

# Mitel Open Integration Gateway

## DEVELOPER GUIDE – SESSION MANAGEMENT SERVICE

Release 3.0

November 2015



## **NOTICE**

The information contained in this document is believed to be accurate in all respects but is not warranted by Mitel Networks™ Corporation (Mitel®). The information is subject to change without notice and should not be construed in any way as a commitment by Mitel or any of its affiliates or subsidiaries. Mitel and its affiliates and subsidiaries assume no responsibility for any errors or omissions in this document. Revisions of this document or new editions of it may be issued to incorporate such changes.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>). This product includes cryptographic software written by Eric Young ([eay@cryptsoft.com](mailto:eay@cryptsoft.com)). This product includes software written by Tim Hudson ([tjh@cryptsoft.com](mailto:tjh@cryptsoft.com)).

No part of this document can be reproduced or transmitted in any form or by any means - electronic or mechanical - for any purpose without written permission from Mitel Networks Corporation.

## **TRADEMARKS**

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

### **Mitel Open Integration Gateway Developer Guide - Session Management Service Release 3.0 November 2015**

®,™ Trademark of Mitel Networks Corporation  
© Copyright 2014-2015, Mitel Networks Corporation  
All rights reserved

INTRODUCTION .....	1
Mitel OIG web services .....	2
Mitel OIG documentation .....	3
Web Service Messaging Formats (SOAP & REST).....	4
Messaging formats guidelines.....	4
MITEL OIG SESSION MANAGEMENT SERVICE.....	6
Log in parameters .....	6
Web services types .....	7
MITEL OIG SESSION MANAGEMENT SERVICE OPERATIONS .....	8
Special Instructions for providing a Mitel certificate in the loginEx operation .....	9
Special Instructions for signing Authentication Data when using the Mitel OIG authenticate operation during Advanced Login.....	10
MITEL OIG SESSION MANAGEMENT SERVICE OPERATIONS USING SOAP / XML .....	11
Session Service - Standard TypeloginEx.....	11
logout.....	12
resetSessionTimer .....	12
serviceVersions .....	13
Session Service – Advanced type.....	14
loginEx.....	14
authenticate .....	15
MITEL OIG SESSION MANAGEMENT SERVICE OPERATIONS USING REST / JSON .....	16
REST API Versioning.....	16
Sessions Resource .....	16
Get Operation Login.....	16
Put Operation Reset Session Timer.....	17
Delete Operation Logout .....	17
GLOSSARY .....	18



## Introduction

The Mitel Open Integration Gateway (Mitel OIG) is a web server that provides each Mitel OIG application a single point of access to web services available within a Mitel communication system.

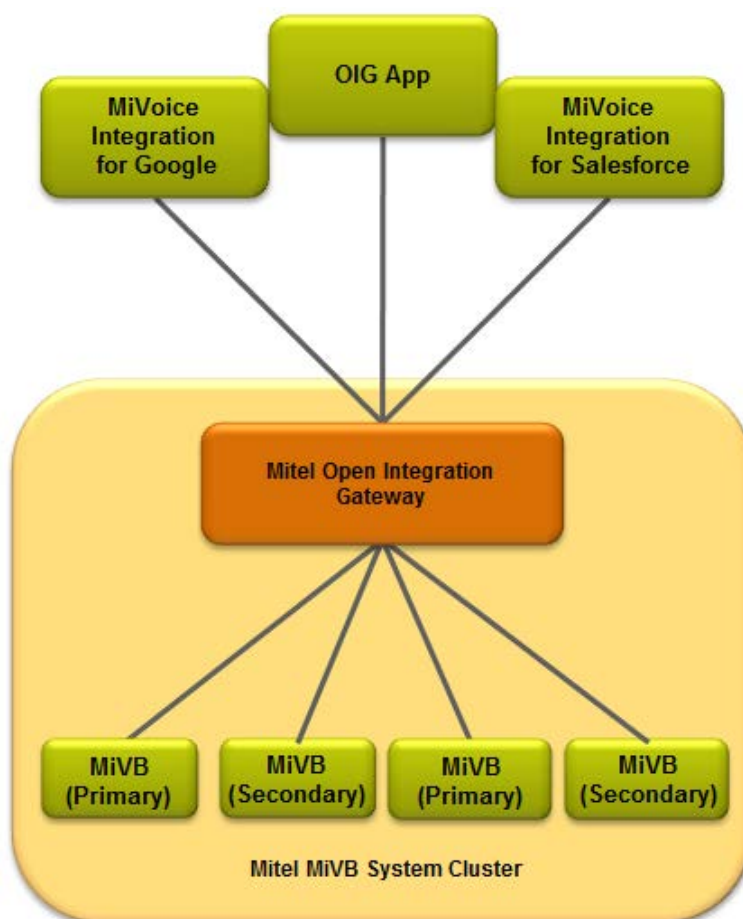
The Mitel OIG uses a services-oriented architecture. A Mitel OIG application opens a communication session with a Mitel OIG by logging in (example: sending a service operation or request to the Mitel OIG). After the Mitel OIG application is authenticated and authorized, the application can use this one communication session to access all of the Mitel OIG web services that the application is authorized to use.

The Mitel OIG allows applications to access features and functionality offered by a MiVoice Business system cluster.



**Note:** The Mitel OIG can communicate with a single MiVoice Business or a cluster of MiVoice Business instances. When there are two or more MiVoice Business instances, the MiVoice Business instances must be configured in a cluster. Mitel Open Integration Gateway cannot communicate with more than one MiVoice Business cluster. The Mitel OIG assumes the directory number (DN) of a Mitel phone is unique in the MiVoice Business system cluster; two Mitel phones in the system cannot have the same DN.

**Figure 1: Mitel OIG system configuration**



## Mitel OIG web services

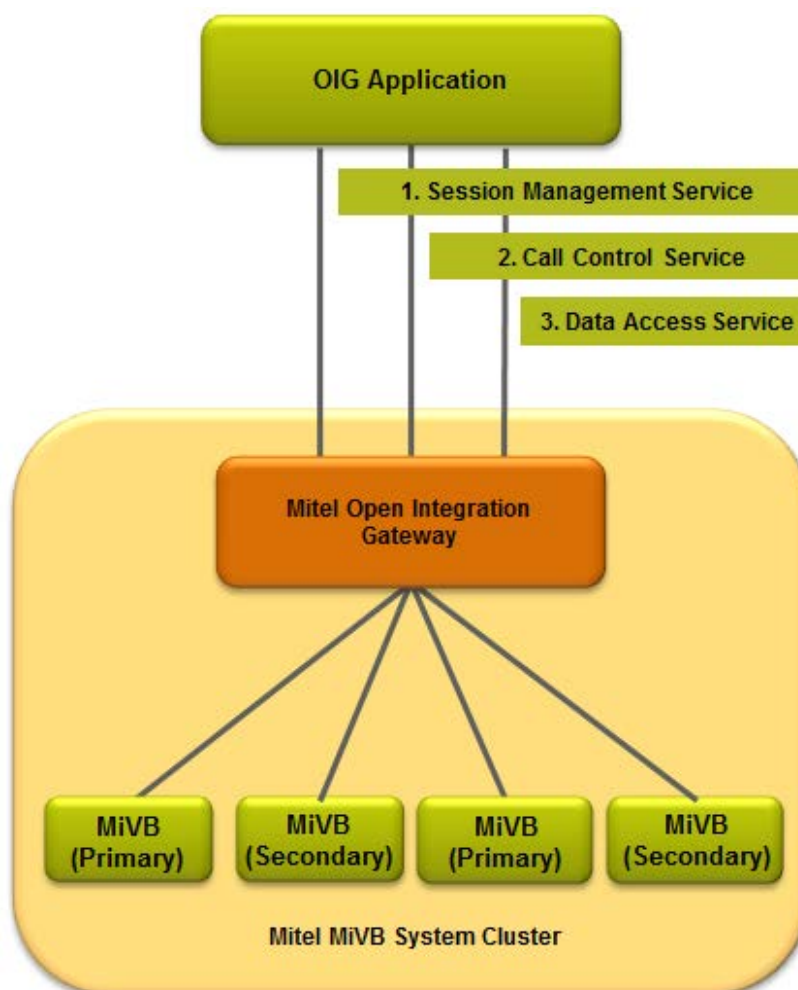
The Mitel OIG supports the following web services:

- Session Management service - Open communication session with Mitel OIG for services
- Call Control service - Control and monitor CTI behavior in Mitel communication system
- Data Access service - Register for MiVoice Business configuration data change notifications and read MiVoice Business configuration data.



**Note:** Mitel also offers applications that use the Mitel OIG:

- MiVoice Integration for Salesforce
- MiVoice Integration for Google

**Figure 2: Mitel OIG application, server, and services relationship**

## Mitel OIG documentation

This developer guide is specific to the Mitel OIG Session Management service. The following Mitel OIG documents are also available.

### **Mitel OIG Installation and Maintenance Guide**

This document provides details and instructions for installing the Mitel OIG and licensing it for applications and services, including MiVoice Integration applications.

### **Mitel OIG Engineering Guidelines**

The Engineering Guidelines provides guidance on network and system level requirements and performance.

## **Mitel OIG Developer Guide - Fundamentals**

The fundamentals guide introduces the Mitel OIG application developer environment and general information that applies to developing applications relative for any of the Mitel OIG web services.



**Note:** Mitel recommends that you become familiar with the content of the *Mitel OIG Developer Guide - Fundamentals* before attempting to create Mitel OIG applications.

## **Mitel OIG Developer Guide - Data Access Service**

This developer guide provides application developers detailed requirements for working with the MiVoice Business data management services..

## **Mitel OIG Developer Guide - Call Control Service**

This developer guide describes the Call Control Service details needed for creating telephony control applications.

## **MiVoice Integration for Salesforce Administration Guide**

This guide includes instructions for administrators setting up and maintaining MiVoice Integration for Salesforce for their organization.

## **MiVoice Integration for Salesforce User Guide**

This guide introduces the installation, functions, and use of MiVoice Integration for Salesforce.

## **MiVoice Integration for Google Administration Guide**

This guide includes instructions for administrators setting up and maintaining MiVoice Integration for Google for their organization.

## **MiVoice Integration for Google User Guide**

This guide introduces the installation, functions, and use of MiVoice Integration for Google.

## **Web Service Messaging Formats (SOAP & REST)**

The *Mitel OIG Developer Guide - Fundamentals* provides a summary of the web service messaging formats.

### **Messaging formats guidelines**

- A Mitel OIG application wanting to use the Call Control service (provided using SOAP / XML) will log in to a Mitel OIG using the Session Management service (provided using SOAP/ XML) using the loginEx operation.
- A Mitel OIG application wanting to use the Data Access service (provided using REST / JSON) can log in to a Mitel OIG using Session Management service (provided using REST / JSON) using the REST session resource.



A Mitel OIG application wanting to use both Call Control service (SOAP / XML) and Data Access service (REST / JSON) must log in to a Mitel OIG using Session Management service (provided using SOAP/ XML) using the loginEx operationRecent Changes Affecting Mitel OIG Applications

The *Mitel OIG Developer Guide - Fundamentals* provides a summary of all the changes introduced as part of this release.

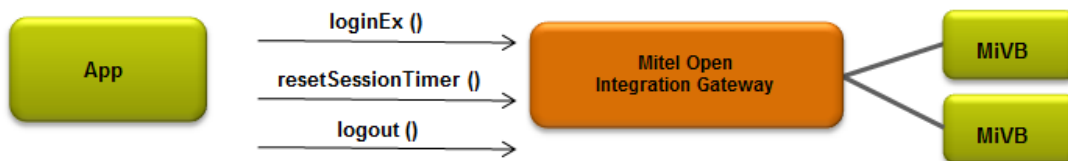
## Mitel OIG Session Management Service

Each Mitel OIG application uses the Mitel OIG session management service to communicate with the Mitel OIG. The session management service describes how an application logs in, logs out, and maintains a Mitel OIG communication session keep-alive timer. The application must reset a keep-alive timer every 10 seconds or the Mitel OIG will terminate the communication session.

The Mitel OIG Session Management Service allows an application to open a communication session with a Mitel OIG. After the communication session is established and the Mitel OIG application is authenticated and authorized, the application can use the one communication session to access all Mitel OIG web services.

See the Mitel OIG Developer Guides for more details about Mitel OIG services, operations and events.

**Figure 3: Session management**



### Log in parameters

To log in to a Mitel OIG, each application needs at least five parameters. Applications requiring advanced services must provide a Mitel certificate plus the following five parameters.

- company name
- application name
- application password
- local password
- version

Company name, application name, and application password are defined by the application developer when registering their application with Mitel. These three parameters do not change after they are defined in the application. The local password parameter is site-specific. A Mitel OIG administrator uses the Mitel OIG admin UI to create a local password specific to each application and a specific Mitel OIG. The local password controls the access of each individual application to each individual Mitel OIG. Each application that uses a Mitel OIG must provide a mechanism for a user to enter the local password, and the application must include the local password in the Mitel OIG loginEx operation, along with the company name, application name, and application password parameters. The version parameter is automatically included. The Mitel OIG Application does not need to provide the version in the application login request (loginEx operation). The version parameter allows the Mitel OIG to determine the service version used by the application.

## Web services types

By providing or not providing a Mitel Certificate during log in, the application indicates what overall type of web services (advanced or standard) the application is using with the Mitel OIG. The Mitel OIG offers two types of web services:

### *Standard*

The standard type web services provide the basic operations within Mitel OIG services. Standard operations focus on a system user; what an application needs to control and monitor what a user does within the Mitel system. To request this web services type, a Mitel OIG application needs an application account from Mitel.

### *Advanced*

Advanced type web services provide more sophisticated behavior within Mitel OIG services. Advanced type operations focus on overall system behaviors for many users. Advanced web services includes operations offered in the Standard web services type. To request Advanced web services type, a Mitel OIG application needs an application account and a Mitel certificate.

See the *Mitel OIG Developer Guide - Fundamentals* for more details about requesting an application account and a Mitel certificate for Advanced type web services. The *Mitel OIG Developer Guide - Fundamentals* provides more detail about all Mitel OIG services, operations, and events.

## Mitel OIG Session Management Service Operations

The Session Management Service allows an application to open a communication session with a Mitel OIG. If the application requires advanced Mitel OIG services, the application must use the loginEx and authenticate operations for Advanced Service Type. If the application only requires standard Mitel OIG services, the application will use only loginEX for the Standard Service Type.

The application must call resetSessionTimer with a frequency less than every 10 seconds.

The application can use (optional) serviceVersions to get software versions from the Mitel OIG. See the serviceVersions operation description below in the Session Service – Standard Type section of this document.

APPLICATION TO MITEL OIG SESSION	SERVICE TYPE
loginEx (localPassword, companyName, applicationName, applicationPassword, version) returns result (true/false), sessionId if true, errorDescription if false	Standard
loginEx (localPassword, companyName, applicationName, applicationPassword, version, certificate) returns result (true/false), errorDescription if false, authenticationData if true, sessionId if true <b>Note:</b> An Application must provide a Mitel certificate in the loginEx operation to access Mitel OIG Advanced Services.	Advanced
authenticate (signedAuthenticatedData, sessionId) returns result (true/false), errorDescription if false, sessionId if true <b>Note:</b> If result is false, the application is denied access to Advanced services. The authenticationData was provided by the Mitel OIG as part of login and must be signed by the application using the application private key. See the special instructions below about how to sign the authentication data.	Advanced
logout (sessionId) returns result (true/false), errorDescription if false <b>Note:</b> If an application does not close all monitors before logging out, the Mitel OIG ensures monitors are closed properly.	Both
resetSessionTimer (sessionId) returns result (true/false), errorDescription if false	Both
serviceVersions (sessionId) returns result (true/false), errorDescription if false, sessionId if true, serviceVersions if true. <b>Note:</b> This operations returns the version of the Mitel OIG and the version of any connected MiVoice Business.	Both

## Special Instructions for providing a Mitel certificate in the loginEx operation

An application must send a Mitel certificate within the Advanced loginEx operation and in the same PEM format as received in an email from Mitel. The developer must add carriage returns (\n must be added at the end of each line) and then turn the lines into a single string. For instance, assume the user has received the following Mitel certificate by email:

```
-----BEGIN CERTIFICATE-----
MIIDRDCCAiygAwIBAgIGAT0XHU7HMA0GCSqGSIb3DQEBAUAMFcxCzAJBgNVBAYT
AkNBMQswCQYDVQQIEwJPTjEPMA0GA1UEBxMGT3R0YXdhMQ4wDAYDVQQKEwVNaXRI
bDEMMAoGA1UECwwDUiZEMQwwCgYDVQQDEwNMQ1MwIBcNMTIwMjAxMTUyNzA5WhgP
MjExMjAyMDExNTI3MDIaMG0xDzANBgNVBAYTBkNhbmcFkYTELMAkGA1UECBMCT04x
DzANBgNVBAcTBk90dGF3YTEPMA0GA1UEChMGY29tMjAwMQwwCgYDVQQQLDANSJkQx
HTAbBgNVBAMTFGFwcDIwMDotLTpOb25UcnVzdGVkMIIBIjANBgkqhkiG9w0BAQEF
AAOCAQ8AMIIBCgKCAQEAlaD2AZaSmJNYa+5OYfDAFchwLPb6z6whulZDABdwTM0A
dzEu2ZofXF7dQY1zVV1hE8Lj4tO7p9vL2peKxROOB1zFb9NWRBfIKtozmp3Y2y/b
emo1paFiE8S4VfuyxmBi2Fq48IFzK90kpQy80DCxdhHVcgBQYZGFuPcFC+osiUOW
1KJXfejMzJl8wc+tfuKS2TDYZOSELIYaybC4NXSnmH5miYr5fDpG5aqTgfQQGqFO
7YYwQ2A3HkmOSv+jzHni5n4PeiVpuVF+IQoP28k+8D8YIZ6fNWiSf/2X9DshTKLu
142+a9NIR72zxohRizPrYvToJcl/wLA4tU9I62RMtQIDAQABMA0GCSqGSIb3DQEB
BAUAA4IBAQCefiPmNkiOqfLXSFmMErfUHHbH6vAaU9LzQ4Jdg1PLgPH0gktJ++6a
NZo4I9toDAJHxOVPemMyTW8C1CmeLrXmC2xv53EjhxstjoAHwR2z0HL+IOXF43zW
9BjE1jLsbALQT5uEABYrc2hfU5Kf0JwoKGxubtd7phxJhvi7onAZ3tiOurlSE44r
fgU897UI/AZdkmKNsjsmFy3ADSI1VC9s0Tjls5QEIORWe0StY8EotzPU7RnyoQTh
5TtWsHMYWrzoJAIRBeYSc/uH7a4UExubyhul6sadVqEatY9TpOqBpmhNda5lhYvL
IEFDnmUZyJG/8eGjztWdTi7oTcQ19exV
-----END CERTIFICATE-----
```

The developer must add carriage returns at the end of each line and then concatenate the lines. For example, in Java, the resulting Mitel certificate string will look as shown here when defined in code. This is the certificate that the application must provide in the Advanced loginEx operation to the Mitel OIG.

```
private String bpiCertPEM="-----BEGIN CERTIFICATE-----\n" +
"MIIDMDCCAhiGAWIBAgIGATiQf1jgMA0GCSqGSIb3DQEBAUAMFcxCzAJBgNVBAYT\n" +
"AkNBMQswCQYDVQQIEwJPTjEPMA0GA1UEBxMGT3R0YXdhMQ4wDAYDVQQKEwVNaXRI\n" +
"bDEMMAoGA1UECwwDUiZEMQwwCgYDVQQDEwNMQ1MwIBcNMTIwMjAxMTUyNzA5WhgP\n" +
"MjExMjAyMDExNTI3MDIaMG0xDzANBgNVBAYTBkNhbmcFkYTELMAkGA1UECBMCT04x\n" +
"A1UEBxMGT3R0YXdhMQ4wDAYDVQQKEwVNaXRIbDEMMAoGA1UECwwDUiZEMQ4wDAYD\n" +
"+
"VQQDEwVCUEIHVzCCASlwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAM3cqKHz\n" +
```

```
"Uz+pzALj3CNM3yhG4PyWCBH2t4nui8soNB0H2GZcKVQ2aHafOjQj2na6pM8tI8I9\n" +
"MUmq0YQ+Vlq/crFe4f4H/MxohqkW/R+VJJk/YvAmsRi3UA56StIAn9N8tVsW0IGu\n" +
"IxxBK/Z+gKOsQxhF5vXLBNiSplwQ9qgO8itwDtFkW3wcA571x09WoN3CI5z1bf\n" +
"/DSrAdhzWsk9vAobfgw1NTMrKIFy/X826I1Ls8ywXxx9KWizrk6ihmfYYeh60TPe\n" +
"00xVIYnxJnVa4VvmjE7UV2QkpWcBnhpa3WM0UmXNPcE9PlqfMhVYmQG0iEp1tXN3\n" +
"nYfX5oy02kdIbakCAwEAATANBgqhkiG9w0BAQQFAAOCAQEAC85fFBaf01eFp3S0\n" +
"efl88Sy8vMSSSKRsGz3mEu5S/2XvTu7nzlKppLBZrGBT1h2UiCxeG78VI1XmXUiu\n" +
"+ahO9ITCmdst5SBfdz2iAboeYBBEQ+DoZoKud5EmY0gueq8dTVbh7c7IGraCmpf\n" +
"yDIFvFNMhdV0NyHartDyUMk5ybUzKpSkfrlcZiW44JVnlmTAckZ04kl8PO0aXmOG\n" +
"w4jrvkCXwuJe4ePCrEQLf6152cyyShRdcx0MNAgsBOR7KQ8u/oGGknJ/wr+7LvZ\n" +
"RxDMlfbpjg2yYx7ITAT/tQ7JfJ5CQDh2XcNJw1vS3ewp4A1sHO4ydofHH5HvqkOb\n" +
"Oqi31g==\n" +
"-----END CERTIFICATE-----";
```

## Special Instructions for signing Authentication Data when using the Mitel OIG authenticate operation during Advanced Login

When calling the authenticate operation, an application must first sign the authentication data that is returned by the Mitel OIG. The authentication data is returned after the application calls the advanced loginEx operation. A session ID is also returned after the application calls the advanced loginEx operation. This session ID must be provided in the authenticate operation. Mitel has provided sample code in C# that demonstrates one way of doing this.

To sign the authentication data, the application must do the following:

Sign the data using the private key provided by Mitel during the Mitel certificate request process. The application must add the private key as a string (see instructions for formatting a Mitel certificate above) or in a key store. The Mitel OIG sample code uses a string. Also look at OpenSSL and BouncyCastle for examples. The signature algorithm needed is MD5.

Make sure the signature generated is an array of bytes with values ranging from -128 to 127 (meaning a signed representation of bytes). If the language in use generates an unsigned representation ("0-256") then the developer must cast the unsigned representation into a signed representation.

Finally, convert the array into a string of the following format : [13, -24, 6, 120, 43,...].

The final format must be as shown in the following example, and sent as a string argument of the authenticate operation:

```
[-108,116,-41,19,84,-100,-63,28,90,-26,36,-87,-91,70,-58,-101,-64,-53,-116,65,-
45,96,16,122,42,90,125,-120,42,-50,-91,-88,30,93,-25,-78,-3,65,117,23,-18,116,-52,113,-
15,75,95,127,-109,-21,70,18,-34,-30,69,-36,9,-46,12,-62,106,-64,102,-5,-125,-111,-4,-
128,59,70,101,83,7,-22,-94,-77,-116,-53,111,-43,74,112,-56,-110,-57,-60,57,94,-71,79,-66,9,-66,66,-
22,-11,-103,-45,119,88,-60,30,-63,11,-93,31,-85,79,-65,41,-32,14,-40,-80,40,-14,99,78,1,-42,61,-
88,37,1,-104,-117,62,4,59,-77,-6,66,47,102,105,86,-79,87,84,127,-4,86,-21,78,-89,123,20,-44,-
17,26,-103,-127,104,-31,31,-80,108,45,-3,44,28,-102,39,-61,-14,-56,31,-22,10,38,18,-
49,92,90,67,101,-39,-59,127,-118,-101,25,-93,127,-124,-70,-75,58,-119,-2,40,-56,29,-37,8,37,-102,-
63,13,125,29,-28,-83,-115,-85,-65,63,-8,34,127,14,85,62,99,-39,-3,-70 ...]
```

# Mitel OIG Session Management Service Operations Using SOAP / XML

## Session Service - Standard TypeloginEx

### Definition

loginEx (localPassword, companyName, applicationName, applicationPassword, version)

### Description

This operation creates a communication session between the application and the Mitel OIG. The application uses this operation with this set of attributes when requesting standard service from the Mitel OIG. The first four attributes are required and must be provided by the application. If the fifth attribute is not provided by the application, the Mitel OIG defaults to latest WSDL version.



**Note:** The Version attribute must be provided by the application for backward compatibility.

### Attributes

ATTRIBUTE	DESCRIPTION
localPassword (required)	Each application requires a local password defined in the Mitel OIG. The application must allow entry of the local password at run-time. The Mitel OIG administrator of a specific Mitel OIG creates a local password for each application. The application local password is unique to each Mitel OIG installation.
companyName (required)	The company name is known to the developer at application creation. companyName is NOT provided to the application at run-time; it is hard coded. companyName is provided to Mitel as part of Application Registration.
applicationName (required)	The application name is known to the developer at application creation. applicationName is NOT provided to the application at run-time; it is hard coded. applicationName is provided to Mitel as part of Application Registration.
applicationPassword (required)	The application password is known to the developer at application creation. applicationPassword is NOT provided to the application at run-time; it is hard coded. applicationPassword is provided to Mitel as part of Application Registration.
Version (optional)	The Version attribute must be set to match the version of the Mitel OIG server being used (eg, 2.1). If the application does not set this value, the Mitel OIG assumes the application is using a WSDL version matching its software version. For example, Mitel OIG 2.1 uses WSDL 2.1. The version attribute is used to identify the version of WSDL that the application is using.

### Returns

result – true or false

errorDescription – if result false

sessionId – if result true

### Notes

1. sessionId identifies the communication session between the application and the Mitel OIG.
2. The application must set the version attribute value to match the version of WSDL the application is using (i.e., 1.2). WSDL versions match Mitel OIG versions. When Mitel OIG 1.2 is released, the server supports WSDL version 1.2

logout

### Definition

logout (sessionId)

### Description

This operation terminates a communication session between the application and the Mitel OIG.

### Attributes

ATTRIBUTE	DESCRIPTION
sessionId	sessionId is provided by the Mitel OIG upon successful login.

### Returns

Result – true or false

errorDescription – if result false

### Notes

1. sessionId identifies the communication session between the application and the Mitel OIG.

resetSessionTimer

### Definition

resetSessionTimer (sessionId)



*Description*

This operation resets the keep-alive timer used to detect stale communication sessions between the application and the Mitel OIG. If the session timer is not reset, the Mitel OIG closes the communication session. If the session times out, the application must log in again and get a new sessionId.

*Attributes*

ATTRIBUTE	DESCRIPTION
sessionId	sessionId is provided by the Mitel OIG upon successful login.

*Returns*

result – true or false

errorDescription – if result false

*Notes*

2. sessionId identifies the communication session between the application and the Mitel OIG.

*serviceVersions**Definition*

serviceVersions (sessionId)

*Description*

This operation obtains the software version of the Mitel OIG and the version of any connected MiVoice Business.

*Attributes*

ATTRIBUTE	DESCRIPTION
sessionId	sessionId is provided by the Mitel OIG upon successful login.

*Returns*

result – true or false

errorDescription – if result false

serviceVersions – if result true the software versions for Mitel OIG and the connected MiVoice Business.

*Notes*

1. sessionId identifies the session between the application and the Mitel OIG.

## Session Service – Advanced type

### loginEx

#### Definition

loginEx (localPassword, companyName, applicationName, applicationPassword, version, certificate)

#### Description

This operation creates a session between the application and the Mitel OIG for advanced services. The application uses this operation with this set of attributes when requesting advanced service from the Mitel OIG. The first four attributes and the last attribute are required and must be provided by the application. If the fifth attribute is not provided by application, the Mitel OIG defaults to latest WSDL version.



**Note:** The Version attribute must be provided by the application for backward compatibility.

#### Attributes

ATTRIBUTE	DESCRIPTION
localPassword (required)	Each application requires a local password defined in the Mitel OIG. The application must allow entry of the local password at run-time. The Mitel OIG administrator of a specific Mitel OIG creates a local password for each application. The application local password is unique to each Mitel OIG installation.
companyName (required)	The company name is known to the developer at application creation. companyName is NOT provided to the application at run-time; it is hard coded. companyName is provided to Mitel as part of Application Registration.
applicationName (required)	The application name is known to the developer at application creation. applicationName is NOT provided to the application at run-time; it is hard coded. Application Name is provided to Mitel as part of Application Registration.  NOTE: Developers who start with Mitel Sample Apps MUST change the application name (and register the new name on the ACL) before deploying their modified app.
applicationPassword (required)	The application password is known to the developer at application creation. applicationPassword is NOT provided to the application at run-time; it is hard coded. applicationPassword is provided to Mitel as part of Application Registration.
Version (optional)	The Version attribute must be set to match the version of the Mitel OIG server being used (eg, 2.1). If the application does not set this value, the Mitel OIG assumes the application is using a WSDL version matching its software version. For example, Mitel OIG 2.1 uses WSDL 2.1. The Version attribute is used to identify the version of WSDL that the application is using.
Certificate (required)	Each application that requires advanced services from an Mitel OIG requires a Mitel certificate. Application developers request a Mitel certificate using the Mitel OnLine MSA web portal. The same certificate is used in all instances of an application.

*Returns*

result – true or false

errorDescription – if result false

authenticateData – if result true

sessionId - if result true

*Notes*

1. sessionId identifies the session between the application and the Mitel OIG.
2. authenticateData needs to be sent back to the Mitel OIG to authenticate the session. The application must sign the data before sending to the Mitel OIG. See authenticate operation below.
3. The application must set the version attribute value to match the version of WSDL the application is using (e.g., 2.1). WSDL versions match Mitel OIG versions. When Mitel OIG 2.1 is released, the server supports WSDL version 2.1.

*authenticate**Definition*

authenticate (signedAuthenticatedData, sessionId)

*Description*

This operation confirms that the application that provided the Mitel certificate has the associated private key. The application signs the authenticateData provided from the loginEx operation.

*Attributes*

ATTRIBUTE	DESCRIPTION
sessionId	sessionId is provided by the Mitel OIG upon successful login.
signedAuthenticatedData	The application must sign the authenticateData provided by the Mitel OIG with the application's private key. The application private key is provided by Mitel as part of the Mitel certificate process.

*Returns*

result – true or false

errorDescription – if result false

sessionId – if result true

*Notes*

1. sessionId identifies the session between the application and the Mitel OIG. If authenticate fails in the Mitel OIG, application access is denied.

## Mitel OIG Session Management Service Operations Using REST / Json

The Mitel OIG session management service is provided using REST and JSON over HTTPS. Applications do not require software from Mitel to communicate with a Mitel OIG. An application does not need to integrate or compile in any Mitel code. Application developers are free to choose a programming language, a software development environment, an operating system, and a hardware platform for their application. The web service model decouples the Mitel OIG software from the application.

The Mitel OIG session management service is defined using a request and response model. An application sends a request to activate a service operation and the Mitel OIG responds with success or failure and the return data. The application must check for the success or failure of each operation. Operations trigger changes in the Mitel OIG and the MiVoice Business system.

### REST API Versioning

All of the URLs supported by the Mitel OIG REST operations contain a version number such as “v1”. The version number will only change if the API for that operation has changed. The following is an example of one of the URLs:

`http(s)://OIGFQDN/mitel/oig/rest/resources/v1/session` (where OIGFQDN can be the Mitel OIG server IP address or the FQDN of the Mitel OIG server).

If the API for the GET operation changed for this resource the new URL would be the following:

`http(s)://oigIPAddress/mitel/oig/rest/resources/v2/ session` (where oigIPAddress can be the Mitel OIG server IP address or the FQDN of the Mitel OIG server).



**Note:** If possible the Mitel OIG will strive to be backwards compatible. Therefore, it's possible that the Mitel OIG would support both a v1/session login operation as well as v2/session login operation but the interfaces could be different.

### Sessions Resource

The Sessions resources are available here:

URL: `http(s)://OIGFQDN/mitel/oig/rest/resources/v1/sessions`



**Note:** OIGFQDN can be the Mitel OIG server IP address or the FQDN of the Mitel OIG server.

#### Get Operation Login

Equivalent SOAP operation is “loginEx”.

**HTTP Header Parameters:**

- “Authorization” string – localPassword
- “ApplicationInfo” string - applicationPassword

**Query Parameters:**

- applicationName
- companyName

**Return Parameters:**

- result – Boolean indicating operation success or failure
- error – If operation failed
- sessionId – If operation was successful

This operation is used by an application to establish an application session with a specific Mitel OIG. A globally unique “sessionId” is returned by the Mitel OIG server in the response message. The “sessionId” returned in the response gives the application “Standard” services access.

**Put Operation Reset Session Timer**

Equivalent SOAP operation is “resetSessionTimer”.

**HTTP Header Parameters:**

- “Authorization” string – sessionId

**Return Parameters:**

- result – Boolean indicating operation success or failure
- error – If operation failed

This operation is used by an application to reset the inactivity timer associated with an application session. The Mitel OIG “SessionManager” will terminate any sessions which have not had any activity in 10 minutes.

**Delete Operation Logout**

Equivalent SOAP operation is “logout”.

**HTTP Header Parameters:**

- “Authorization” string – sessionId

**Return Parameters:**

- result – Boolean indicating operation success or failure
- error – If operation failed

This operation is used by an application to terminate an application session.

## Glossary

ACD	Automatic Call Distribution
ACL	Access Control List
AMC	Applications Management Center (licensing server)
API	Application Programming Interface
CCS	Call Control Service
COS	Class of Service
DLL	Dynamic Link Library
DMZ	De-Militarized Zone
DNS	Domain Name Server
ICP	IP Communications Platform
IP	Internet Protocol
IVR	Interactive Voice Response
LAN	Local Area Network
MCS	Mitel Certificate Server
MiTAI	Mitel Telephony Application Interface
MiVB	MiVoice Business
MOL	Mitel OnLine
MSA	Mitel Solutions Alliance (Mitel developer partner program)
MSL	Mitel Standard Linux (operating system)
MSP	Media Service Provider
Mitel OIG	Open Integration Gateway
PBX	Private Branch Exchange
PSTN	Public Switched Telephone Network
TDM	Time Division Multiplexing
VOIP	Voice over IP
vLAN	Virtual Local Area Network
WAN	Wide Area Network
WSDL	Web Service Description Language

