API Overview

Links to Other API pages: Cisco_Unity_Connection_APIs

API Overview

Links to Other API pages: Cisco_Unity_Connection_APIs

Introduction

Cisco Unity Connection Provisioning Interface (CUPI) is a provisioning API for Cisco Unity Connection that has been designed to be stable and simple to use. It is based on leading industry standards for web-based API development, and provides access to the most commonly provisioned data on Connection systems (users, contacts, distribution lists, and call handlers).

Note   With Cisco Unity Connection 9.1(1), the single sign-on feature is enabled for all the Connection Rest APIs. For more information, see the "Single Sign-On in Cisco Unity Connection" chapter in *Security Guide for Cisco Unity Connection 9.x* available at http://www.cisco.com/en/US/docs/voice_ip_comm/connection/9x/security/guide/9xcucsec061.htm

Benefits

By using CUPI, you can securely do the following:

- Create, read, update, and delete users and user configurations
- Reset passwords
- Create, read, and update distribution lists
- Create, read, update, and delete call handlers
- Create, read, update, and delete contacts

> **Note** All the above functions associated with CUPI API support both the IPv4 and IPv6 addresses. However, the IPv6 address works only when Connection platform is configured in Dual (IPv4/IPv6) mode.

For more information see the chapter "Adding or Changing the IPv6 Addresses of Cisco Unity Connection 8.5 and Later Servers" at the following link https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/connection/9x/upgrade/guide/9xcucrugx/9xcucrug051.html

Technical Details

Previous provisioning APIs required knowledge of the underlying database, and were vulnerable to changes in the database schema. CUPI provides a layer over the database to make the interface more stable.

CUPI is standards based. CUPI was developed by using the latest advances in web-based interfaces. It is a REST interface that standardizes operations such as add, delete, and modify. The XML comes with standard XML schema definitions that are annotated with information about what is in them.

Other product groups in VTG are developing REST-based interfaces, and as the interfaces converge this positions CUPI well to fit in with them. REST interfaces also work well with Web 2.0 applications.

As a web-based interface, CUPI is independent of operating system and programming language and does not require any client libraries to use.

> **Note** It is recommended to perform provisioning only on the Publisher server in Active-Active mode and on Subscriber (Acting Primary) in case of cluster failover. The password change and password setting modification for User PIN/Web application should be provisioned on Publisher server in Active-Active mode.

Getting Started

In order to begin developing with the Cisco Unity Connection CUPI API, you need to obtain the following:

Hardware

- Cisco Media Convergence Server (MCS) for Cisco Unity Connection version 8.0 and later
- For detailed hardware requirements, see the Cisco Unity Connection 8.x Supported Platforms List

Software

- Cisco Unity Connection Software Ordering
- Not for Resale Kits (Must be eligible to purchase)
- Communications System Release Kit

Discounts for some of the required hardware and software may be available for participants in the Cisco Technology Developer Program.

We recommend that all developers have an up-to-date Cisco Developer Services support agreement. This provides the developer with access to professional support and assistance for application development

Other CUPI Resources

Additional information about CUPI is also available on the Cisco Developer Network (link to CDN). Note, however, that the documentation here on the DocWiki is the most up-to-date documentation available for CUPI.

To participate in the CUPI forum, see the CUPI forum on CDN.

On the CUPI page on CDN, you will also find links to the CUPI WADL and CUPI XML schema.

Troubleshooting

See the following for information on troubleshooting all Connection APIs:

Troubleshooting (applies to all Connection APIs)