

## **Cisco CallManager Extension Mobility API Developer's Guide**

### **Corporate Headquarters**

Cisco Systems, Inc.  
170 West Tasman Drive  
San Jose, CA 95134-1706  
USA

<http://www.cisco.com>

Tel: 408 526-4000  
800 553-NETS (6387)

Fax: 408 526-4100

Text Part Number: OL-1406-01

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

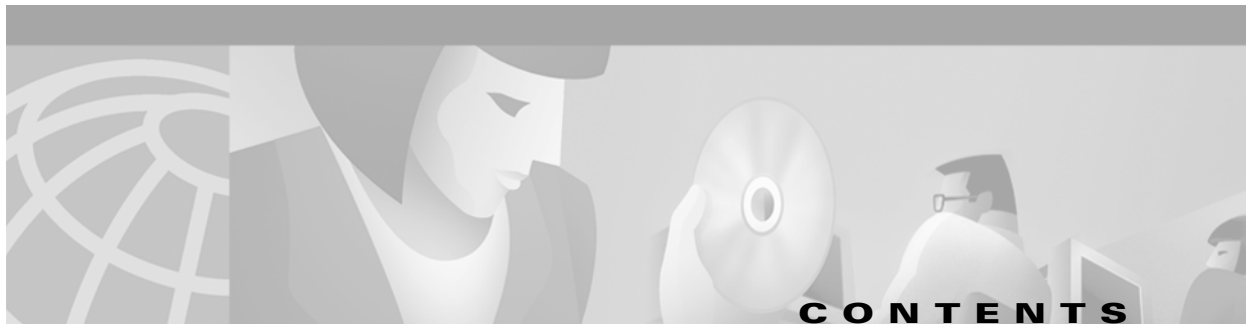
NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

AccessPath, AtmDirector, Browse with Me, CCIP, CCSI, CD-PAC, *CiscoLink*, the Cisco *Powered* Network logo, Cisco Systems Networking Academy, the Cisco Systems Networking Academy logo, Fast Step, Follow Me Browsing, FormShare, FrameShare, GigaStack, IGX, Internet Quotient, IP/VC, iQ Breakthrough, iQ Expertise, iQ FastTrack, the iQ Logo, iQ Net Readiness Scorecard, MGX, the Networkers logo, *Packet*, RateMUX, ScriptBuilder, ScriptShare, SlideCast, SMARTnet, TransPath, Unity, Voice LAN, Wavelength Router, and WebViewer are trademarks of Cisco Systems, Inc.; Changing the Way We Work, Live, Play, and Learn, Discover All That's Possible, and Empowering the Internet Generation, are service marks of Cisco Systems, Inc.; and Aironet, ASIST, BPX, Catalyst, CCDA, CCDP, CCIE, CCNA, CCNP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, the Cisco IOS logo, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Enterprise/Solver, EtherChannel, EtherSwitch, FastHub, FastSwitch, IOS, IP/TV, LightStream, MICA, Network Registrar, PIX, Post-Routing, Pre-Routing, Registrar, StrataView Plus, Stratm, SwitchProbe, TeleRouter, and VCO are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and certain other countries.

All other trademarks mentioned in this document or Web site are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0106R)

*Cisco CallManager Extension Mobility API Developer's Guide*  
Copyright © 2001, Cisco Systems, Inc.  
All rights reserved.



## **Preface v**

- Audience **v**
- Organization **v**
- Related Documentation **vi**
- Conventions **vi**
- Obtaining Documentation **vii**
  - World Wide Web **vii**
  - Documentation CD-ROM **vii**
  - Ordering Documentation **vii**
  - Documentation Feedback **viii**
- Obtaining Technical Assistance **viii**
  - Cisco.com **viii**
  - Developer Support **ix**
  - Technical Assistance Center **ix**
    - Contacting TAC by Using the Cisco TAC Website **ix**
    - Contacting TAC by Telephone **x**

---

## **CHAPTER 1**

### **Understanding Cisco CallManager Extension Mobility 1-1**

- Cisco CallManager Extension Mobility Architecture **1-2**
  - The Cisco CallManager Extension Mobility System Components **1-2**
  - How the Cisco CallManager Extension Mobility System Works **1-3**
  - The Cisco CallManager Extension Mobility Login Service **1-4**
- Device Profiles **1-5**
  - Logout Device Profile **1-5**
- Login Policy **1-6**
- Automatic Logout **1-6**
- Authentication **1-6**

---

## **CHAPTER 2**

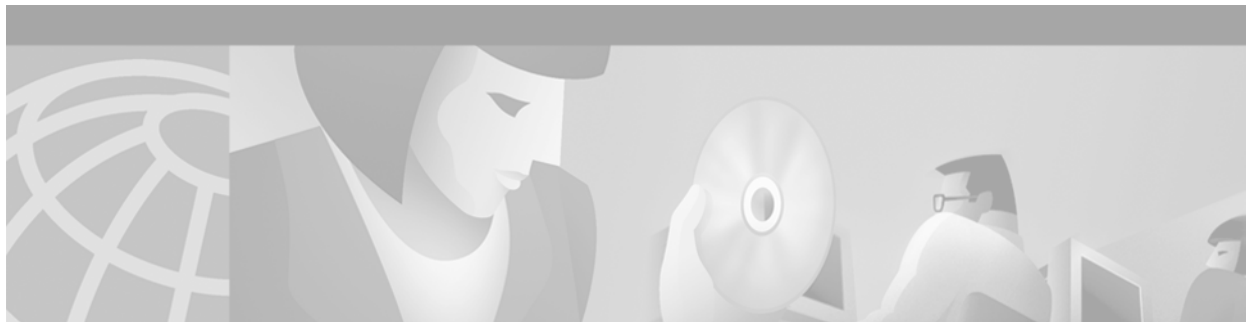
### **Using the Cisco CallManager Extension Mobility API 2-1**

- Introduction **2-1**
- Configuration **2-1**
- Messages **2-2**
  - Login Requests **2-2**

- Logout Requests [2-2](#)
- Device-User Queries [2-3](#)
- User-Devices Queries [2-3](#)
- Securing Cisco CallManager Extension Mobility Messages [2-3](#)
- Message Document Type Definitions [2-3](#)
  - The Request DTD [2-4](#)
  - The Login or Logout Response DTD [2-4](#)
  - The Query DTD [2-4](#)
  - The Query Response DTD [2-5](#)
- Message Examples [2-5](#)
  - Request Examples [2-6](#)
  - Request Response Examples [2-8](#)
  - Query Examples [2-8](#)
  - Query Response Examples [2-9](#)
- Login Service Error Codes [2-10](#)

---

**INDEX**



## Preface

---

Cisco CallManager Extension Mobility is a feature of Cisco CallManager Release 3.1 that allows a device, usually a Cisco IP Phone, to temporarily embody a new device profile, including lines, speed dials, and services.

The Cisco CallManager Extension Mobility Service provides a fairly rich API which enables Extension Mobility on 7960 and 7940 IP phones, allowing application control over authentication, scheduling, and availability.

An application that uses Cisco CallManager Extension Mobility is the Cisco IP Phone 7960 service which allows a user to enter a userID and PIN at the phone itself and log into the phone. Due to the architecture and implementation of Extension Mobility, many other applications are possible.

Some examples are:

- An application which automatically activates phones for employees when they reserve a particular desk for a particular time (the scheduling application).
- A lobby phone which does not have a line appearance until it is logged into.

## Audience

The *Cisco CallManager Extension Mobility API Developer's Guide* is for developers who write applications that extend the functionality of Cisco CallManager Extension Mobility. You should be familiar with Extensible Markup Language (XML).

## Organization

This guide is organized as follows:

**Table 1**     **Organization**

Chapter	Title
Chapter 1	Understanding Cisco CallManager Extension Mobility
Chapter 2	Using the Cisco CallManager Extension Mobility API

## Related Documentation

For more information on Extension Mobility, refer to the following documents:

- *Cisco CallManager Administration Guide*
- *Cisco CallManager System Guide*
- *Cisco CallManager Extended Services Administrator's Guide*

## Conventions

This document uses the following conventions:

**Table 2**    **Conventions**

Convention	Description
<b>boldface font</b>	Commands and keywords are in <b>boldface</b> .
<i>italic font</i>	Arguments for which you supply values are in <i>italics</i> .
[ ]	Elements in square brackets are optional.
{ x   y   z }	Alternative keywords are grouped in braces and separated by vertical bars.
[ x   y   z ]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
screen font	Terminal sessions and information the system displays are in <code>screen font</code> .
<b>boldface screen font</b>	Information you must enter is in <b>boldface screen font</b> .
<i>italic screen font</i>	Arguments for which you supply values are in <i>italic screen font</i> .
^	The symbol ^ represents the key labeled Control—for example, the key combination ^D in a screen display means hold down the Control key while you press the D key.
< >	Nonprinting characters, such as passwords are in angle brackets.

Notes use the following conventions:



### Note

Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the publication.

Timesavers use the following conventions:



### Timesaver

Means *the described action saves time*. You can save time by performing the action described in the paragraph.

Tips use the following conventions:

**Tip**

---

Means *the following are useful tips*.

---

Cautions use the following conventions:

**Caution**

---

Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

---

Warnings use the following conventions:

**Warning**

---

**This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, you must be aware of the hazards involved with electrical circuitry and familiar with standard practices for preventing accidents.**

---

## Obtaining Documentation

The following sections provide sources for obtaining documentation from Cisco Systems.

### World Wide Web

You can access the most current Cisco documentation on the World Wide Web at the following sites:

- <http://www.cisco.com>
- <http://www-china.cisco.com>
- <http://www-europe.cisco.com>

### Documentation CD-ROM

Cisco documentation and additional literature are available in a CD-ROM package, which ships with your product. The Documentation CD-ROM is updated monthly and may be more current than printed documentation. The CD-ROM package is available as a single unit or as an annual subscription.

### Ordering Documentation

Cisco documentation is available in the following ways:

- Registered Cisco Direct Customers can order Cisco Product documentation from the Networking Products MarketPlace:  
[http://www.cisco.com/cgi-bin/order/order\\_root.pl](http://www.cisco.com/cgi-bin/order/order_root.pl)
- Registered Cisco.com users can order the Documentation CD-ROM through the online Subscription Store:  
<http://www.cisco.com/go/subscription>

- Nonregistered Cisco.com users can order documentation through a local account representative by calling Cisco corporate headquarters (California, USA) at 408 526-7208 or, in North America, by calling 800 553-NETS(6387).

## Documentation Feedback

If you are reading Cisco product documentation on the World Wide Web, you can submit technical comments electronically. Click **Feedback** in the toolbar and select **Documentation**. After you complete the form, click **Submit** to send it to Cisco.

You can e-mail your comments to [bug-doc@cisco.com](mailto:bug-doc@cisco.com).

To submit your comments by mail, use the response card behind the front cover of your document, or write to the following address:

Attn Document Resource Connection  
Cisco Systems, Inc.  
170 West Tasman Drive  
San Jose, CA 95134-9883

We appreciate your comments.

## Obtaining Technical Assistance

Cisco provides Cisco.com as a starting point for all technical assistance. Customers and partners can obtain documentation, troubleshooting tips, and sample configurations from online tools. For Cisco.com registered users, additional troubleshooting tools are available from the TAC website.

## Cisco.com

Cisco.com is the foundation of a suite of interactive, networked services that provides immediate, open access to Cisco information and resources at anytime, from anywhere in the world. This highly integrated Internet application is a powerful, easy-to-use tool for doing business with Cisco.

Cisco.com provides a broad range of features and services to help customers and partners streamline business processes and improve productivity. Through Cisco.com, you can find information about Cisco and our networking solutions, services, and programs. In addition, you can resolve technical issues with online technical support, download and test software packages, and order Cisco learning materials and merchandise. Valuable online skill assessment, training, and certification programs are also available.

Customers and partners can self-register on Cisco.com to obtain additional personalized information and services. Registered users can order products, check on the status of an order, access technical support, and view benefits specific to their relationships with Cisco.

To access Cisco.com, go to the following website:

<http://www.cisco.com>



## Developer Support

The Developer Support Program provides formalized support for Cisco Systems interfaces to enable developers, customers, and partners in the Cisco Service Provider Solutions Ecosystem and Cisco AVVID Partner programs to accelerate their delivery of compatible solutions.

The Developer Support Engineers are an extension of the product technology engineering teams. They have direct access to the resources necessary to provide expert support in a timely manner.

For additional information on this program, refer to the Developer Support Program Web Site at [www.cisco.com/go/developer-support/](http://www.cisco.com/go/developer-support/).

Developers using the Cisco CallManager Extension Mobility API are encouraged to join the Cisco Developer Support Program. This new program provides a consistent level of support while leveraging Cisco interfaces in development projects.

**Note**

---

Cisco Technical Assistance Center (TAC) support does not include Cisco CallManager Extension Mobility API support and is limited to Cisco AVVID installation/configuration and Cisco-developed applications. For more information about the Developer Support Program, please contact Cisco at [developer-support@cisco.com](mailto:developer-support@cisco.com).

---

## Technical Assistance Center

The Cisco TAC website is available to all customers who need technical assistance with a Cisco product or technology that is under warranty or covered by a maintenance contract.

### Contacting TAC by Using the Cisco TAC Website

If you have a priority level 3 (P3) or priority level 4 (P4) problem, contact TAC by going to the TAC website:

<http://www.cisco.com/tac>

P3 and P4 level problems are defined as follows:

- P3—Your network performance is degraded. Network functionality is noticeably impaired, but most business operations continue.
- P4—You need information or assistance on Cisco product capabilities, product installation, or basic product configuration.

In each of the above cases, use the Cisco TAC website to quickly find answers to your questions.

To register for Cisco.com, go to the following website:

<http://www.cisco.com/register/>

If you cannot resolve your technical issue by using the TAC online resources, Cisco.com registered users can open a case online by using the TAC Case Open tool at the following website:

<http://www.cisco.com/tac/caseopen>

## Contacting TAC by Telephone

If you have a priority level 1 (P1) or priority level 2 (P2) problem, contact TAC by telephone and immediately open a case. To obtain a directory of toll-free numbers for your country, go to the following website:

<http://www.cisco.com/warp/public/687/Directory/DirTAC.shtml>

P1 and P2 level problems are defined as follows:

- P1—Your production network is down, causing a critical impact to business operations if service is not restored quickly. No workaround is available.
- P2—Your production network is severely degraded, affecting significant aspects of your business operations. No workaround is available.



# Understanding Cisco CallManager Extension Mobility

---

Cisco CallManager Extension Mobility is a feature of Cisco CallManager Release 3.1 that allows a device, usually a Cisco IP Phone, to temporarily embody a new device profile, including lines, speed dials, and services.

You can use the XML-based Cisco CallManager Extension Mobility API with your applications so that they can take advantage of Cisco CallManager Extension Mobility functionality. For details about how to use the Cisco CallManager Extension Mobility API, refer to [Using the Cisco CallManager Extension Mobility API](#).

To successfully develop an application that uses the Cisco CallManager Extension Mobility service, it is important to understand how the service operates and how your application fits into the Cisco CallManager Extension Mobility system.

This chapter includes these high-level concepts, important in understanding the Cisco CallManager Extension Mobility system:

- [Cisco CallManager Extension Mobility Architecture, page 1-2](#)
- [Device Profiles, page 1-5](#)
- [Login Policy, page 1-6](#)
- [Automatic Logout, page 1-6](#)
- [Authentication, page 1-6](#)

# Cisco CallManager Extension Mobility Architecture

This section explains the Cisco CallManager Extension Mobility system components and how they work together with your application. It also gives a more detailed explanation of how the Login Service component works because the Login Service is the main component that your application communicates with.

## The Cisco CallManager Extension Mobility System Components

[Table 1-1](#) lists the basic architectural components of the Cisco CallManager Extension Mobility system and a description of each component. The Cisco CallManager Extension Mobility system is composed of your application and the Cisco CallManager Extension Mobility service. Your application is the Login Application. Every other component belongs to the Cisco CallManager Extension Mobility service. To see a diagram of how all the components of the Cisco CallManager Extension Mobility system are related to each other, see [Figure 1-1 on page 1-3](#).

**Table 1-1 Cisco CallManager Extension Mobility System Components**

Component	Description
<sup>1</sup> Login Application	Any application or process which submits a request to the Login Service via the XML over HTTP interface.
Login Service	The service that validates requests from a Login Application and sends back request and query responses.
LDAP Directory	Lightweight Directory Access Protocol Directory (LDAP) is the directory that stores information for Cisco CallManager.
DBL	Database Layer (DBL) manages Cisco CallManager database transactions that involve data such as Device Profiles, etc.
Database Layer Monitor	The service that notifies other processes of changes in the Cisco CallManager database.
CallProcessing	The core Cisco CallManager process, responsible for maintaining device connections.
CTI	Computer Telephony Interface (CTI) is the set of processes that expose programmable APIs for call control.
TAPI/JTAPI	Telephony Application Programming Interfaces that support call control.

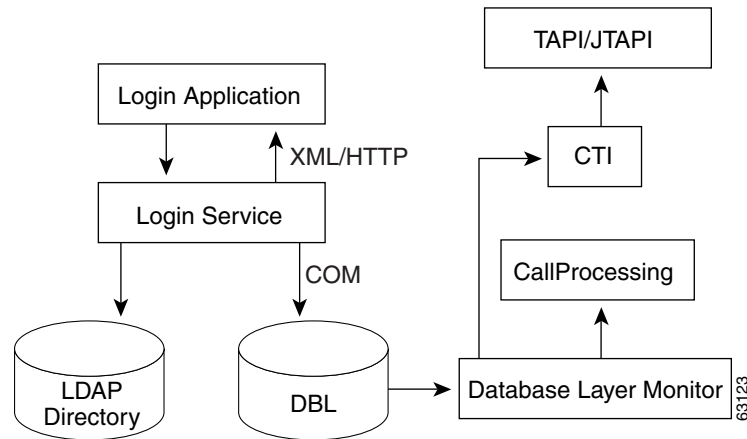
1. The Login Application is your application.

## How the Cisco CallManager Extension Mobility System Works

This section describes what happens when your application sends a message to the Login Service to use Cisco CallManager Extension Mobility functionality.

Figure 1-1 shows how Cisco CallManager Extension Mobility system components are connected to each other.

**Figure 1-1 The Cisco CallManager Extension Mobility System**



Your Login Application submits an XML message to the Login Service using HyperText Transfer Protocol (HTTP). The Login Service uses the LDAP Directory to check the application ID and password in the message from the Login Application.

If the application ID and password are valid, the Login Service executes the request by talking to the Database Layer (DBL) through Microsoft's Component Object Model interface (COM). For more details about how the Login Service works, see [“The Cisco CallManager Extension Mobility Login Service” section on page 1-4](#).

If the DBL changes the Device Profile for the device (for a login or logout request), it tells the Database Layer Monitor. The Database Layer Monitor passes this information on to the CallProcessing and CTI components. CallProcessing tells the Cisco IP phone that it needs to restart itself to load the new Device Profile. For information about Device Profiles, see the [“Device Profiles” section on page 1-5](#).

The CTI layer notifies JTAPI and TAPI applications that are monitoring the device or user that the application control list has changed.

When the DBL successfully completes a transaction, it tells the Login Service. The Login Service then sends an XML response that the transaction was successful back to your Login Application using HTTP.



### Note

If the transaction is not successful, the Login Service sends your Login Application an appropriate error message.

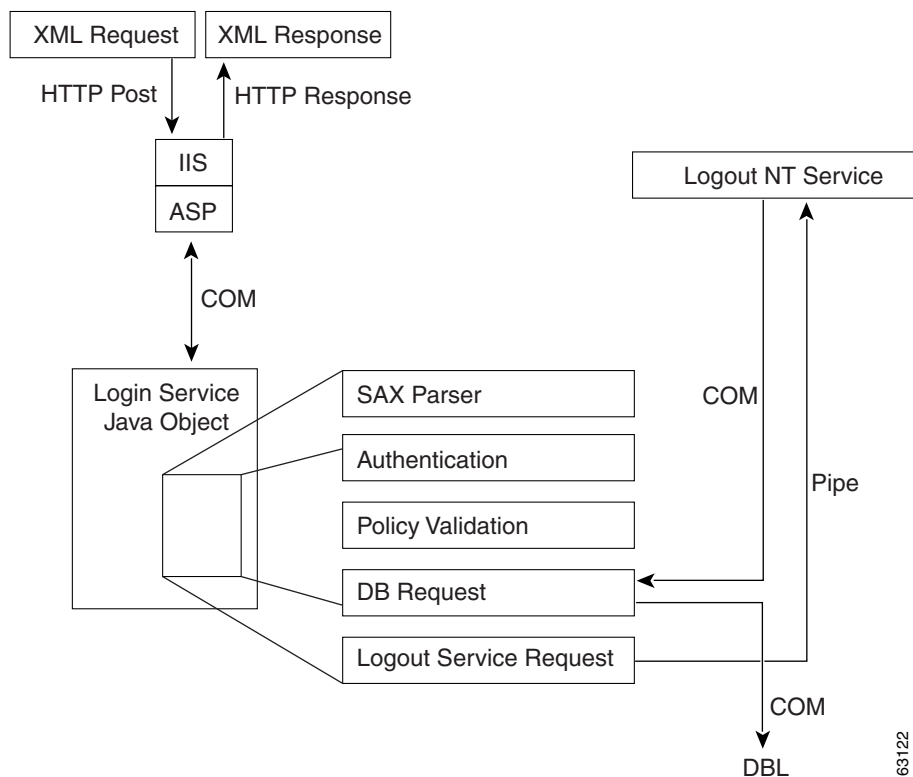
## The Cisco CallManager Extension Mobility Login Service

Your Login Application talks to the Cisco CallManager Extension Mobility service through the Cisco CallManager Extension Mobility Login Service component.

When the Login Service component receives an HTML message from your Login Application, it uses HTTP to send an XML response message back. The response to a request is a success or failure message and the response to a query is a query result message. For details about messages, see the “Messages” section on page 2-2.

Figure 1-2 shows a close-up of the Login Service component of the Cisco CallManager Extension Mobility system.

**Figure 1-2 The Cisco CallManager Extension Mobility Login Service Component**



The Login Service component has three main functional parts: the Active Server Pages (ASPs), the Login Service Java Object, and the Logout NT Service.

The Login Service ASPs are a small set of ASPs running under Microsoft Internet Information Server (IIS). The Login Service ASPs receive the HTTP POST request or query messages and pass the contained XML to the Login Service Java Object. The ASPs then take the results from the Login Service Java Object and pass them back to the Login Application as the HTTP response.

The Login Service Java Object is wrapped as a Dynamic Link Library (DLL). The Login Service Java Object has a SAX Parser that parses the XML message. The Login Service Java Object's Authentication module checks whether the application ID and password in the message are valid. For more information about authentication, see the “Authentication” section on page 1-6. The Login Service Java Object's Policy Validation engine checks the pre-conditions. For information about login policies, see the “Login Policy” section on page 1-6.

63122

Then, the DB Request module of the Login Service Java Object makes a request to the Database Layer (DBL).

The Logout NT Service implements the automatic logout feature. The Logout NT Service maintains the state of a device. The device state includes the login time and login duration if the automatic logout feature is invoked. When the login duration expires, the Logout NT Service sends a logout request to the DBL through the DB Request part of the Login Service Java Object. For information about the automatic logout feature, see the [“Automatic Logout” section on page 1-6](#).

**Note**

The Login Service component sends back an appropriate XML error response to your Login Application if authentication fails, a pre-condition is not met, it cannot contact the DBL, or the DBL returns an error.

**Note**

The Logout NT Service records Errors but does not report them.

## Device Profiles

The basic unit of transaction for Cisco CallManager Extension Mobility is a device profile. A device profile contains all the configuration information for a particular device, such as line appearances, speed dials, and services. It can be thought of as a "virtual device". It has all the properties of a device except physical characteristics such as a Media Access Control (MAC) address and a directory URL.

When a user logs in, the current device configuration is replaced by that users' User Device Profile. When a user logs out, the User Device Profile is replaced by the Logout Device Profile.

## Logout Device Profile

Cisco CallManager Extension Mobility requires a Logout Device Profile for each configured device. Cisco CallManager Extension Mobility uses the Logout Device Profile, which can be either an Auto-Generated or User Device Profile, as the "logged out" configuration of the device.

There are two types of device profiles, Auto-Generated Device Profiles and User Device Profiles.

- Auto-Generated Device Profile— can only be used as a Logout Device Profile. This is a snapshot of the existing device's configuration. It cannot be associated with a user.
- User Device Profile — is generated by an administrator and associated with a user in the same manner as any other device.

**Note**

To create an Auto-Generated Device Profile, a device is configured and a snapshot of the device is taken and saved as a device profile with the prefix ADP (Auto-Generated Device Profile) and the MAC address of the device. For example, the Auto-Generated Device Profile for the device SEP000011112222 is ADP000011112222.

**Note**

Cisco CallManager Extension Mobility fully supports the Cisco IP Phone 7960 and the Cisco IP Phone 7940 but not the IP Phone model 7910 and older devices.

## Login Policy

Only a single user is allowed to log in at a time on a particular device. Subsequent attempts by users to log in on a device before the previous user has logged out will fail. You can also not log out of a device to which no user has logged in. These are error conditions and generate error messages.

## Automatic Logout

A login request can specify a login duration, after which time the system automatically logs out the device. A system-wide maximum login duration can be specified. The login duration state is maintained by an NT service, the Cisco User Logout Service.

**Note**

---

The Cisco User Logout Service is used to time all login occurrences if you have specified a system maximum login time. If you have not set the login duration, the automatic logout period for that device defaults to the system maximum time.

---

## Authentication

The Cisco CallManager Extension Mobility Service allows authentication by proxy. That is, a user with Cisco CallManager Extension Mobility proxy rights is allowed to log in any user to any device.

What this means is that an application can be responsible for authenticating a user in whatever way the designer of the application sees fit: by using a password, PIN, hardware key, biometrics, etc. Then the application must provide valid credentials for itself (the application) so that the Cisco CallManager Extension Mobility Service knows that the application is provisioned in the system and allowed to log users in and out.

To this end, a special user that corresponds to the application must be configured in the Directory. This user, representing the application, has a standard LDAP userID and password, hereafter referred to as an appID and appPassword. The application must send a valid appID and appPassword to log in or log out a user from a device.

**Note**

---

This mechanism requires configuring a userID, the appID, for the application, which can be done via the Cisco CallManager User Administration.

---





# Using the Cisco CallManager Extension Mobility API

---

## Introduction

The Cisco CallManager Extension Mobility API is exposed as an Extensible Markup Language (XML) interface via HTTP. A website is designated by the administrator of the system as the entry point to the API, and all requests and queries are made through those URLs. The Document Type Definitions (DTDs) which define the XML for requests, queries, and responses are also made available at this website. The DTDs are included in this document, along with examples.

The XML input is submitted via an HTTP POST. A field named "xml" contains the XML string which defines the request or query. The response to this HTTP POST is a pure XML response with either a success or failure indicator for a request or the response to a query.

This chapter is organized as follows:

- [Configuration, page 2-1](#)
- [Messages, page 2-2](#)
- [Message Document Type Definitions, page 2-3](#)
- [Message Examples, page 2-5](#)
- [Login Service Error Codes, page 2-10](#)
- [, page 2-11](#)

## Configuration

Cisco CallManager Extension Mobility is an application designed to accompany Cisco CallManager Release 3.1. As such, all necessary Cisco CallManager Extension Mobility API components are installed with the standard Cisco CallManager installation.

To use Cisco CallManager Extension Mobility, a device profile must be created for the user logging in and for the target device.

These are the steps necessary to configure Cisco CallManager Extension Mobility:

- Create a User Device Profile.
- Assign a User Device Profile to a User.
- Assign authentication proxy rights to an appID.

- Assign a Logout Device Profile to a target device.
- Configure the System Parameters.

**Note**

- 
- Technically, it is not necessary to assign a profile to a user. The device profile may be specified at login time.
  - System Parameters use defaults if not manually configured.
  - Extension Mobility must be enabled on a device-by-device basis.
- 

For details on how to configure the User Device Profile, refer to the *Cisco CallManager Administration Guide* or the *Cisco CallManager Extended Services Administrator's Guide*.

## Messages

You communicate between your login application and the Cisco CallManager Extension Mobility Login Service by sending and receiving messages written in Extensible Markup Language (XML). The XML messages you send must follow the rules set by the Message Document Type Definitions (DTDs) described in the [“Message Document Type Definitions” section on page 2-3](#).

The two different kinds of messages which can be sent to the Cisco CallManager Extension Mobility API must be sent to distinct URLs. The default URLs are:

- Login/Logout requests: `http://<machine-address>/LoginService/login.asp`
- System queries: `http://<machine-address>/LoginService/query.asp`

Since there are two types of requests, Login and Logout, and there are two different types of queries, Device-User and User-Devices, the Cisco CallManager Extension Mobility API supports four distinct types of messages.

The application sends authentication information, including an Application ID and an Application Certificate, at the start of message.

The only type of certificate which is currently supported is a password. All messages must include a valid appID and appPassword, or they will not be processed.

For examples of legal Cisco CallManager Extension Mobility messages, see the [“Message Examples” section on page 2-5](#).

## Login Requests

Login requests are the cornerstone of this service, and currently they are the most flexible and complex message type. The information required to process a login request must include the device which is to be logged into and the userid of the user logging into that device. If a device profile other than the default Device Profile which has been associated with the user is to be used, that profile name can be specified optionally. If the system is to log the user out automatically after a particular period of time, that can also be specified optionally.

## Logout Requests

To Log out, you only need to provide the device name in the message.

## Device-User Queries

A Device-User query is a query wherein the application specifies a list of one or more devices, and the system returns the userid of the user currently logged on to each device.

## User-Devices Queries

A User-Devices query is a query in which the application specifies a list of one or more users, and the system returns the list of devices for each user that the particular user is currently logged into.

# Securing Cisco CallManager Extension Mobility Messages

There is a security mechanism which can be taken advantage of in order to prevent the appID and password from being sent in cleartext over the HTTP connection in the request or query. This mechanism uses a public-key encryption system which is relatively safe, but will slow down response time and increase processor load as the requests must be decrypted before being serviced.

In order to take advantage of this mechanism, these two steps must be taken:

1. The request or query XML string must be encrypted.
2. The request or query must be POSTed to loginSecure.asp or querySecure.asp rather than login.asp or query.asp, respectively.

To encrypt the XML string, use either the EMEncoder.dll or EMEncoder.jar, both of which may be obtained from the Cisco CallManager server which supplies the Cisco CallManager Extension Mobility service at:

http://<Cisco CallManager ip-address>/LoginService/Tools/EMEncoder.dll

and

http://<Cisco CallManager ip-address>/LoginService/Tools/EMEncoder.jar

Use the .dll or .jar to instantiate the EMEncoder object, which has a single method, encode(String). This method takes the XML string as input and returns the encrypted string, which is to be posted to the appropriate Cisco CallManager Extension Mobility service Active Server Page, either loginSecure.asp or querySecure.asp, exactly as its unencrypted counterpart would be posted to login.asp or query.asp.

The response is still in cleartext.

## Message Document Type Definitions

A Message Document Type Definition (DTD) is an XML list that specifies precisely which elements can appear in a request, query, or response document. It also specifies the contents and attributes of the elements.

You communicate between your login application and the Cisco CallManager Extension Mobility Login Service by sending and receiving XML documents. These XML documents must follow the rules set by the Message DTDs.

For more details about messages, see the [“Messages” section on page 2-2](#). For examples of how the Message DTDs are used, see the [“Message Examples” section on page 2-5](#).

## The Request DTD

The Request DTD defines the login and logout messages your application can send to the Cisco CallManager Extension Mobility Login Service.

```
<!-- login requests DTD -->
<!ELEMENT request (appInfo, (login | logout))>
<!ELEMENT appInfo (appID, appCertificate)>
<!ELEMENT appID      (#PCDATA)>
<!ELEMENT appCertificate (#PCDATA)>
<!ELEMENT login (deviceName, userID, deviceProfile?, exclusiveDuration?)>
<!ELEMENT logout (deviceName)>
<!ELEMENT deviceName  (#PCDATA)>
<!ELEMENT userID      (#PCDATA)>
<!ELEMENT deviceProfile (#PCDATA)>
<!ELEMENT exclusiveDuration (time | indefinite)>
<!ELEMENT time        (#PCDATA)>
<!ELEMENT indefinite EMPTY>
```

## The Login or Logout Response DTD

Login or Logout Response DTD defines the messages your application receives from the Cisco CallManager Extension Mobility Login Service when it sends a login or logout request message.

```
<!-- login response DTD -->
<!ELEMENT response (success | failure)>
<!ELEMENT success EMPTY>
<!ELEMENT failure (error)>
<!ELEMENT error (#PCDATA)>
<!ATTLIST error
  code NMTOKEN #REQUIRED>
```

## The Query DTD

The Query DTD defines the Device-User and User-Devices messages your application sends the Cisco CallManager Extension Mobility service to find out what user is logged into a device or what devices users are logged into.

```
<!-- login query DTD -->
<!ELEMENT query (appInfo, (deviceUserQuery | userDevicesQuery))>
```

```

<!ELEMENT appInfo (appID, appCertificate)>
<!ELEMENT appID      (#PCDATA)>
<!ELEMENT appCertificate (#PCDATA)>
<!ELEMENT deviceUserQuery (deviceName+)>
<!ELEMENT userDevicesQuery (userID+)>
<!ELEMENT deviceName  (#PCDATA)>
<!ELEMENT userID      (#PCDATA)>

```

## The Query Response DTD

The Query Response DTD defines the messages your application receives from the Cisco CallManager Extension Mobility service when it sends the service a Device-User or User-Devices query.

```

<!-- login query results DTD -->
<!ELEMENT response (deviceUserResults | userDevicesResults | failure)>
<!ELEMENT deviceUserResults (device+)>
<!ELEMENT userDevicesResults (user+)>
<!ELEMENT device (userID | none | doesNotExist)>
<!ATTLIST device
      name NMTOKEN #REQUIRED>
<!ELEMENT user (deviceName+ | none | doesNotExist)>
<!ATTLIST user
      id NMTOKEN #REQUIRED>
<!ELEMENT userID (#PCDATA)>
<!ELEMENT deviceName (#PCDATA)>
<!ELEMENT none EMPTY>
<!ELEMENT doesNotExist EMPTY>
<!ELEMENT failure (errorMessage)>
<!ELEMENT errorMessage (#PCDATA)>

```

## Message Examples

This section provides examples of various types of messages to aid in understanding how to use the message DTDs to communicate between your application and the Cisco CallManager Extension Mobility service. Table 2-1 lists each example's type, a description of what the example message means, and a reference to that example. For more details about messages, see the [“Messages” section on page 2-2](#). For details about the DTDs, see the [“Message Document Type Definitions” section on page 2-3](#).

Table 2-1 Message Examples

Message Example Type	Description	Example Reference
Login Request	The 7960LoginApp application requests that user rknotts be logged into device SEP003094C25B15	<a href="#">Example 2-1 on page 2-6</a>
Login Request	The WebLoginApp application makes a login request specifying the RyanTravelPhone profile and limiting the login time to 60 minutes.	<a href="#">Example 2-2 on page 2-7</a>
Login Request	WebLoginApp requests that user rknotts be logged in to the specified device for an unlimited duration.	<a href="#">Example 2-3 on page 2-7</a>
Logout Request	The 7960LoginApp application requests that the current user be logged out of device SEP003094C25B15	<a href="#">Example 2-4 on page 2-7</a>
Request Response	Response of Success to a login or logout request	<a href="#">Example 2-5 on page 2-8</a>
Request Response	Failure response with error indicating incorrect appID or password	<a href="#">Example 2-6 on page 2-8</a>
Device-User Query	Querying what user is logged into device SEP003094C25B15	<a href="#">Example 2-7 on page 2-8</a>
User-Devices Query	Querying which devices are user rknotts and fwragge logged into	<a href="#">Example 2-8 on page 2-9</a>
Device-User Response	Response saying that rknotts is the user logged into device SEP003094C25B15	<a href="#">Example 2-9 on page 2-9</a>
User-Devices Response	Response saying that rknotts is logged into devices SEP003094C25B15 and SEP003094C25B49 and fwragge is logged into device SEP003094C25B46	<a href="#">Example 2-10 on page 2-9</a>

## Request Examples

Request examples demonstrate three different login requests and one logout request. The login requests show a simple login message and two that specify options like using a particular device profile or setting a login duration.

### Example 2-1 A Simple Login Request

```
<request>
  <appInfo>
    <appID>7960LoginApp</appID>
    <appCertificate>password</appCertificate>
  </appInfo>
  <login>
    <deviceName>SEP003094C25B15</deviceName>
    <userID>rknotts</userID>
```

```
</login>
</request>
```

**Example 2-2 Login Request Specifying a Profile and a Time Restriction**

```
<request>
  <appInfo>
    <appID>WebLoginApp</appID>
    <appCertificate>password</appCertificate>
  </appInfo>
  <login>
    <deviceName>SEP003094C25B15</deviceName>
    <userID>rknotts</userID>
    <deviceProfile>RyanTravelPhone</deviceProfile>
    <exclusiveDuration>
      <time>60</time>
    </exclusiveDuration>
  </login>
</request>
```

**Example 2-3 Login Request Specifying an Unlimited Duration**

```
<request>
  <appInfo>
    <appID>WebLoginApp</appID>
    <appCertificate>password</appCertificate>
  </appInfo>
  <login>
    <deviceName>SEP003094C25B15</deviceName>
    <userID>rknotts</userID>
    <exclusiveDuration>
      <indefinite/>
    </exclusiveDuration>
  </login>
</request>
```

**Example 2-4 Logout Request**

```
<request>
  <appInfo>
    <appID>7960LoginApp</appID>
```

```

    <appCertificate>password</appCertificate>
  </appInfo>
  <logout>
    <deviceName>SEP003094C25B15</deviceName>
  </logout>
</request>

```

## Request Response Examples

The request response examples show a success message (for either login or logout) and a failure message indicating the type of error that the login or logout attempt generated.

### **Example 2-5 Success Response**

```

<response>
  <success/>
</response>

```

### **Example 2-6 Failure Response**

```

<response>
  <failure>
    <error code="3">Could not authenticate 'appid'</error>
  </failure>
</response>

```

## Query Examples

Query examples show a typical Device-User Query message and a typical User-Devices Query message sent by an application to the Cisco CallManager Extension Mobility Service.

### **Example 2-7 A Device-User Query**

```

<query>
  <appInfo>
    <appID>applicationName</appID>
    <appCertificate>password</appCertificate>
  </appInfo>
  <deviceUserQuery>
    <deviceName>SEP003094C25B15</deviceName>
  </deviceUserQuery>
</query>

```



**Example 2-8 A User-Devices Query**

```

<query>
  <appInfo>
    <appID>applicationName</appID>
    <appCertificate>password</appCertificate>
  </appInfo>
  <userDevicesQuery>
    <userID>rknotts</userID>
    <userID>fwragge</userID>
  </userDevicesQuery>
</query>

```

## Query Response Examples

Query Response examples show messages sent to the login application by the Cisco CallManager Extension Mobility Service after the login application has sent a Device-User query message or a User-Devices query message.

**Example 2-9 A Device-User Response**

```

<results>
  <deviceUserResults>
    <device name="SEP003094C25B15">
      <userID>rknotts</userID>
    </device>
  <deviceUserResults>
</results>
<results>

```

**Example 2-10 A User-Devices Response**

```

<userDevicesResults>
  <user id="rknotts">
    <deviceName>SEP003094C25B15</deviceName>
    <deviceName>SEP003094C25B49</deviceName>
  </user>
  <user id="fwragge">
    <deviceName>SEP003094C249A6</deviceName>
  </user>
</userDeviceResults>
</results>

```

# Login Service Error Codes

Table 2-2 shows the current error codes that the Cisco CallManager Extension Mobility Login Service returns and describes what each error code means.

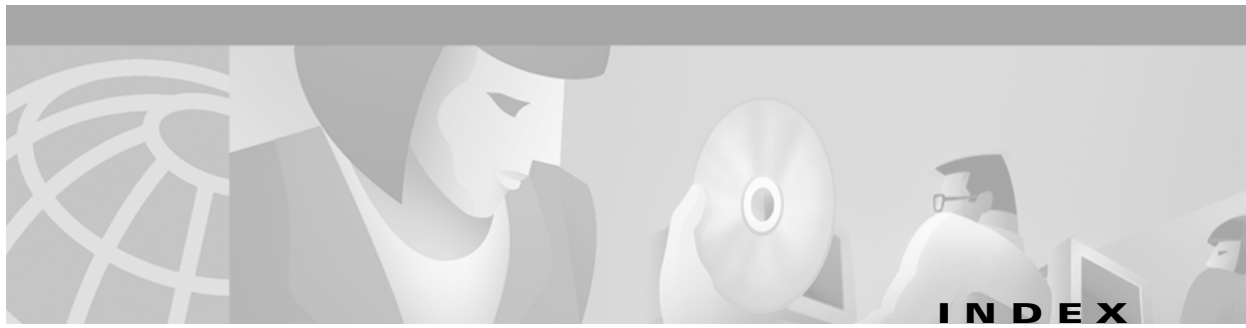
**Table 2-2 Cisco CallManager Extension Mobility Login Service Error Codes**

Error Code	Description
0	Unknown Error
1	XML Validation Error: The request or query was incorrectly formed, and cannot be properly processed
2	Authentication Error: An error occurred in the authentication process, and the validity of the appID and appPassword submitted cannot be confirmed
3	Invalid Authentication: The appID and/or appPassword provided are incorrect
4	Policy Validation Error: Some generic issue regarding determination if the request is allowed or not
5	Request Denied: The request has been denied (failed Policy Validation) for one or more reasons
6	Database Error: The Extension Mobility Service received an error while trying to communicate with the database
7	AutoLogout Setup Error: Could not communicate with the AutoLogout service
8	Query Type Unknown: could not determine whether the query is Device-User or User-Devices
9	DirUser Creation Error: Directory integration error
10	Proxy Authentication Not Allowed: the appID specified does not have rights to login or logout other users
11	Device Does Not Exist: The specified device for login or logout does not exist in the system
12	Device Profile Does Not Exist: Either a profile was specified which does not exist or there is no Logout Device Profile configured for the specified user
13-17	Various Pipe Errors: Could not communicate with the AutoLogout service
18	Device Already Logged In: Could not log in to the specified device because there is already a user logged in
19	Device Not Logged In: Could not log out of the specified device because there is no user logged in
20	Get Device Hoteling Flag Error: Could not determine whether the device allows Cisco CallManager Extension Mobility (also called “hoteling”) or not
21	Get Device Hoteling Status Error: Could not determine whether there is a user currently logged in or not

**Table 2-2 Cisco CallManager Extension Mobility Login Service Error Codes**

<b>Error Code</b>	<b>Description</b>
22	Device Does Not Allow Extension Mobility: The device specified is not configured for Extension Mobility
23	User Does Not Exist: The userid entered for login does not exist in the system
24	System Disabled: The Extension Mobility service has been disabled from the service parameters
25	User Already Logged In Elsewhere: The login is denied because the specified user is already logged into a different device





---

## A

architecture [1-2](#)

---

## C

configuration [2-1](#)

---

## D

device profile

    auto-generated [1-5](#)

    User Device Profile [1-5](#)

device profiles [1-5](#)

    Logout Device Profile [1-5](#)

---

## L

login service [1-4](#)

    authentication [1-6](#)

    automatic logout [1-6](#)

    error codes [2-10](#)

    login policy [1-6](#)

---

## M

messages [2-2](#)

    queries [2-3](#)

        device-user [2-3](#)

        query DTD [2-4](#)

        query examples [2-8](#)

        response DTD [2-5](#)

        response examples [2-9](#)

    user-devices [2-3](#)

requests [2-2](#)

    login [2-2](#)

    logout [2-2](#)

    request DTD [2-4](#)

    request examples [2-6](#)

    response DTD [2-4](#)

    response examples [2-8](#)

---

## S

security [2-3](#)

