- weeklyday, 1 = Sunday thru 7 = Saturday.

PUT - Update System Backup Schedule

To update the backup schedule the PUT request is used at `https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/schedule`

Example:

```
{
  "oncedate": 0,
  "dailytime": "",
  "weeklyday": 0,
  "weeklytime": "",
  "monthlyday": 15,
  "monthlytime": "03:00"
}
```

curl

```
curl -X PUT "https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/schedule" -H "accept: */*" -H "Content-Type: application/json" -d '{"oncedate":0,"dailytime":"","weeklyday":0,"weeklytime":"","monthlyday":0,"monthlytime":"03:00"}'
```

Request URL

```
https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/schedule
```

Response code on success:

201 (Created)

Response body example:

```
{
  "oncedate": 0,
  "dailytime": "",
  "weeklyday": 0,
  "weeklytime": "",
  "monthlyday": 15,
  "monthlytime": "03:00"
}
```

Parameter notes:

1. oncedate, represents the requested one-time scheduled backup start time in UNIX Epoch time (i.e. milliseconds since January 1, 1970 00:00:00 UTC).
2. dailytime, weeklytime and monthlytime, are in 24 hr time with format HH:mm.
3. weeklyday, 1 = Sunday thru 7 = Saturday.

Error cases:

oncedate, if not empty, must be numeric and represent the requested backup start time in UNIX Epoch time (i.e. milliseconds since January 1, 1970 00:00:00 UTC). Default 0 = ignore.

```
{"code": 500, "message": "oncedate is invalid"}
```

dailytime, weeklytime and monthlytime, must be a valid 24 hr time with format HH:mm. eg. 03:30 for 3:30 AM or 18:30 for 6:30 PM. Default "" = ignore.

```
{"code": 500, "message": "dailytime is invalid"}
```

```
{"code": 500, "message": " weeklytime is invalid"}
```

```
{"code": 500, "message": "monthlytime is invalid"}
```

weeklyday, 1 = Sunday ... 7 = Saturday. Default 0 = ignore.

```
{"code": 500, "message": "weeklyday is invalid"}
```

monthlyday, between 1 and 31. (The last day of the month is used if monthlyday exceeds the number of days in the month. Default 0 = ignore.

```
{"code": 500, "message": "monthlyday is invalid"}
```

DELETE - Delete System Backup Schedule

To remove the current backup schedule the DELETE request is used at `https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/schedule`. The AEP UI will show a schedule frequency of "None".

Example:

curl

```
curl -X DELETE "https://<EPM IP
Address>/EPWebServices/rest/management/systemBackup/schedule" -H
"accept: */*"
```

Request URL

```
https://<EPM IP
Address>/EPWebServices/rest/management/systemBackup/schedule
```

Response code on success

204 (Deleted)

GET - Get System Backup History

To retrieve the System Backup History a GET request is used at `https://<EPM IP Address>/EPWebServices/rest/management/systemBackup/history`. The package name and date/time are returned; one entry for each of configured "Number of Backups to Retain".

Example:

curl

```
curl -X GET "https://<EPM IP
Address>/EPWebServices/rest/management/systemBackup/history" -H
"accept: */*"
```

Request URL

```
https://<EPM IP
Address>/EPWebServices/rest/management/systemBackup/history
```

Response code on success

200

Response body example

```
[
  {
    "packagename": "pkgEP8.1.0.0.0163_1618526384151",
    "datetime": "2021-04-15T22:39:44.151Z"
  },
  {
    "packagename": "pkgEP8.1.0.0.0163_1618520250853",
    "datetime": "2021-04-15T20:57:30.853Z"
  }
]
```

Parameter notes:

- packagename, follows the `pkg<Version>_<Timestamp>` format, where the time stamp is in UNIX Epoch time (i.e. milliseconds since January 1, 1970 00:00:00 UTC).
- datetime, is the date and time when the package was backed up, in ISO 8601 format.

POST - Start an On-Demand System Backup

To start an on-demand System Backup the POST request is used at `https://<epm_ip_address>/EPWebServices/rest/management/systemBackup`

Example:

curl

```
curl -X POST "https://<EPM IP Address>/EPWebServices/rest/management/systemBackup" -H "accept: */*" -d ""
```

Request URL

`https://<EPM IP Address>/EPWebServices/rest/management/systemBackup`

Response code on success

201 (Created)

Error cases

`{"code": 500,"message": "error-result=Exception occurred: Previous backup operation is still in progress. Please try later."}`

Input Values

The System backup DTOs (Data Transfer Object) and their properties:

DTO	Property	Description	Type/Format	Default
SystemBackupConfigDTO	serverType	Server Type	Enumeration (String: rhLinux/pcWindows)	rhLinux
	serverAddress	Server Address	String	localhost
	backupfolder	Backup Folder	String	/opt/Avaya/backup
	Numberbackups	Number of Backups to Retain	String	3
	username	Username for pcWindows ServerType	String	N/A
	password	Password for pcWindows ServerType	String	N/A

SystemBackupCustomizationDTO	customfolders	Folder for custom components	String	N/A
	customfiles	Files to include for customfolders	String	N/A
SystemBackupScheduleDTO	oncedate	One time schedule date (Epoch timestamp)	int64	0
	dailytime	Daily Time (hh:mm)	String	N/A
	weeklyday	Weekly Day	int32	0
	weeklytime	Weekly Time (hh:mm)	String	N/A
	monthlyday	Monthly Day	int32	0
	monthlytime	Monthly Time (hh:mm)	String	N/A

Trusted Certificates

Overview

The purpose of this feature is to provide the ability to import, upload, get and delete Trusted Certificates for the EPM sever using Experience Portal's web services. The web service implements the same functionality as the EPM Web UI page Home > Security > Certificates, the "Trusted Certificates" tab. The Trusted Certificate web service test Swagger page can be found at: <https://<EPM address>/EPWebServices/rest-api/#/Certificates>

POST - Import Trusted Certificate

To import a Trusted Certificate the POST request is used. The certificate should be in PEM format.

Required parameters:

1. Certificate name
2. Certificate Type
3. Import URL

The type of the certificate should be either: Application, CRL File, LDAP Server, SIP Connection, Speech server, Platform, System Manager, User.

Request URL

<https://<EPM address>/EPWebServices/rest/management/certificates/trusted/import?name=test-import-cert&type=Platform&url=https%3A%2F%2F192.168.123.122%2Fcert.pem>

Curl command

```
curl -X POST "https://<EPM address>/EPWebServices/rest/management/certificates/trusted/import?name=test-import-cert&type=Platform&url=https%3A%2F%2F192.168.123.122%2Fcert.pem" -H "accept: */*" -d ""
```

Response code on success

201 (Created)

Response headers

```
cache-control: private,no-cache,no-store,no-transform,must-revalidate,proxy-revalidate,max-age=0,s-maxage=0
connection: close
content-length: 2833
content-type: application/json
date: Thu,06 May 2021 15:10:59 GMT
server: Apache
strict-transport-security: max-age=31536000; includeSubdomains;
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Error cases

In case of all unsuccess HTTP 500 (Internal Server Error) is returned.

- Certificate with such name already exists. Error message: "errorCode": 400, "errorMessage": "A security certificate with this name already exists."
- Import URL is invalid. Error message: "errorCode": 400, "errorMessage": "Unable to retrieve security certificate from the specified URL."

POST - Upload Trusted Certificate - PEM

To upload a PEM format Trusted Certificate the POST request is used.

Required parameters:

- Certificate name
- Certificate Type
- File name

Request URL

```
https://<EPM IP  
Address>/EPWebServices/rest/management/certificates/trusted/upload/pem?name=smgr-  
cacert&type=Platform
```

Curl command

```
curl -X POST "https://<EPM  
address>/EPWebServices/rest/management/certificates/trusted/upload/pem?name=smgr-  
cacert&type=Platform" -H "accept: */*" -H "Content-Type: multipart/form-data" -F  
file=@SystemManagerCA1.pem
```

Response code on success

201 (Created)

Response headers

```
cache-control: private,no-cache,no-store,no-transform,must-revalidate,proxy-revalidate,max-  
age=0,s-maxage=0  
connection: close  
content-length: 1324  
content-type: application/json  
date: Thu,06 May 2021 15:38:28 GMT  
server: Apache  
strict-transport-security: max-age=31536000; includeSubdomains;  
x-content-type-options: nosniff  
x-frame-options: SAMEORIGIN  
x-xss-protection: 1; mode=block
```

Error cases

In case of all unsuccess HTTP 500 (Internal Server Error) is returned.

Certificate name has invalid characters. Error message: "code": 1042, "message": "Trusted

Certificate Name is invalid <name>"

POST - Upload Trusted Certificate - PKCS#12

To upload a PKCS#12 format Trusted Certificate the POST request is used.

- Certificate name
- Certificate Type
- File name
- File password

Request URL

```
https://<EPM IP address>/EPWebServices/rest/management/certificates/trusted/upload/pkcs12?name=test-upload-pkcs12&type=Platform&password=<password>
```

Curl command

```
curl -X POST "https://192.168.123.120/EPWebServices/rest/management/certificates/trusted/upload/pkcs12?name=test-upload-pkcs12&type=Platform&password=*****" -H "accept: */*" -H "Content-Type: multipart/form-data" -F file=@cacert.p12;type=application/x-pkcs12
```

Response code on success

201 (Created)

Response headers

```
cache-control: private,no-cache,no-store,no-transform,must-revalidate,proxy-revalidate,max-age=0,s-maxage=0
connection: close
content-length: 1412
content-type: application/json
date: Thu,06 May 2021 15:50:35 GMT
server: Apache
strict-transport-security: max-age=31536000; includeSubdomains;
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Error cases

In case of all unsuccess HTTP 500 (Internal Server Error) is returned.

1. Password incorrect. Error message: "errorCode": 400, "errorMessage": "Invalid security certificate file (<file name>) - keystore password was incorrect."

GET - Get Trusted Certificate - Count

To get a count of Trusted Certificates the GET request is used. No parameter is required.

Request URL

https://<EPM IP Address>/EPWebServices/rest/management/certificates/trusted/count

Curl command

```
curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/certificates/trusted/count" -H "accept: */*"
```

Response code on success

200 (OK)

Error cases

No errors.

GET - Get Trusted Certificate(s) - Names

To get all the Trusted Certificates the GET request is used. To get a specific trusted certificate the name of a certificate can be provided.

Request URL

Certificate name unspecified

https://<EPM IP Address>/EPWebServices/rest/management/certificates/trusted

Certificate name specified

https://<EPM IP Address>/EPWebServices/rest/management/certificates/trusted?name=smgr-cacert

Curl command

Certificate name unspecified

```
curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/certificates/trusted" -H "accept: */*"
```

Certificate name specified

```
curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/certificates/trusted?name=smgr-cacert" -H "accept: */*"
```

Response code on success

200 (OK)

Error cases

Wrong certificate name. Error message: "errorCode": 400, "errorMessage": "trusted certificate was not found in the database by name <name>".

DELETE - Delete Trusted Certificate

To delete a Trusted Certificates the DELETE request is used. The certificate name is the required parameter.

Request URL

`https://<EPM address>/EPWebServices/rest/management/certificates/trusted?name=<certificate name>`

Curl command

```
curl -X DELETE "https://<EPM IP Address>/EPWebServices/rest/management/certificates/trusted?name=test-import-cert" -H "accept: */*"
```

Response code on success

204 (No Content)

Response headers

```
cache-control: private,no-cache,no-store,no-transform,must-revalidate,proxy-revalidate,max-age=0,s-maxage=0
connection: close
date: Thu,06 May 2021 15:28:02 GMT
expires: Thu,01 Jan 1970 00:00:00 GMT
server: Apache
strict-transport-security:
max-age=31536000; includeSubdomains;
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Error cases

Known Issues

DELETE operation always return success even if no certificate exists.

Input Values

The Trusted Certificate properties:

Property	Description	Type/Format)	Default
Name		String	N/A
Type		Enumeration (String: CRL File/ LDAP Server/SIP Connection/Speech Server/Platform/System Manager/User)	Application
URL	URL location of certificate	String	N/A
File	A PEM certificate file to upload	String (Binary)	N/A
File	A PKCS#12 certificate file to upload	String (Binary)	N/A
Password	A password for PKCS#12 certificate	String (Password)	N/A

Configure SNMP Agent settings

Overview

The purpose of this feature is to configure SNMP Agent settings using Experience Portal's web services. The web service implements the same functionality as the EPM Web UI page "SNMP Agent Settings". The SNMP Agent webservice can be found at: <https://<EPM address>/EPWebServices/rest-api/#/SNMP%20Agent>

PUT - Configure SNMP Agent

To configure the SNMP Agent the PUT request is used.

There are no required parameters in a request body, all parameters can be left in their defaults.

Request body

```
{
  "snmpVersion1Enabled": false,
  "snmpVersion1SecurityName": "",
  "snmpVersion2cEnabled": false,
  "snmpVersion2cSecurityName": "",
  "snmpVersion3Enabled": false,
  "snmpVersion3SecurityName": "",
  "snmpVersion3AuthenticationProtocol": "None",
  "snmpVersion3AuthenticationPassword": "",
  "snmpVersion3PrivacyProtocol": "None",
  "snmpVersion3PrivacyPassword": "",
  "allowIpAddressSelection": false,
  "authorizedIpAddress1": "",
  "authorizedIpAddress2": "",
  "authorizedIpAddress3": "",
  "authorizedIpAddress4": "",
  "authorizedIpAddress5": "",
  "transportProtocol": "UDP",
  "portNumberSelection": false,
  "specificPortNumber": 161
}
```

- a. You can enable SNMP version 1, 2c and 3 in any combination.
- b. If you enable a version, you should set a security name for it.
- c. If you set Privacy Protocol for the version 3, you should set Authentication protocol as well.
- d. Valid values for Authentication protocol are: None, MD5, SHA
- e. Valid values for Privacy Protocol are: None, DES, AES128, AES192, AES256
- f. If you enable allowIpAddressSelection, you should set at least one authorized IP address.
- g. The parameter transportProtocol can be set to UDP or TCP, but only UDP is currently supported.
- h. Default port for SNMP Agent is 161. You can change it by parameters portNumberSelection and specificPortNumber for your needs. Port range 1 - 65535. Restart SNMP Agent if Port number is changed.

Request URL

<https://<EPM address>/EPWebServices/rest/management/snmpAgent>

Curl command

```
curl -X PUT "https://<EPM IP Address>/EPWebServices/rest/management/snmpAgent" -
H "accept: */*" -H "Content-Type: application/json" -d
"{\"snmpVersion1Enabled\":false,\"snmpVersion1SecurityName\":\"\",\"snmpVersion2cEnabled\":fal
se,\"snmpVersion2cSecurityName\":\"\",\"snmpVersion3Enabled\":false,\"snmpVersion3SecurityNa
me\":\"\",\"snmpVersion3AuthenticationProtocol\":\"None\",\"snmpVersion3AuthenticationPasswo
rd\":\"\",\"snmpVersion3PrivacyProtocol\":\"None\",\"snmpVersion3PrivacyPassword\":\"\",\"allowIpAd
dressSelection\":false,\"authorizedIpAddress1\":\"\",\"authorizedIpAddress2\":\"\",\"authorizedIpAd
dress3\":\"\",\"authorizedIpAddress4\":\"\",\"authorizedIpAddress5\":\"\",\"transportProtocol\":\"UDP\",
\"portNumberSelection\":false,\"specificPortNumber\":161}"
```

Response code on success

204 (No Content)

Response headers

```
cache-control: private,no-cache,no-store,no-transform,must-revalidate,proxy-revalidate,max-
age=0,s-maxage=0
connection: Keep-Alive
date: Tue,09 Mar 2021 11:56:29 GMT
expires: Thu,01 Jan 1970 00:00:00 GMT
keep-alive: timeout=5,max=100
server: Apache
strict-transport-security: max-age=31536000; includeSubdomains;
x-content-type-options: nosniff
x-frame-options: SAMEORIGIN
x-xss-protection: 1; mode=block
```

Error cases

In all cases of unsuccess the code 500 (Internal Server Error) is returned. Possible errors:

- The specific version is enabled, but its security name is empty. Error message: "empty name".
- The parameter "name" has invalid characters. Error message: "Invalid name <name>"
- Authentication or Privacy protocol is set, but its password is empty. Error message: "empty password".
- Authentication protocol is set incorrectly. Error message: "error-result=Exception occurred: invalidSnmpVersion3AuthenticationProtocol, correct values: [MD5, None, SHA]"
- Privacy protocol is set incorrectly. Error message: "error-result=Exception occurred: invalidSnmpVersion3PrivacyProtocol, correct values: [AES128, AES192, AES256, DES, None]"
- Privacy protocol is set, but Authentication protocol is not set. Error message: "if Privacy Protocol is defined, Authentication Protocol cannot be undefined".
- One of authorized IP addresses is set incorrectly. Error message: "Invalid address <address>"
- Transport protocol is set incorrectly. Error message: "invalid protocol <protocol>"
- Port is set incorrectly. Error message: "invalid port <port>"

Known issues

1. If "specificPortNumber" is out of range, error message is "Invalid protocol" instead of "Invalid port".
2. In case of success the request returns only response code. Should the request return new SNMP Agent configuration?

Input Values

The SNMP Agent DTO (Data Transfer Object) properties:

Property	Description	Type	Default
snmpVersion1Enabled	Enable SNMP Version 1	Boolean	False
snmpVersion1SecurityName	SNMP Version 1 Security Name	String	N/A
snmpVersion2cEnabled	Enable SNMP Version 2c	Boolean	False
snmpVersion2cSecurityName	SNMP Version 2c Security Name	String	N/A
snmpVersion2cEnabled	Enable SNMP Version 3	Boolean	False
snmpVersion3SecurityName	SNMP Version 3 Security Name	String	N/A
snmpVersion3AuthenticationProtocol	SNMP Version 3 Authentication Protocol	String	None
snmpVersion3AuthenticationPassword	SNMP Version 3 Authentication Password	String	N/A
snmpVersion3PrivacyProtocol	SNMP Version 3 Privacy Protocol	String	None
snmpVersion3PrivacyPassword	SNMP Version 3 Privacy Password	String	N/A
allowIpAddressSelection	Allow Only the Following Authorized IP Address	Boolean	False
authorizedIpAddress1	IP Address/Hostname 1	String	N/A
authorizedIpAddress2	IP Address/Hostname 2	String	N/A
authorizedIpAddress3	IP Address/Hostname 3	String	N/A
authorizedIpAddress4	IP Address/Hostname 4	String	N/A
authorizedIpAddress5	IP Address/Hostname 5	String	N/A
transportProtocol	Transport Protocol	String	UDP
portNumberSelection	Select Specific Port Number	Boolean	False
specificPortNumber		Integer (int32)	161

Update the Operation Grace Period

Overview

The purpose of this feature is to change the Operational Grace Period value using Experience Portal's web services. The Operational Grace Period is an EPM Setting available in the Miscellaneous section of Home > System Configuration > EPM Servers > EPM Settings. The Operational Grace Period webservice can be found at:

`https://<hostaddress>/EPWebServices/rest-api/#/EPM%20Settings/setEPMGracePeriod`

It takes one parameter from the user (an integer). This must be a whole number between 0 and 999.

The user will not be allowed to enter a number outside of this range.

If the PUT request is successful a 200 response code will be returned and the EPM's Operational Grace Period setting will be changed to the desired number.

Example

User input:

```
operationalGracePeriod: 1
```

curl:

```
curl -X PUT "https://<EPM IP Address>/EPWebServices/rest/management/epmSettings/operationalGracePeriod?operationalGracePeriod=1" -H "accept: */*" --cacert /root/cacert.pem -u [username]:[password] /root/cacert.pem is the location of cacert.pem (see Web Services Architecture & Detailed Design > CURL HTTPS - Secure Web Service with Certificate Authority (CA) Certificate for instructions on acquiring this certificate).
```

Request URL

```
https://<EPM IP
```

```
Address>/EPWebServices/rest/management/epmSettings/operationalGracePeriod?operationalGracePeriod=1
```

Input Values

The Operational Grace Period is an Integer.

Possible Errors:

If a user tries to input a number outside of 0 to 999 via curl, the following error will be returned:

```
"code":1042,"message":"Operational grace period must be a number between 0 and 999  
Code 1042 = VPEXCEPTION.REQUEST_NOT_VALID
```

Configuring EPM Conversation Store

Overview

The purpose of this feature is to read/update EPM Conversation Store using Experience Portal's

REST web services. The web service corresponds to the 'EPM Servers → Data Storage Settings → Conversations' WEB UI section. The Conversation Store web service can be found at: https://<EPM_IP_Address>/EPWebServices/rest-api/#/Conversation%20Store

Users can manage the Conversation Store in the following ways;

1. Put – request to update the Conversation Store
2. Get – request to retrieve the Conversation Store configuration

PUT - Update Conversation Store

To update Conversation Store parameters the PUT request is used.

Request body:

serverType parameter is mandatory and can be set to one of the following values: 'LocalEPM', 'PrimaryEPM' and 'ContextStore'.

hostAddress and **clientTimeout** parameters should only be set when serverType = 'ContextStore'.

Curl

```
curl -X PUT
"https://192.168.123.137/EPWebServices/rest/management/conversationStore
?serverType=PrimaryEPM" -H "accept: */*"
```

Request URL

```
https://<EPM IP address>/EPWebServices/rest/management/
conversationStore
```

Response code on success

204 (No Content)

Error cases

- serverType = 'ContextStore' and hostAddress is invalid → Response Code 500, "error-result=Exception occurred: conversationContextStoreHost is invalid".
- serverType = 'ContextStore' and clientTimeout is invalid → Response Code 500, "error-result=Exception occurred: conversationContextStoreTimeout is invalid: Specified attribute is not between the expected values of 15 and 300".

GET – Retrieve Conversation Store Parameters

To retrieve existing Conversation Store parameters the GET request is used.

Request body example:

-

Curl

```
curl -X GET
"https://192.168.123.137/EPWebServices/rest/management/conversationStore
" -H "accept: */*"
```

Request URL

https://<EPM IP address>/EPWebServices/rest/management/
conversationStore

Response code on success

200 (OK)

Response body example:

```
{
  "value": "PrimaryEPM"
}
```

Error cases

-

Input Values

Conversation Store properties:

Property	Type	Default
ServerType	String	N/A
Host Address	String	N/A
Conversation Server Timeout	String	60

Configuring LDAP Settings

Overview

The purpose of this feature is to allow cloud engineers to read/update EPM Keycloak Settings using Experience Portal's REST Webservices. The Webservice corresponds to the 'User Management → Login Options → Keycloak Settings' WEB UI section.

The LDAP Settings webservice test Swagger page can be found at:

`https://<EPM IP Address>/EPWebServices/rest-api/#/User%20Settings`

Users can manage the LDAP Settings in the following ways:

1. GET – request to retrieve the LDAP Settings configuration
2. PUT – request to update the Keycloak Settings

GET

To retrieve existing LDAP settings the GET request is used.

Request body:

-

Curl

```
curl -X GET "https://<EPM IP Address>/EPWebServices/rest/management/userSettings/ldap" -H "accept: */*"
```

Request URL

`https://<EPM IP address>/EPWebServices/rest/management/userSettings/ldap`

Response code on success

200 (OK)

Response body example:

```
{
  "ldapEnable": true,
  "ldapURL": "ldap://192.168.123.65:389",
  "ldapUsername": "cn=Administrator,cn=Users,dc=winldap65,dc=ru",
  "ldapPassword": "123!Avaya",
  "ldapMutualCertAuthentication": false,
  "ldapReferrals": "ignore",
  "ldapUserSearchChoice": "UserDNPattern",
  "ldapUserDNPattern": "cn={0},ou=ou1,dc=winldap65,dc=ru",
  "ldapUserSearchFilter": "",
  "ldapUserSearchBaseDN": "",
  "ldapUserSearchSubtree": false,
  "ldapPasswordChoice": "Bind",
  "ldapPasswordAttribute": "",
  "ldapRoleUserEntryAttribute": "memberOf",
```

```

"ldapRoleGroupSearchFilter": "(member={0})",
"ldapRoleGroupEntryAttribute": "cn",
"ldapRoleGroupSearchBaseDN": "ou=oul,dc=winldap65,dc=ru",
"ldapRoleSearchSubtree": false,
"ldapRoleGroupMaps": [
  {
    "groupMapName": "test1",
    "organization": "",
    "assignedRoles": "administration,operations"
  },
  {
    "groupMapName": "test2",
    "organization": "Org1",
    "assignedRoles":
"deptadmin,deptauditor,deptprivacymanager,deptreport,deptuserman,deptwebserv
ices"  }
]
}

```

Error cases

-

PUT

To update LDAP Settings the PUT request is used.

Request body example:

```

{
  "ldapEnable": true,
  "ldapURL": "ldap://192.168.123.65:389",
  "ldapUsername": "cn=Administrator,cn=Users,dc=winldap65,dc=ru",
  "ldapPassword": "123!Avaya",
  "ldapMutualCertAuthentication": false,
  "ldapReferrals": "ignore",
  "ldapUserSearchChoice": "UserDNPattern",
  "ldapUserDNPattern": "cn={0},ou=oul,dc=winldap65,dc=ru",
  "ldapUserSearchFilter": "",
  "ldapUserSearchBaseDN": "",
  "ldapUserSearchSubtree": false,
  "ldapPasswordChoice": "Bind",
  "ldapPasswordAttribute": "",
  "ldapRoleUserEntryAttribute": "memberOf",
  "ldapRoleGroupSearchFilter": "(member={0})",
  "ldapRoleGroupEntryAttribute": "cn",
  "ldapRoleGroupSearchBaseDN": "ou=oul,dc=winldap65,dc=ru",
  "ldapRoleSearchSubtree": false,
  "ldapRoleGroupMaps": [
    {

```

```

    "groupMapName": "test1",
    "organization": "",
    "assignedRoles": "administration,operations"
  },
  {
    "groupMapName": "test2",
    "organization": "Org1",
    "assignedRoles":
"deptadmin,deptauditor,deptprivacymanager,deptreport,deptuserman,deptwebservices"
  }
]
}

```

Curl

```

curl -X PUT
"https://192.168.123.137/EPWebServices/rest/management/userSettings/ldap" -H
"accept: */*" -H "Content-Type: application/json" -d
"{\"ldapEnable\":true,\"ldapURL\":\"ldap://192.168.123.65:389\",\"ldapUsername\":
\"cn=Administrator,cn=Users,dc=winldap65,dc=ru\",\"ldapPassword\":\"123!Avaya\"
,\"ldapMutualCertAuthentication\":false,\"ldapReferrals\":\"ignore\",\"ldapUserS
earchChoice\":\"UserDNPattern\",\"ldapUserDNPattern\":\"cn={0},ou=ou1,dc=winldap
65,dc=ru\",\"ldapUserSearchFilter\":\"\",\"ldapUserSearchBaseDN\":\"\",\"ldapUse
rSearchSubtree\":false,\"ldapPasswordChoice\":\"Bind\",\"ldapPasswordAttribute\"
:\"\",\"ldapRoleUserEntryAttribute\":\"memberOf\",\"ldapRoleGroupSearchFilter\":
\"(memberOf={0})\",\"ldapRoleGroupEntryAttribute\":\"cn\",\"ldapRoleGroupSearchBas
eDN\":\"ou=ou1,dc=winldap65,dc=ru\",\"ldapRoleSearchSubtree\":false,\"ldapRoleGr
oupMaps\":[{\"groupMapName\":\"test1\",\"organization\":\"\",\"assignedRoles\":\
\"administration,operations\"},{\"groupMapName\":\"test2\",\"organization\":\
\"Org1\",\"assignedRoles\":\"deptadmin,deptauditor,deptprivacymanager,deptreport,dept
userman,deptwebservices\"}]}"

```

Request URL

https://<EPM IP address>/EPWebServices/rest/management/userSettings/ldap

Response code on success

200 (OK)

Response body example:

```

{
  "ldapEnable": true,
  "ldapURL": "ldap://192.168.123.65:389",
  "ldapUsername": "cn=Administrator,cn=Users,dc=winldap65,dc=ru",
  "ldapPassword": "123!Avaya",
  "ldapMutualCertAuthentication": false,
  "ldapReferrals": "ignore",
  "ldapUserSearchChoice": "UserDNPattern",
  "ldapUserDNPattern": "cn={0},ou=ou1,dc=winldap65,dc=ru",
  "ldapUserSearchFilter": "",
  "ldapUserSearchBaseDN": "",
  "ldapUserSearchSubtree": false,
  "ldapPasswordChoice": "Bind",

```



```

"ldapPasswordAttribute": "",
"ldapRoleUserEntryAttribute": "memberOf",
"ldapRoleGroupSearchFilter": "(member={0})",
"ldapRoleGroupEntryAttribute": "cn",
"ldapRoleGroupSearchBaseDN": "ou=oul,dc=winldap65,dc=ru",
"ldapRoleSearchSubtree": false,
"ldapRoleGroupMaps": [
  {
    "groupMapName": "test1",
    "organization": "",
    "assignedRoles": "administration,operations"
  },
  {
    "groupMapName": "test2",
    "organization": "Org1",
    "assignedRoles":
"deptadmin,deptauditor,deptprivacymanager,deptreport,deptuserman,deptwebserv
ices"
  }
]
}

```

Error cases

1. In case of one of the mandatory parameters is set to empty string - the related error message is displayed.
2. In case of a parameter has invalid value - the related error message is displayed.

Input Values

IdapEnable → expected values - true/false

WebUI Connection Settings

IdapURL → expected values - LDAP URL string value

IdapUsername → expected values - LDAP user name string value

IdapPassword → expected values - LDAP password string value

IdapMutualCertAuthentication → expected values - true/false; relates to WebUI field Mutual Certificate Authentication

IdapReferrals → expected values - ignore/follow; relates to WebUI field Referrals

WebUI User Entry Settings

IdapUserSearchChoice → expected values - UserDNPattern/SearchFilter; relates to WebUI radio button for User Entry Settings

IdapUserDNPattern → expected value - User DN Pattern string value; relates to WebUI User DN Pattern edit box

IdapUserSearchFilter → expected value - User Search Filter string value; relates to WebUI Search Filter edit box

IdapUserSearchBaseDN → expected value - User Search Base DN string value; relates to WebUI Base DN edit box

IdapUserSearchSubtree → expected values - true/false; relates to WebUI Search Subtree check box

WebUI Password Verification Settings

IdapPasswordChoice → expected values - Bind/Attribute; relates to WebUI radio button for Password Verification Settings

IdapPasswordAttribute → expected value - Attribute string value; relates to WebUI Attribute edit box

WebUI Role Assignment Settings

IdapRoleUserEntryAttribute → expected value - User Entry Attribute string value; relates to WebUI User Entry Attribute edit box

IdapRoleGroupSearchFilter → expected value - Group Search Filter string value; relates to WebUI Group Search Filter edit box

IdapRoleGroupEntryAttribute → expected value - Group Entry Attribute string value; relates to WebUI Group Entry Attribute edit box

IdapRoleGroupSearchBaseDN → expected value - Group Search Base DN string value; relates to WebUI Group Search Base DN edit box

IdapRoleSearchSubtree → expected values - true/false; relates to WebUI Search Subtree check box

IdapRoleGroupMaps - roles map

Important info

1. There is a possibility to update one or more parameters without passing the whole list. In this case you should pass only those parameters that need to be updated, the other parameters are stayed without change (except roles map - you should pass the whole expected roles list in case of update).
In the example below: LDAP password settings are updated, the roles map updated as follows: map "test1" is not changed, "test2" is deleted (see previous example), "test3" is added. The other fields are stayed as it were.

Example:

```
{
  "ldapEnable": true, "ldapPasswordChoice": "Attribute",
  "ldapPasswordAttribute": "cn", "ldapRoleGroupMaps": [
    {
      "groupMapName": "test1",
      "organization": "",
      "assignedRoles": "administration, operations"
    },
    {
      "groupMapName": "test3",
      "organization": "Org2",

```

```
    "assignedRoles":  
    "deptadmin,deptauditor,deptprivacymanager,deptreport,deptuserman,deptwe  
bervices"  
    }  
  ]  
}
```

2. In case of some field(s) are missed - it means they should not be updated.
In case of some fields(s) are set as empty string ("") - it means they should be set as empty.

Configuring SSO Keycloak Settings

Overview

The purpose of this feature is to allow cloud engineers to read/update EPM Keycloak Settings using Experience Portal's REST Webservices. The Webservice corresponds to the 'User Management → Login Options → Keycloak Settings' WEB UI section.

The Keycloak Settings webservice test Swagger page can be found at: <https://<EPM IP Address>/EPWebServices/rest-api/#/User%20Settings>

Users can manage the Keycloak Settings in the following ways:

3. Get – request to retrieve the Keycloak Settings configuration
4. Put – request to update the Keycloak Settings

GET

To retrieve existing Keycloak settings the GET request is used.

Request body:

-

Curl

```
curl -X GET
"https://192.168.123.137/EPWebServices/rest/management/userSettings/keycloak" -H "accept: */*"
```

Request URL

```
https://<EPM IP address>/EPWebServices/rest/management/userSettings/keycloak
```

Response code on success

200 (OK)

Response body example:

```
{
  "keycloakSsoEnable": true,
  "keycloakAuthServerURL": "10.11.12.13",
  "defaultKeycloakRealm": "default_realm_test",
  "keycloakClientId": "client_id_test",
  "keycloakClientSecret": "client_secret_test",
  "keycloakRealmOrganizationMap": [
    {
      "keycloakRealm": "realm1",
      "organization": ""
    },
    {
      "keycloakRealm": "realm2",
      "organization": "Org2"
    }
  ]
}
```

```

],
"keycloakRolesMap": [
  {
    "groupMapName": "roleName22",
    "keycloakRealm": "realm1",
    "organization": "",
    "assignedRoles": "administration"
  },
  {
    "groupMapName": "roleName23",
    "keycloakRealm": "realm2",
    "organization": "Org2",
    "assignedRoles": "deptadmin,deptreport"
  }
]
}

```

Error cases

-

PUT

To update Keycloak Settings the PUT request is used.

Request body example:

```

{
  "keycloakSsoEnable": true,
  "keycloakAuthServerURL": "10.11.12.13",
  "defaultKeycloakRealm": "default_realm_test",
  "keycloakClientId": "client_id_test",
  "keycloakClientSecret": "client_secret_test",
  "keycloakRealmOrganizationMap": [
    {
      "keycloakRealm": "realm1",
      "organization": ""
    },
    {
      "keycloakRealm": "realm2",
      "organization": "Org2"
    }
  ],
  "keycloakRolesMap": [
    {
      "groupMapName": "roleName22",
      "keycloakRealm": "realm1",
      "organization": "",
      "assignedRoles": "administration"
    },
    {

```

```

        "groupMapName": "roleName23",
        "keycloakRealm": "realm2",
        "organization": "Org2",
        "assignedRoles": "deptadmin,deptreport"
    }
]
}

```

Curl

```

curl -X PUT
"https://192.168.123.137/EPWebServices/rest/management/userSettings/keycloak" -H "accept: */*" -H "Content-Type: application/json" -d
"{\"keycloakSsoEnable\":true, \"keycloakAuthServerURL\": \"10.11.12.13\", \"defaultKeycloakRealm\": \"default_realm_test\", \"keycloakClientId\": \"client_id_test\", \"keycloakClientSecret\": \"client_secret_test\", \"keycloakRealmOrganizationMap\": [{\"keycloakRealm\": \"realm1\", \"organization\": \"\"}, {\"keycloakRealm\": \"realm2\", \"organization\": \"Org2\"}], \"keycloakRolesMap\": [{\"groupMapName\": \"roleName22\", \"keycloakRealm\": \"realm1\", \"organization\": \"\", \"assignedRoles\": \"administration\"}, {\"groupMapName\": \"roleName23\", \"keycloakRealm\": \"realm2\", \"organization\": \"Org2\", \"assignedRoles\": \"deptadmin,deptreport\"}]}"

```

Request URL

https://<EPM IP address>/EPWebServices/rest/management/userSettings/keycloak

Response code on success

200 (OK)

Response body example:

```

{
  "keycloakSsoEnable": true,
  "keycloakAuthServerURL": "10.11.12.13",
  "defaultKeycloakRealm": "default_realm_test",
  "keycloakClientId": "client_id_test",
  "keycloakClientSecret": "client_secret_test",
  "keycloakRealmOrganizationMap": [
    {
      "keycloakRealm": "realm1",
      "organization": ""
    },
    {
      "keycloakRealm": "realm2",
      "organization": "Org2"
    }
  ],
  "keycloakRolesMap": [
    {
      "groupMapName": "roleName22",
      "keycloakRealm": "realm1",
      "organization": "",
      "assignedRoles": "administration"
    },
    {
      "groupMapName": "roleName23",

```

```

    "keycloakRealm": "realm2",
    "organization": "Org2",
    "assignedRoles": "deptadmin,deptreport"
  }
]
}

```

Error cases

1. In case of one of the mandatory parameters is set to empty string - the related error message is displayed.
2. In case of a parameter has invalid value - the related error message is displayed.

Input Values

keycloakSsoEnable → expected values - true/false

keycloakAuthServerURL → expected value - Keycloak server authentication URL (string value)

keycloakClientId → expected value - Keycloak client Id (string value)

keycloakClientSecret → expected value - Keycloak client secret (string value)

defaultKeycloakRealm → expected value - default Keycloak realm (string value)

keycloakRealmOrganizationMap → realms-orgs map

keycloakRolesMap -> roles map

Important info

1. There is a possibility to update one or more parameters without passing the whole list. In this case you should pass only those parameters that need to be updated, the other parameters are stayed without change (except roles map and realms-orgs map - you should pass the whole expected list in case of update).
In the example below: Keycloak setting "defaultKeycloakRealm" is updated (in compare with the above example), the roles map updated as follows: map "roleName22" is not changed, "roleName23" is deleted, "roleName24" is added.
The other fields are stayed as it were.

Example:

```

{
  "defaultKeycloakRealm": "default_realm_test1",
  "keycloakRolesMap": [
    {
      "groupMapName": "roleName22",
      "keycloakRealm": "realm1",
      "organization": "",
      "assignedRoles": "administration"
    },
    {
      "groupMapName": "roleName24",
      "keycloakRealm": "realm2",
      "organization": "Org2",
      "assignedRoles": "deptadmin"
    }
  ]
}

```

```
]
}
```

2. In case of some field(s) are missed - it means they should not be updated.

In case of some fields(s) are set as empty string ("") - it means they should be set as empty.

Configuring MPP Settings – Resource & Trace Levels

Overview

The purpose of this feature is to allow cloud engineers to change MPP Settings using Experience Portal's Web services.

The MPP Settings web service allows users to perform the following MPP Settings changes:

- Resource Alert Threshold
- Trace Logger
- Transcription Retention Period
- Record Handling.
- Category Trace Levels.

The MPP Settings web service can be found at:

<https://<hostaddress>/EPWebServices/rest-api/#/MPP%20Settings/>

GET – Resource Alert Threshold

To retrieve the current values of Resource Alert Threshold parameters a GET request is used.

Location: <https://<hostname>/EPWebServices/rest-api/#/MPP%20Settings/resourceAlertThreshold>

Input Value:

The user will select a resource from a drop-down menu. The choices for resources:

- CPU
- Disk
- Memory

Response

The response body returned will contain the values of the 'High Water' and 'Low Water' thresholds for the selected resource. These values are Integers between 0 and 100.

Properties of Resource Alert Threshold Values

Property	Type
Resource	Enumeration
High Water	Integer
Low Water	Integer

PUT – Resource Alert Threshold

To update the values of Resource Alert Threshold parameters a PUT request is used.
Location: <https://<hostname>/EPWebServices/rest-api/#/MPP%20Settings/resourceAlertThreshold>

Input Value:

The user will select a resource from a drop-down menu. The choices for resources:

- CPU
- Disk
- Memory

The user will input an Integer between 0 and 100 for 'highWater' and 'lowWater'.

Response

A '200' response code will be returned after successful execution.

Error cases

If a user attempts to enter a value for highWater which is lower than lowWater values, a 400 error will be returned along with the message 'High water must be a higher value than low water'.

Properties of Resource Alert Threshold Values

Property	Type
Resource	Enumeration
High Water	Integer
Low Water	Integer

GET – Trace Logger

To retrieve the current values of the Trace Logger parameters a GET request is used.
Location: <https://<hostname>/EPWebServices/rest-api/#/MPP%20Settings/traceLogger>
There are no input values for this web service.

Response

The response body returned will contain the values of the 'logFileMaxSize' and 'numOfLogsToRetain'.

Properties of Trace Logger

Property	Type
logFileMaxSize	Integer

numOfLogsToRetain	Integer
-------------------	---------

PUT – Trace Logger

To update the values of Trace Logger a PUT web service is used.

Location: <https://<hostaddress>/EPWebServices/rest-api/#/MPP%20Settings/traceLogger>

Input Value:

This web service takes two parameters:

- logFileMaxSize: type - Integer (must be between 1 and 100)
- numOfLogsToRetain: type – Integer (must be between 1 and 5)

Response

A '200' response code will be returned after successful execution.

Error cases

If a user attempts to enter a value for logFileMaxSize which is outside of 1 and 100, a '400' error code will be returned along with the message 'Max file size must be between 1 and 100'.

If a user attempts to enter a value for numOfLogsToRetain which is outside of 1 and 5, a '400' error code will be returned along with the message 'number of logs retained must be between 1 and 5'.

GET – Transcription Retention Period

To retrieve the value of the Transcription Retention Period a GET web service is used.

Location:

<https://<hostaddress>/EPWebServices/restapi/#/MPP%20Settings/transcriptionRetentionPeriod>

This web service has no parameters.

Response

A '200' response code will be returned along with the value of the Transcription Retention Period.

PUT – Transcription Retention Period

To update the value of the Transcription Retention Period a PUT web service is used.

Location:

<https://<hostaddress>/EPWebServices/restapi/#/MPP%20Settings/transcriptionRetentionPeriod>

Input Value:

This web service takes one parameter:

- transcriptionRetentionPeriod: type - Integer (must be between 0 and 999)

Response

A '200' response code will be returned after successful execution.

Error cases

If a user inputs a value outside of 0 and 999 the request will be unsuccessful. They will receive a 400 error with the message 'Retention period must be between 0 and 999.'

GET – Record Handling

To retrieve the current values of Record Handling parameters a GET request is used.
Location: <https://<hostname>/EPWebServices/rest-api/#/MPP%20Settings/recordHandling>

Input Value:

The user will select a record from a drop-down menu. The choices for record:

- Session Data
- Call Data
- VXML/CCXML Log Tags

Response

The response body returned will contain the values of 'Enable' and 'Retention Period' for the selected record.

Properties of Record Handling

Property	Type
Record	Enumeration
Enable	Boolean
Retention Period	Integer

PUT – Record Handling

To update the values of Resource Alert Threshold parameters a PUT request is used.
Location: <https://<hostname>/EPWebServices/rest-api/#/MPP%20Settings/recordHandling>

Input Value:

The user will select a resource from a drop-down menu. The choices for record:

- Session Data
- Call Data
- VXML/CCXML Log Tags

The user will input an Integer between 0 and 999 for 'Retention Period'.
The user will select 'True' or 'False' for enable.

Response

A '200' response code will be returned after successful execution.

Error cases

If a user attempts to enter a value for retentionPeriod which is outside of 0 and 999, a 400 error will be returned along with the message 'Retention Period must be a between 0 and 999'.

Properties of Record Handling

Property	Type
Record	Enumeration
Enable	Boolean
Retention Period	Integer

GET – Category Trace Level

To retrieve the current values of Category Trace Levels a GET request is used.

Location: <https://<hostname>/EPWebServices/rest-api/#/MPP%20Settings/categoryTraceLevels>

There are no parameters.

Response

A DTO containing all of the Category Trace Levels will be returned.

Properties of Category Trace Levels

MppTraceLevelsDTO	Name	Description	Type	Default
	avbObjLogLevel	Voice Browser Object	String	off
	avbIntLogLevel	Voice Browser Interpreter	String	off
	avbInetLogLevel	Voice Browser INET	String	off
	ttsTracelogLevel	TTS	String	off
	avbTelLogLevel	Voice Browser Telephony	String	off
	avbJsiLogLevel	Voice Browser Java Script Interface	String	off

	cdrTracelogLevel	Reporting	String	off
	sipMessagingTracelogLevel	SIP Messages Tracing	String	off
	vxmlTracelogLevel	Voice Browser Platform	String	off
	asrTracelogLevel	ASR	String	off
	avbClientLogLevel	Voice Browser Client	String	off
	webServiceTraceTracelogLevel	Trace Logger	String	off
	ccxmlTracelogLevel	CCXML Browser	String	off
	sessionTracelogLevel	Session Manager	String	off
	mediaMgrTracelogLevel	Media Manager	String	off
	mgtTracelogLevel	MPP System Manager	String	off
	avbPromptLogLevel	Voice Browser Prompt	String	off
	avbRecLogLevel	Voice Browser Recognition	String	off
	mediaVideoMgrTracelogLevel	Media Video Manager	String	off
	teleTracelogLevel	Telephony	String	off
	mediaEndpointMgrTracelogLevel	Media Endpoint Manager	String	off
	mrpcTracelogLevel	MCRP	String	off

PUT – Category Trace Level

To update the current values of Category Trace Levels a GET request is used.

Location: <https://<hostname>/EPWebServices/rest-api/#/MPP%20Settings/categoryTraceLevels>

There are no parameters.

Input Value:

The user will fill out a DTO containing all of the Category Trace Levels (see properties below).

Response

A '200' response along with a DTO containing all of the Category Trace Levels will be returned.

Properties of Category Trace Levels

MppTraceLevelsDTO	Name	Description	Type	Default
	avbObjLogLevel	Voice Browser Object	String	off
	avbIntLogLevel	Voice Browser Interpreter	String	off
	avbInetLogLevel	Voice Browser INET	String	off
	ttsTracelogLevel	TTS	String	off
	avbTelLogLevel	Voice Browser Telephony	String	off
	avbJsiLogLevel	Voice Browser Java Script Interface	String	off
	cdrTracelogLevel	Reporting	String	off
	sipMessagingTracelogLevel	SIP Messages Tracing	String	off
	vxmlTracelogLevel	Voice Browser Platform	String	off
	asrTracelogLevel	ASR	String	off
	avbClientLogLevel	Voice Browser Client	String	off
	webServiceTraceTracelogLevel	Trace Logger	String	off
	ccxmlTracelogLevel	CCXML Browser	String	off
	sessionTracelogLevel	Session Manager	String	off
	mediaMgrTracelogLevel	Media Manager	String	off
	mgtTracelogLevel	MPP System Manager	String	off
	avbPromptLogLevel	Voice Browser Prompt	String	off
	avbRecLogLevel	Voice Browser Recognition	String	off
	mediaVideoMgrTracelogLevel	Media Video Manager	String	off
	teleTracelogLevel	Telephony	String	off
	mediaEndpointMgrTracelogLevel	Media Endpoint Manager	String	off
	mrpcTracelogLevel	MCRP	String	off

Error cases

If a user inputs a value other than 'off', 'fine', 'finer' or 'finest' for trace level the request will be unsuccessful. They will receive a 400 error along with the message 'Error: unrecognized trace level'

Retrieve list of EPMs

Overview

The purpose of this feature is to allow cloud engineers to retrieve the names of the configured EPM server(s).

The EPM list web service MPP Settings web service can be found at:

`https://<hostaddress>/EPWebServices/rest-api/#/EPM%20Servers`

GET

To retrieve the list of configured EPMs a GET request is used.

Location: `https://<hostaddress>/EPWebServices/rest-api/#/EPM%20Servers`

Response code on success

200

Response body example

```
[  
  "EPM Primary1",  
  "EPM Aux 2",  
  "EPM Aux 3"  
]
```


Chapter 4: Resources

Documentation

The following table lists the documents related to Experience Portal. Download the documents from the Avaya Support website at <http://www.avaya.com/support>:

Title	Description	Audience
<i>Avaya Experience Portal Documentation Roadmap</i>	Lists all the documents related to Experience Portal and describes the organization of content across the documents.	Avaya Professional Services Implementation engineers
<i>Avaya Experience Portal Overview and Specification</i>	Describes tested product characteristics and capabilities, including product overview and feature descriptions, interoperability, performance specifications, security, and licensing requirements.	Implementation engineers
<i>Implementing Avaya Experience Portal on a single server</i>	Provides procedures to install and configure the Avaya Experience Portal software on a single server.	Implementation engineers
<i>Implementing Avaya Experience Portal on multiple servers</i>	Provides procedures to install and configure Avaya Experience Portal software on two or more dedicated servers.	Implementation engineers
<i>Deploying Avaya Experience Portal in an Avaya Customer Experience Virtualized Environment</i>	Provides procedures for deploying the Experience Portal virtual application in the Avaya Customer Experience Virtualized Environment. This document includes installation, configuration, initial administration, troubleshooting, and basic maintenance checklists and procedures.	Implementation engineers

Table continues...

Title	Description	Audience
<i>Administering Avaya Experience Portal</i>	Provides general information about and procedures for administering and configuring specific Experience Portal functions and features using a web-based interface.	Implementation engineers
<i>Troubleshooting Avaya Experience Portal</i>	Provides general information about troubleshooting and resolving system problems. This document also provides detailed information and procedures for finding and resolving specific problems.	Implementation engineers
<i>Avaya Experience Portal Security White Paper</i>	Provides information about the security strategy for Experience Portal, and provides suggestions that companies can use to improve the security of the Experience Portal systems and applications.	Avaya Professional Services Implementation engineers
Avaya Experience Portal 8.0 Mobile Web Best Practices White Paper	Provides recommended strategies for deploying Avaya Orchestration Designer Mobile Web applications with Avaya Experience Portal 8.0, detailing configuration for security, scalability and high availability.	Avaya Professional Services Implementation engineers

Finding documents on the Avaya Support website

Procedure

- Go to <https://support.avaya.com>.
- At the top of the screen, type your username and password and click **Login**.
- Click **Support by Product > Documents**.
- In **Enter your Product Here**, type the product name and then select the product from the list.
- In **Choose Release**, select the appropriate release number.
The **Choose Release** field is not available if there is only one release for the product.
- In the **Content Type** filter, click a document type, or click **Select All** to see a list of all available documents.

For example, for user guides, click **User Guides** in the **Content Type** filter. The list only displays the documents for the selected category.

- Click **Enter**.



Avaya Documentation Center navigation

The latest customer documentation for some programs is now available on the Avaya Documentation Center website at <https://documentation.avaya.com>.


Important:

For documents that are not available on Avaya Documentation Center, click **More Sites > Support** on the top menu to open <https://support.avaya.com>.

Using the Avaya Documentation Center, you can:

- Search for content by doing one of the following:
 - Click **Filters** to select a product and then type key words in **Search**.
 - From **Products & Solutions**, select a solution category and product, and then select the appropriate document from the list.
- Sort documents on the search results page.
- Click **Languages** () to change the display language and view localized) to change the display language and view localized documents.
- Publish a PDF of the current section in a document, the section and its subsections, or the entire document.
- Add content to your collection by using **My Docs** ().

Navigate to the **Manage Content > My Docs** menu, and do any of the following:

- Create, rename, and delete a collection.
- Add topics from various documents to a collection.
- Save a PDF of selected content in a collection and download it to your computer.
- Share content in a collection with others through email.
- Receive collection that others have shared with you.
- Add yourself as a watcher using the **Watch icon** ().).

Navigate to the **Manage Content > Watchlist** menu, and do the following:

- Enable **Include in email notification** to receive email alerts.
- Unwatch selected content, all content in a document, or all content on the Watch list page.

As a watcher, you are notified when content is updated or deleted from a document, or the document is removed from the website.

- Share a section on social media platforms, such as Facebook, LinkedIn, and Twitter.

- Send feedback on a section and rate the content.

 **Note:**

Some functionality is only available when you log on to the website. The available functionality depends on the role with which you are logged in.

Viewing Avaya Mentor videos

Avaya Mentor videos provide technical content on how to install, configure, and troubleshoot Avaya products.

About this task

Videos are available on the Avaya Support website, listed under the video document type, and on the Avaya-run channel on YouTube.

- To find videos on the Avaya Support website, go to <https://support.avaya.com/> and do one of the following:
 - o In **Search**, type `Avaya Mentor Videos`, click **Clear All** and select **Video** in the **Content Type**.
 - o In **Search**, type the product name. On the Search Results page, click **Clear All** and select **Video** in the **Content Type**.

The **Video** content type is displayed only when videos are available for that product.

In the right pane, the page displays a list of available videos.

- To find the Avaya Mentor videos on YouTube, go to www.youtube.com/AvayaMentor and do one of the following:
 - o Enter a key word or key words in the **Search Channel** to search for a specific product or topic.
 - o Scroll down Playlists, and click a topic name to see the list of videos available for the topic. For example, Contact Centers.

 **Note:**

Videos are not available for all products.

Support

Go to the Avaya Support website at <https://support.avaya.com> for the most up-to-date documentation, product notices, and knowledge articles. You can also search for release notes, downloads, and resolutions to issues. Use the online service request system to create a service

request. Chat with live agents to get answers to questions, or request an agent to connect you to a support team if an issue requires additional expertise.