



CTI OS System Manager Guide for Cisco Unified ICM, Release 12.5(1)

First Published: 2020-02-06

Americas 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: 408 527-0883

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 NON-INFRINGEMENT 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.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com go trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned 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. (1721R)

© 1994–2020 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface	ix
Change History	ix
About this Guide	ix
Audience	ix
Related Documents	x
Communications, Services, and Additional Information	x
Field Notice	x
Documentation Feedback	xi
Conventions	xi

CHAPTER 1

Introduction	1
Overview of CTI OS	1
Advantages of CTI OS as Interface to Unified ICM Enterprise	2
Key Benefits of CTI OS for CTI Application Developers	2
System Manager Responsibilities	3
System Requirements	3
Set User Privileges	4

CHAPTER 2

CTI OS Server Installation	5
CTI OS Server Installation Guidelines	5
Upgrade from Previous Version	6
Install CTI OS Server	6
Uninstalling CTI OS Server	10
Determine Version Number of Installed Files	10

CHAPTER 3

CTI Toolkit Desktop Client Installation	13
--	-----------

CTI Toolkit Desktop Client 13

Install Cisco CTI Toolkit Desktop Client Component 14

 Localization 15

 Installed Files 15

Uninstall CTI Toolkit 17

Determine Version Number of Installed CTI Toolkit Files 17

Unified CM Intercept Configuration Requirement 18

Configure Supervisory Assistance Features 18

CHAPTER 4

CTI OS Silent Monitor Installation and Configuration 19

Overview of CTI OS 19

 Advantages of CTI OS as Interface to Unified ICM Enterprise 20

 Key Benefits of CTI OS for CTI Application Developers 20

System Manager Responsibilities 21

System Requirements 21

 Set User Privileges 22

Silent monitoring 22

 Silent Monitor Differences Between Unified CM and CTI OS 22

 Unified CM-Based Silent Monitoring 23

 Unified CM Silent Monitor Advantages 24

 Unified Communications Manager Silent Monitor Limitations and Restrictions 24

 CTI OS-Based Silent Monitoring 25

 Network Topology for Silent Monitoring 25

 Unified CM-Based Silent Monitoring 25

 CTI OS-Based Silent Monitoring 26

 Silent Monitoring and Mobile Agent Topology 27

 Calculation of Additional Needed Bandwidth 27

CHAPTER 5

CTI OS Component Installation 29

Silent Installation of CTI OS Components 29

 Create a Response File 30

 Run CTI OS Silent Install on Other Machines 30

Uninstall Components 31

Recover from Failed Installation of CTI OS 31

CHAPTER 6	Unified Communications Manager-Based Silent Monitor Configuration	33
	Silent Monitor Service Installation and Configuration	33
	Silent Monitor Service Overview	33
	CTI OS Connections	34
	How desktops Connect to Silent Monitor Services	34
	Configure ESXi Server	34
	Configure LAN Switch	35
	Upgrade Silent Monitor Service	36
	Run Silent Monitor Service Installer	36
	Sign a Silent Monitor Server Certificate Request with Self-Signed CA	38
	Sign a Silent Monitor Service Certificate Request with Third-Party CA	39
	Additional Configuration Steps	39
	Rerunning CTI OS Server Setup	40
	Additional Configuration for Mobile Agent Environments	40
	Silent Monitor Service Clusters	41
	Installation of Silent Monitor Service with Windows Firewall Service Enabled	41
	Harden Silent Monitor Server Security	41
	Add Silent Monitor Service to Windows Firewall Exceptions	42
	Silent Monitor Service Deployments	42
	Unified CCE Deployment	43
	Mobile Agent Using Analog/PSTN Phone	44
	Mobile Agents IP Phones Topology	44
	Mobile Agent with IP Phone	45

CHAPTER 7	CTI OS Security	47
	CTI OS Security Certificate Configuration	47
	CTI OS Security Setup Programs	47
	Sign CTI Toolkit Desktop Client Certificate Request with Self-Signed CA	48
	Sign CTI OS Server Certificate Request with Self-Signed CA	49
	Sign CTI Toolkit Desktop Client Certificate Request with Third-Party CA	50
	Sign CTI OS Server Certificate Request with Third-Party CA	50
	CTI OS Security Passwords	51
	CTI OS Security Registry Keys	51

- Mode Security Monitoring 53
- Security Compatibility 53
 - Wire Level Encryption 54
 - Authentication Mechanism 54

CHAPTER 8 **CTI OS Configuration 55**

- Use Windows Registry Editor 55
 - Silent Monitor Type Configuration for CTI OS 57
- Virtual Desktop Infrastructure 57
 - CTI OS Desktop Installations on VDI Agent Desktops 57
 - Prerequisites 57
 - Install CTI OS Desktop on VDI Agent 57
 - Notes and Restrictions 58
- CTI Driver Key 58
- EMS Tracing Values 59
- Server Registry Key 60
 - Agent Registry Key 61
 - ReasonCodes Registry Key 63
 - WrapupStrings Registry Key 64
 - CallObject Registry Key 65
 - Connections Registry Key 65
 - Device Registry Key 67
 - Peers Registry Key 67
 - Peripherals Registry Key 67
 - SkillGroup Registry Key 68
 - Supervisor Registry Key 68
 - ThreadPoolSize Registry Key 69
 - TimerService Registry Key 69
- MainScreen Registry Key 70
- Unified CCE Silent Monitor Configuration 70
- ConnectionProfiles Registry Key 71
 - SilentMonitorService Subkey 74
- Configuration of Additional Connection Profiles 75
 - Creation of Second Profile 75

Two Profiles for Server- and Desktop-Based Silent Monitoring Scenario	76
Call Appearance Grid Configuration	77
Configure Automatic Call Appearance Grid	79
Customize Agent Statistics Grid Configuration	82
Automatic Skill Group Statistics Grid Configuration	83
Configure Additional Peripherals	85
Quality of Service/Type of Service	85
Basic Configuration	86
Important Additional Configuration Information	86
Caveats	86

CHAPTER 9

Startup, Shutdown, and Failover	89
Unified CCE Service Control	89
CTI OS Failover	90
Failover of CTI OS Related Components	90
IP Phones	90
Switches	90
Peripheral Gateway	90
CTI Server Failure	90
CTI OS Server Failure	91

CHAPTER 10

Peripheral-Specific Support	93
TDM peripherals	93
General Unified ICM Support	93
Peripheral-Specific Terminology	94
Unified ICM Feature Limitations	95
CTI OS Support	95
Call Events	95
Client Control Requests	96
Peripheral-Specific Limitations and Differences	97
Aspect Contact Server	98
Avaya DEFINITY ECS	98
Unified CCE System PG	116
UCCE Error Codes	118

Avaya Aura CC (Symposium) 133
 Agent States 139

CHAPTER 11

Cisco Unified Mobile Agent 141
 Log in to CTI OS Agent Desktop 141
 Verify Login 142
 Enable Ready State 142
 Transfer a Call 142

APPENDIX A

Ethernet Card Testing 145
 Ethernet Cards for Silent Monitor 145
 Test Procedure 145
 Prepare Test Target 146
 Prepare Packet Generator Host 147
 Executing a Test 148



Preface

- [Change History](#), on page ix
- [About this Guide](#), on page ix
- [Audience](#), on page ix
- [Related Documents](#), on page x
- [Communications, Services, and Additional Information](#), on page x
- [Field Notice](#), on page x
- [Documentation Feedback](#), on page xi
- [Conventions](#), on page xi

Change History

Change	See	Date
Initial Release of Document for Release 12.5(1)		
Updated CTI OS client and server version to 12.5	Across the document.	
Updated Windows Server version to 2016	Across the document.	

About this Guide

This guide describes how to install, configure, and run the Cisco CTI Object Server (CTI OS) product.

Audience

This document is intended for system administrators and other personnel who are responsible for installing and maintaining CTI OS and its associated components.

Related Documents

Subject	Link
Related documentation includes the documentation sets for Cisco CTI Object Server (CTI OS), Cisco Unified Contact Center Management Portal, Cisco Unified Customer Voice Portal (Unified CVP), Cisco Unified IP IVR, and Cisco Unified Intelligence Center	To see all related documentation sets, go to https://www.cisco.com/cisco/web/psa/default.html?mode=prod . Select Products > Customer Collaboration > Contact Center .
Cisco Unified Communications Manager documentation set	Go to https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/tsd-products-support-general-information.html

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at [Cisco Profile Manager](#).
- To get the business impact you're looking for with the technologies that matter, visit [Cisco Services](#).
- To submit a service request, visit [Cisco Support](#).
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit [Cisco Marketplace](#).
- To obtain general networking, training, and certification titles, visit [Cisco Press](#).
- To find warranty information for a specific product or product family, access [Cisco Warranty Finder](#).

Cisco Bug Search Tool

[Cisco Bug Search Tool](#) (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Field Notice

Cisco publishes Field Notices to notify customers and partners about significant issues in Cisco products that typically require an upgrade, workaround, or other user action. For more information, see *Product Field Notice Summary* at <https://www.cisco.com/c/en/us/support/web/tsd-products-field-notice-summary.html>.

You can create custom subscriptions for Cisco products, series, or software to receive email alerts or consume RSS feeds when new announcements are released for the following notices:

- Cisco Security Advisories
- Field Notices
- End of Sale or Support Announcements

- Software Updates
- Updates to Known Bugs

For more information on creating custom subscriptions, see *My Notifications* at <https://cway.cisco.com/mynotifications>.

Documentation Feedback

To provide comments about this document, send an email message to the following address: contactcenterproducts_docfeedback@cisco.com.

We appreciate your comments.

Conventions

This document uses the following conventions:

Convention	Description
boldface font	<p>Boldface font is used to indicate commands, such as user entries, keys, buttons, folder names, and submenu names.</p> <p>For example:</p> <ul style="list-style-type: none"> • Choose Edit > Find. • Click Finish.
<i>italic font</i>	<p>Italic font is used to indicate the following:</p> <ul style="list-style-type: none"> • To introduce a new term. Example: A <i>skill group</i> is a collection of agents who share similar skills. • A syntax value that the user must replace. Example: IF (<i>condition, true-value, false-value</i>) • A book title. Example: See the <i>Cisco Unified Contact Center Enterprise Installation and Upgrade Guide</i>.
window font	<p>Window font, such as Courier, is used for the following:</p> <ul style="list-style-type: none"> • Text as it appears in code or that the window displays. Example: <pre><html><title>Cisco Systems, Inc. </title></html></pre>
< >	<p>Angle brackets are used to indicate the following:</p> <ul style="list-style-type: none"> • For arguments where the context does not allow italic, such as ASCII output. • A character string that the user enters but that does not appear on the window such as a password.



CHAPTER 1

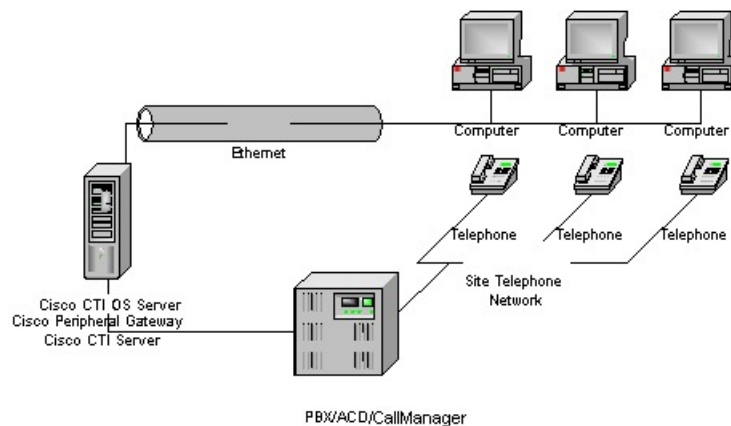
Introduction

- [Overview of CTI OS, on page 1](#)
- [System Manager Responsibilities, on page 3](#)
- [System Requirements, on page 3](#)

Overview of CTI OS

CTI OS combines a powerful, feature-rich server and an object-oriented software development toolkit to enable rapid development and deployment of complex CTI applications. Together, the Cisco CTI Server Interface, CTI OS Server, and CTI OS Client Interface Library (CIL) create a high performance, scalable, fault-tolerant three-tiered CTI architecture, as illustrated in following figure.

Figure 1: CTI OS Three-Tiered Architecture Topology



The CTI OS application architecture employs three tiers:

- The CIL is the first tier, providing an application-level interface to developers.
- The CTI OS Server is the second tier, providing the bulk of the event and request processing and enabling the object services of the CTI OS system.

- The Cisco CTI Server is the third tier, providing the event source and the back-end handling of telephony requests.

Advantages of CTI OS as Interface to Unified ICM Enterprise

CTI OS brings several major advances to developing custom CTI integration solutions. The CIL provides an object-oriented and event-driven Application Programming Interface (API), while the CTI OS Server does the *heavy-lifting* of the CTI integration: updating call context information, determining which buttons to enable on softphones, providing easy access to supervisor features, and automatically recovering from failover scenarios.

The key advantages of CTI OS include:

- **Rapid integration.** Developing CTI applications with CTI OS is easier and faster than any previously available Cisco CTI integration platform. The same object-oriented interface is used across programming languages, enabling rapid integrations in C++, Visual Basic, .NET, Java, or any Microsoft COM-compliant container environment.



Note The inclusion of the .NET toolkit allows for custom applications written in C#, VB.NET, or any other CLR-compliant language. By starting with the code for the .NET sample, the CTI Toolkit Combo Desktop developers can quickly customize the code without having to start from scratch.

CTI OS enables developers to create a screen-pop application in as little as five minutes. The only custom-development effort required is within the homegrown application to which you add CTI.

- **Complex solutions made simple.** CTI OS enables complex server-to-server integrations and multiple agent monitoring-type applications. The CIL provides a single object-oriented interface that you can use in two modes: agent mode and monitor mode. For more information about these two modes, see CTI OS Developer Guide for Cisco Unified ICM at: https://www.cisco.com/en/US/products/sw/custcosw/ps14/products_programming_reference_guides_list.html.
- **Fault tolerant.** CTI OS is built upon the Unified ICM Node Manager fault-tolerance platform, which automatically detects process failure and restarts the process, enabling work to continue. Upon recovery from a failure, CTI OS initiates a complete, system-wide snapshot of all agents, calls, and supervisors and propagates updates to all client-side objects.

Key Benefits of CTI OS for CTI Application Developers

The CTI OS CIL provides programmers with the tools required to rapidly develop high-quality CTI-enabled applications, taking advantage of the rich features of the CTI OS Server. Every feature of CTI OS was designed with ease of integration in mind, to remove the traditional barriers to entry for CTI integrations:

- **Object-oriented interactions.** CTI OS provides an object-oriented CTI interface by defining objects for all call center interactions. Programmers interact directly with Session, Agent, SkillGroup, and Call objects to perform all functions. CIL objects are thin proxies for the server-side objects, where all the 'heavy-lifting' is done. The Session object manages all objects within the CIL. A UniqueObjectID identifies each object. Programmers can access an object by its UniqueObjectID or by iterating through the object collections.

- **Connection and session management.** The CTI OS CIL provides out-of-the-box connection and session management with the CTI OS Server, hiding all of the details of the TCP/IP sockets connection. The CIL also provides out-of-the-box failover recovery. Upon recovery from a failure, the CIL automatically reconnects to another CTI OS Server (or reconnects to the same CTI OS Server after restart), reestablishes the session, and recovers all objects for that session.
- **All parameters are key-value pairs.** The CTI OS CIL provides helper classes to treat all event and request parameters as simply a set of key-value pairs. All properties on the CTI OS objects are accessible by name via a simple `Value = GetValue("key")` mechanism. Client programmers can add values of any type to the CTI OS Arguments structure using the enumerated CTI OS keywords or their own string keywords (for example, `AddItem["DialedNumber", "1234"]`). This provides for future enhancement of the interface without requiring any changes to the method signatures.
- **Simple event subscription model.** The CTI OS CIL implements a publisher-subscriber design pattern to enable easy subscription to event interfaces. Programmers can subscribe to the event interface that suits their needs, or use the AllInOne interface to subscribe to all events. Subclassable event adapter classes enable programmers to subscribe to event interfaces and only add minimal custom code for the events they use, and no code at all for events they do not use.

System Manager Responsibilities

The remainder of this document provides step-by-step procedures for the tasks a system manager must perform to set up and configure CTI OS. These tasks include:

- Installing CTI OS Server.
- Installing CTI Toolkit Agent Desktop, Supervisor Desktop, Tools, Documentation, Win32 SDK, Java SDK, and .NET SDK.
- Enabling CTI OS security.
- Using the Windows Registry Editor (`regedit.exe`) to configure the required CTI OS registry keys.
- Starting CTI OS and its associated processes from Unified CCE Service Control.



Note You *must* have administrator privileges to perform the procedures discussed in this manual.

System Requirements

See the *Solution Design Guide for Cisco Unified Contact Center Enterprise* at http://www.cisco.com/en/US/products/sw/custcosw/ps1844/products_implementation_design_guides_list.html

For more information on system requirements, see the *Virtualization for Unified Contact Center Enterprise* at http://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/virtualization-unified-contact-center-enterprise.html

Set User Privileges

On supported Windows client machines, users must have privileges that enables them to run legacy applications and have read/write access to the Cisco registry keys that the desktop applications use. To set user privileges to enable users to run CTI OS Agent Desktop and CTI OS Supervisor Desktop, an administrator must perform the following steps.

Procedure

- Step 1** On the Microsoft Windows Start Menu, select **Start > Run**.
- Step 2** Type in **regedt32** and click **OK**.
The Microsoft Windows Registry Editor window appears.
- Step 3** Go to the following registry location:
`HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTI Desktop\Ctios`
- Step 4** Select **Security > Permissions**.
A Permissions dialog box appears.
- Step 5** If you are adding a new user, perform the following steps.
- a) Click **Add**.
A Select Users dialog box appears.
 - b) Select the user to be added from the list in the top half of the Select Users dialog box.
 - c) Click **Add**, then click **OK**.
You return to the Permissions dialog box; the user you just added is now on the list.
- Step 6** Click the user whose privileges you want to set.
- Step 7** Set the Full Control permissions for this user to **Allow**.
- Step 8** Click **Apply**.
- Step 9** Click **OK**.
- Step 10** Exit Registry Editor.
-



CHAPTER 2

CTI OS Server Installation

This chapter lists some guidelines to consider when you install the CTI OS Server and provides procedures for these tasks.



Caution

You cannot run the installer remotely. Mount the installer ISO file only to a local machine. Various errors can occur during installation over the network. Keep in mind that for installation of major releases, there is no way to roll the installation back to the previous release if the installation or upgrade fails part way through.

- [CTI OS Server Installation Guidelines, on page 5](#)
- [Upgrade from Previous Version, on page 6](#)
- [Install CTI OS Server, on page 6](#)
- [Uninstalling CTI OS Server, on page 10](#)
- [Determine Version Number of Installed Files, on page 10](#)

CTI OS Server Installation Guidelines

Following are some guidelines to consider when you install CTI OS Server:

- CTI OS is typically installed in a redundant configuration. Two CTI OS Servers installed on separate systems work in parallel to provide redundancy. Installing only one CTI OS Server prevents failover recovery by client systems.
- CTI OS must be colocated on the same box as the PG/CG.
- Ensure that your CTI OS system meets the specified requirements. See the Contact Center Enterprise Compatibility Matrix for more information.



Caution

You cannot run the installer remotely. Mount the installer ISO file only to a local machine. Various errors can occur during installation over the network. Keep in mind that for installation of major releases, there is no way to roll the installation back to the previous release if the installation or upgrade fails part way through.

Upgrade from Previous Version



Note If you are upgrading from a CTI OS Server Release 12.0, you need not uninstall CTI OS Server before you install CTI OS Server Release 12.5(1).

While installing CTI OS Server 12.5(1), the listen ports for CTI OS Server and silent monitor are registered as firewall exceptions.

Silent upgrade is not supported for CTI OS Security Server and Client.

Procedure

- Step 1** Run the **Setup.exe**.
- A warning message appears indicating that the Cisco Contact Center SNMP Management Service is stopped before the CTI OS Server begins to install.
- Click **Yes** to continue.
- Step 2** In the Software License Agreement dialog box, click **Yes**.
- Step 3** In the CTI OS Server Installer dialog box, leave the **Location** field blank and click **Next**.
- Step 4** The CTI OS Instances dialog box is displayed. Click **Upgrade All**.
- Step 5** In case you have a version of CTI OS Server already installed, and you are attempting to install the latest version, the dialog box to confirm the upgrade is displayed. Click **Yes**.
-

Install CTI OS Server

To install a new CTI OS Server, perform the following steps:



Note The CTI OS Server installation procedure includes windows for mobile agents and silent monitor server.

Procedure

- Step 1** From the Server directory on the CD, run **setup.exe**.
- A warning message appears indicating that the Cisco Contact Center SNMP Management Service is stopped before the CTI OS Server begins to install.
- Note** When you run programs from a Windows Server system with User Account Control enabled, Windows needs your permission to continue. Click **Yes** in the **User Account Control** window to run the program.

- Step 2** Click **Yes** to continue.
- Step 3** In the Software License Agreement window, click **Yes**.
- Step 4** In the CTI OS Server Installer window, leave the **Location** field blank and click **Next**.
The **CTI OS Instances** window appears.
The **CTI OS Instances** window allows you to create CTI OS instances and add CTI OS Servers to a configured instance of CTI OS.
- Note** The Add buttons are disabled if you cannot create another CTI OS instance.
- Step 5** Under the CTI OS Instance List, click **Add**. The Add CTIOS Server Instance window is displayed.
- Step 6** Enter an instance name and click **OK**. For example, if you enter an instance called **cisco**, the following window appears:
- Step 7** Click **Add** in the CTI OS Server List.
The **Add CTIOS Server** window appears.
The CTIOS Server Name field is populated with the instance name you provided, followed by the next available index for a CTI OS Server. If a CTI OS Server has been deleted, the CTIOS Server Name is populated with the index that was deleted.
- Step 8** If you are installing CTI OS Server for the first time, an **Enter Desktop Drive** window appears. Accept the default installation drive or choose another drive from the **Drive** drop-down list.
- Step 9** Click **OK**.
The **CTI Server Information** window appears.
The Instance Name and CTIOS Server Name is already populated.
- Step 10** For secured connection with the CTI server, check the **Enable Secure Connection** check box.
- Note** For secured connection, you must provide the secure port number of the CTI Server.
If the Enable Secure Connection check box is not checked, the connection established between the CTIOS Server and the CTI Server is non-secured. In this case, you must provide the non-secure port number of the CTI Server.
Before you enable secured connection between the components, ensure to complete the security certificate management process.
For more information, see the *Security Guide for Cisco Unified ICM/Contact Center Enterprise* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-and-configuration-guides-list.html>.
- Step 11** Enter the **Name** or **IP Address** and the **Port Number** for your CTI Server.
If the peripheral is configured for a previous CTI OS Server, the **Name or IP Address** field pre-populates with the name of that CTI OS Server.
- Step 12** Click **Next**.
The **Peripheral Identifier** window appears.
The Peripheral Type field is pre-populated with the peripheral type if it is already configured for a previous CTI OS Server.

Note When you configure multiple CTI OS Servers to use a single CTI Server, every CTI OS Server configured after the first CTI OS Server has the same configuration as of the first CTI OS Server.

Step 13 If the peripheral has not been configured for a previous CTI OS Server, specify the following information:

- A **Logical Name** for your peripheral. The name can be any valid logical name that uniquely identifies your peripheral.

Note The Login By and Enable Mobile Agent group boxes are enabled only for UCCE peripheral types (UCCE System and UCCEHosted Edition).

In the Login By box, you can choose between signing in by Agent ID or by Login Name. The Login By setting determines how the CTI Toolkit Agent and Supervisor Desktops allow Login and Chat request (either Agent ID or Login Name). This setting does not affect other CTI applications. CTI OS Server itself can service Login requests both ways (by Agent ID and Login Name) for UCCE.

- In the **Peripheral ID** field, enter the identifier of the switch that your phone is connected to.
- From the **Peripheral Type** drop-down list, choose the switch that your phone is connected to.
- Check the **Enable Mobile Agent** check box to activate this option.
- Select the **Login By** option.
- The **Mobile agent mode** drop-down lists the following options. Choose one:
 - **Agent chooses**—Agent chooses the mode.
 - **Call by call**—The agent's remote phone is dialed for each individual call.
 - **Nailed connection**—The agent is called once upon signing in and remains connected.

Note You can specify information for only one peripheral during CTI OS Server setup. To configure more peripherals, follow the procedure in the section [Configure Additional Peripherals, on page 85](#).

Step 14 Click **Next**.

The **Connection Information** window appears.

Enter the port number and the heartbeat information for your CTI OS Server instance.

Note For all peripheral types, accept the default Listen Port value of 42028. For subsequent instances, use any available port.

Important Ensure that you configure the CTI OS client that connects to the CTI OS Server with the same port that you selected while installing the CTI OS client.

Step 15 Click **Next**.

The **Statistics Information** window appears.

Note

- Enabling CAD Agent disables the agent statistics polling interval from the CTI OS Server. CAD agents receive only Skillgroup statistics from CTI OS Server.
- After performing an **Upgrade All**, rerun setup to access this window and reconfigure the server for appropriate statistical information.

Step 16 Enter the default polling interval for Skillgroup statistics (in seconds).

Note Because Quality of Service (QoS) enablement and statistics enablement are mutually exclusive, enabling QoS zeros disables all the information relating to statistics.

Step 17 Click **Next**.

The **UCCE Silent Monitor Type** window appears.

Step 18 Choose the type of silent monitor.

- If you choose **Unified CM Based** or **Disabled**, clicking **Next** takes you to the **Peer CTI OS Server** window. Proceed to Step 17.

Note If you want to use Unified CM based type silent monitor, see [Unified Communications Manager-Based Silent Monitor Configuration, on page 33](#).

- If you choose **Disabled**, the CTI OS based silent monitor is configured, but disabled. This sets the registry settings to the following values:

Key	Setting
HKLM\SOFTWARE\Cisco Systems, Inc.\CtiOS_<instance>\CTIOS\EnterpriseDesktopSettings\AllDesktops\IPCCSilentMonitor\Name\Settings\CCMBasedSilentMonitor	0
HKLM\SOFTWARE\Cisco Systems, Inc.\CtiOS_<instance>\CTIOS\EnterpriseDesktopSettings\AllDesktops\Login\ConnectionProfiles\Name\UCCE\IPCCSilentMonitorEnabled	0

- If you choose **CTI OS Based** silent monitor, clicking **Next** takes you to the **Silent Monitor Information** window.

Step 19 On the **Silent Monitor Information** window, enter the following information:

- The port number used by the client to connect to the silent monitor service.
- The set of silent monitor servers that the desktop connects to. The desktop randomly connects to one of the silent monitor servers specified here. If the client is configured to use secure connections, the client attempts to connect to the silent monitor server using a secure connection. If the silent monitor server is configured to use secure connections, then a secure connection is established with the silent monitor server. Otherwise, an unsecured connection is used.

A client uses the same certificates it uses to communicate with CTI OS Server to establish a secure connection to the silent monitor server.

Step 20 Click **Next**.

The **Peer CTIOS Server** window appears.

Step 21 You can configure a CTI OS Peer Server using this window. You can also configure Chat and CTI OS silent monitoring. Enter the Peer CTI OS Server and Port details.

After you click **Finish** and the files are created, the service is registered and entries to the registry are made.

Note You can configure the chat window to beep every time a new message arrives. To do that, set the following registry key to a nonzero value.

HKEY_LOCAL_MACHINE\Cisco Systems, Inc.\CTI Desktop\CtiOs\BeepOnMsgReceived

If the registry key does not exist or if its value is set to zero, the chat window does not beep.

Step 22 The CTI OS Server Security window appears.

If you wish to disable security, click **OK**; otherwise, select the **Enable Security** check box, enter the appropriate information, and click **OK**.

Note To simplify deployments, either enable or disable security for all CTI OS components (clients, CTI OS Server, and silent monitor server).

Step 23 The CTI OS Security InstallShield Wizard appears if you have enabled security:

After the CTI OS Server security installation is complete, click **Finish**.



Note

- CTI OS Multi-Instance setup does not allow two or more CTI OS Servers to connect to the same CTI Server.
- The setup does not allow two or more CTI OS Servers to use the same listen port.
- Rerun the CTI OS Server setup after you complete the installation.

Uninstalling CTI OS Server

To uninstall the CTI OS Server, rerun the Setup program for the previous release and delete the Unified ICM Customer Instance that you specified during CTI OS Server Setup.

Determine Version Number of Installed Files

You can determine the version number of an installed CTI OS Server file by performing the following steps.

Procedure

- Step 1** Open a window for the ICM\CTIOS_bin subdirectory.
- Step 2** Highlight the file **ctiosservernode.exe**.
- Step 3** Right-click the highlighted file.
- Step 4** Select **Properties** from the drop-down menu.
The Properties dialog box appears.
- Step 5** Select the **Details** tab.

This tab contains version information (release number and build number) for the file.



CHAPTER 3

CTI Toolkit Desktop Client Installation

- [CTI Toolkit Desktop Client, on page 13](#)
- [Install Cisco CTI Toolkit Desktop Client Component, on page 14](#)
- [Uninstall CTI Toolkit, on page 17](#)
- [Determine Version Number of Installed CTI Toolkit Files, on page 17](#)
- [Unified CM Intercept Configuration Requirement, on page 18](#)
- [Configure Supervisory Assistance Features, on page 18](#)

CTI Toolkit Desktop Client

The CTI Toolkit Desktop Client consists of the following components:

- CTI Toolkit Desktop applications:
 - Agent desktop (including silent monitor)
 - UCCE Supervisor Desktop (including silent monitor)
 - Tools
- CTI Toolkit SDK (previously the CTI OS Developer's Toolkit, including necessary files, controls, documentation, and samples needed to write custom applications):
 - Win32
 - Java
 - .NET



Note Before you begin installation, verify that your system meets the hardware and software requirements for the components that you plan to install. See the *Contact Center Enterprise Compatibility Matrix* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html>.

For details on using Unified CCE in a virtualized environment, see the *Virtualization for Unified Contact Center Enterprise* at http://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/virtualization-unified-contact-center-enterprise.html.

Install Cisco CTI Toolkit Desktop Client Component

To install the CTI Toolkit Desktop Client components, perform the following steps.

Procedure

- Step 1** From the CTIOSClient directory on the CD, run **Setscreenup.exe**.
- Step 2** Click the **Next** button on the Welcome screen. The Software License Agreement screen appears.
- Step 3** Click the **Yes** button.
- A window displays the destination directory of the CTI Toolkit Desktop Client.
- Note** If you want to change the destination directory, you must uninstall and reinstall the Cisco CTI Toolkit Desktop Client Component.
- Step 4** Click **OK**.
The **Select Features** window appears.
- Step 5** Select the CTI Toolkit Desktop Client features that you want to install.
- Note** If you plan to use the Silent Monitor Service, you must select at least one of the CTI Toolkit Desktop Software components or the CTI Desktop SDK Win32 component.
- Step 6** Click **Next**.
The IMPORTANT NOTE window appears.
- Step 7** Click **Next**.
- If you select CTI Toolkit Agent Desktop or CTI Toolkit UCCE Supervisor Desktop, the CTIOS Server Information window appears.
- Note** Phones that are configured to use SRTP cannot be silently monitored. Customers who wish to silently monitor agents must not configure the agent phones to use SRTP.
- Enter the Name or IP Address and the Port Number for your CTI OS systems.
- Note** If you enable the QoS checkbox during the CTI OS Server Installation, you must select the checkbox at this stage.
- Step 8** The CVP Video Agent Desktop window appears. To enable the Cisco Voice Portal Video Agent Browser configuration, click the **Enable CVP Video** checkbox.
- Step 9** Click the **Next** button.
The Start Copying Files window appears.
- Step 10** Click the **Next** button to begin installation.
- Step 11** After the installation is complete, the following window appears, prompting you to install the Security feature. For more information about CTI OS Security, see [CTI OS Security, on page 47](#).
For more information about what Security Certificate option you must select, see [CTI OS Security, on page 47](#)
- Step 12** Click **OK**.

The CTI OS Security InstallShield Wizard appears. Click **Next**.

While Security is being configured, several status messages appear.

Step 13

Lastly, the CTIOS Setup Completed dialog box appears.

Step 14

Specify whether or not you want to restart your computer. Click the **Finish** button to exit Setup.

Localization

Next, import the configLanguages.reg registry, which is the registry file to configure language libraries for the CTI OS Agent and Supervisor phones. The following example provides steps for setting the CTIOS desktop language to French Canadian.

1. Log out of the CTIOS desktop.
2. Open the Registry Editor.
3. Import the **ConfigLanguages.reg** from the location `C:\Program Files (x86)\Cisco Systems\CTIOS Client\CTIOS Toolkit\Win32 CIL\Internationalization Kit\Languages`.
4. Refresh the Registry Editor and navigate to `HKEY_CURRENT_USER\Software\Cisco Systems, Inc.\CTI Desktop\Shared\Languages\<LAN>` where **<LAN>** is the abbreviation for the language code. In this example it will be **FRC**, as this customer is setting a French Canadian Localization.
5. Set the value of the DLL Key using the following path: `C:\Program Files (x86)\Cisco Systems\CTIOS Client\CTIOS Toolkit\Win32 CIL\Internationalization Kit\Languages\ctioslanguage.FRC.dll`.
6. Set the value of the Language Code key to **c0c (Hexadecimal)**.
7. Navigate to the `HKEY_CURRENT_USER\Software\Cisco Systems, Inc.\CTI Desktop\Shared\Languages\Last Language` and set the value of Language Code to **c0c (Hexadecimal)**.



Note If the value Last Language key is not set, the CTIOS uses the default locale of the Windows operating system.

8. Re-launch the CTIOS client desktop.

Installed Files

When you install the CTI Toolkit Agent Desktop or the CTI Toolkit UCCE Supervisor Desktop, the CTI Toolkit installation process installs a number of dynamic link libraries (DLLs). The installation process registers many of these DLLs automatically, but you must manually register some of these DLLs to work correctly.

The following table lists the Windows DLLs that are installed with the CTI Toolkit Agent Desktop or the CTI Toolkit UCCE Supervisor Desktop, along with the command line entry for manually registering the DLL (if needed).

The following Softphone Controls DLLs are installed with the CTI Toolkit Agent Desktop or the CTI Toolkit UCCE Supervisor Desktop:

- AgentGreetingCtl.dll
- AgentSelectCtl.dll
- AgentStateCtl.dll
- AlternateCtl.dll
- AnswerCtl.dll
- Arguments.dll
- BadLineCtl.dll
- ButtonControl.dll
- ChatCtl.dll
- conferenceCtl.dll
- CtiCommonDlgs.dll
- CTIOSAgentStatistics.dll
- ChatCtl.dll
- CTIOSCallAppearance.dll
- CTIOSClient.dll
- CTIOSSessionResolver.dll
- CTIOSSkillGroupStatistics.dll
- CTIOSStatusBar.dll
- EmergencyAssistCtl.dll
- GridControl.dll
- HoldCtl.dll
- IntlResourceLoader.dll
- MakeCallCtl.dll
- ReconnectCtl.dll
- RecordCtl.dll
- SilentMonitorCtl.dll
- SubclassForm.dll
- SupervisorOnlyCtl.dll
- TransferCtl.dll

If the CTI Toolkit Agent Desktop or the CTI Toolkit UCCE Supervisor Desktop indicate that a given DLL is not registered, you can manually register the DLL by using the following command:

```
regsvr32 <DLL filename>
```

For example, you can register CtiosStatusbar.dll by using the following command:

```
regsvr32 CtiosStatusbar.dll
```

With interoperability, the Win32 COM controls work under the .NET framework. The installation lays down the following files and installs them into the Global Access Cache (GAC):

AxInterop.AgentSelectCtl.dll	Cisco.CTICOMMONDLGSLib.dll	Interop.AgentSelectCtl.dll
AxInterop.AgentStateCtl.dll	Cisco.CTIOSARGUMENTSLib.dll	Interop.AgentStateCtl.dll
AxInterop.AlternateCtl.dll	Cisco.CTIOSCLIENTLib.dll	Interop.AlternateCtl.dll
AxInterop.AnswerCtl.dll	Cisco.CTIOSESSIONRESOLVERLib.dll	Interop.AnswerCtl.dll

AxInterop.BadLineCtl.dll	Cisco.INTLRESOURCELOADERLib.dll	Interop.BadLineCtl.dll
AxInterop.ButtonControl.dll		Interop.ButtonControl.dll
AxInterop.ChatCtl.dll		Interop.ChatCtl.dll
AxInterop.ConferenceCtl.dll		Interop.ConferenceCtl.dll
AxInterop.CTIOSAgentStatistics.dll		Interop.CTIOSAgentStatistics.dll
AxInterop.CTIOSCallAppearance.dll		Interop.CTIOSCallAppearance.dll
AxInterop.CTIOSSkillGroupStatistics.dll		Interop.CTIOSSkillGroupStatistics.dll
AxInterop.CTIOSStatusBar.dll		Interop.CTIOSStatusBar.dll
AxInterop.EmergencyAssistCtl.dll		Interop.EmergencyAssistCtl.dll
AxInterop.GridControl.dll		Interop.GridControl.dll
AxInterop.HoldCtl.dll		Interop.HoldCtl.dll
AxInterop.MakeCallCtl.dll		Interop.MakeCallCtl.dll
AxInterop.ReconnectCtl.dll		Interop.ReconnectCtl.dll
AxInterop.RecordCtl.dll		Interop.RecordCtl.dll
AxInterop.SilentMonitorCtl.dll		Interop.SilentMonitorCtl.dll
AxInterop.SubclassForm.dll		Interop.SubclassForm.dll
AxInterop.SupervisorOnlyCtl.dll		Interop.SupervisorOnlyCtl.dll
AxInterop.TransferCtl.dll		Interop.TransferCtl.dll
AxInterop.AgentGreetingCtl.dll		Interop.SHDocVw.dll

Uninstall CTI Toolkit

To uninstall the CTI Toolkit, run **Add/Remove** programs from the Windows Control Panel and select **Cisco CTI Toolkit Uninstall**.

Determine Version Number of Installed CTI Toolkit Files

If the CTI Toolkit Agent Desktop or the CTI Toolkit Supervisor Desktop for UCCE are currently running, the title bars of the desktop windows display the CTI Toolkit version number.

If these desktops are *not* currently running, you can determine the version number of an installed CTI Toolkit file by performing the following steps.

Procedure

- Step 1** Go to the directory:
Program Files\Cisco Systems\CTIOS Client\CTIOS Toolkit\Win32 CIL\COM Servers and Activex Controls
- Step 2** Highlight, and then right-click the **ctiosclient.dll** file.
- Step 3** Select **Properties** from the drop-down menu.
The Properties dialog box appears.
- Step 4** Select the **Version** tab.
This tab contains version information (release number and build number) for the file.
-

Unified CM Intercept Configuration Requirement

You must set the Cisco Unified CM service parameter named Drop Ad Hoc Conference to “never” (the default value), otherwise during the Intercept function, all the parties in the call get dropped.

Configure Supervisory Assistance Features

The CTI Toolkit Agent Desktop includes buttons that enable an agent to make an emergency call to a supervisor or to place a call to request assistance from a supervisor. To enable the functionality for these buttons, a *Unified ICM system administrator* must perform the following steps.

Procedure

- Step 1** Perform the following tasks from the Unified ICM Configuration Manager (for more information, see *Configuration Guide for Cisco Unified ICM/Contact Center Enterprise*).
- On the Dialed Number List screen, create a Dialed Number for the supervisor.
 - On the Agent Team List screen, enter the Dialed Number in the Supervisor script dialed number field.
- Step 2** Perform the following task from the Script Editor (for more information, see *Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise*).
- On the Call Type Manager screen, associate the Dialed Number with your script.
-



CHAPTER 4

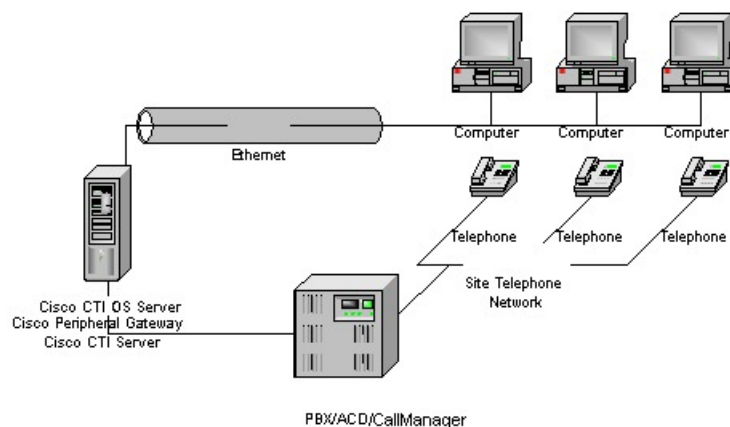
CTI OS Silent Monitor Installation and Configuration

- [Overview of CTI OS, on page 19](#)
- [System Manager Responsibilities, on page 21](#)
- [System Requirements, on page 21](#)
- [Silent monitoring, on page 22](#)

Overview of CTI OS

CTI OS combines a powerful, feature-rich server and an object-oriented software development toolkit to enable rapid development and deployment of complex CTI applications. Together, the Cisco CTI Server Interface, CTI OS Server, and CTI OS Client Interface Library (CIL) create a high performance, scalable, fault-tolerant three-tiered CTI architecture, as illustrated in following figure.

Figure 2: CTI OS Three-Tiered Architecture Topology



The CTI OS application architecture employs three tiers:

- The CIL is the first tier, providing an application-level interface to developers.

- The CTI OS Server is the second tier, providing the bulk of the event and request processing and enabling the object services of the CTI OS system.
- The Cisco CTI Server is the third tier, providing the event source and the back-end handling of telephony requests.

Advantages of CTI OS as Interface to Unified ICM Enterprise

CTI OS brings several major advances to developing custom CTI integration solutions. The CIL provides an object-oriented and event-driven Application Programming Interface (API), while the CTI OS Server does the *heavy-lifting* of the CTI integration: updating call context information, determining which buttons to enable on softphones, providing easy access to supervisor features, and automatically recovering from failover scenarios.

The key advantages of CTI OS include:

- **Rapid integration.** Developing CTI applications with CTI OS is easier and faster than any previously available Cisco CTI integration platform. The same object-oriented interface is used across programming languages, enabling rapid integrations in C++, Visual Basic, .NET, Java, or any Microsoft COM-compliant container environment.



Note The inclusion of the .NET toolkit allows for custom applications written in C#, VB.NET, or any other CLR-compliant language. By starting with the code for the .NET sample, the CTI Toolkit Combo Desktop developers can quickly customize the code without having to start from scratch.

CTI OS enables developers to create a screen-pop application in as little as five minutes. The only custom-development effort required is within the homegrown application to which you add CTI.

- **Complex solutions made simple.** CTI OS enables complex server-to-server integrations and multiple agent monitoring-type applications. The CIL provides a single object-oriented interface that you can use in two modes: agent mode and monitor mode. For more information about these two modes, see CTI OS Developer Guide for Cisco Unified ICM at: https://www.cisco.com/en/US/products/sw/custcosw/ps14/products_programming_reference_guides_list.html.
- **Fault tolerant.** CTI OS is built upon the Unified ICM Node Manager fault-tolerance platform, which automatically detects process failure and restarts the process, enabling work to continue. Upon recovery from a failure, CTI OS initiates a complete, system-wide snapshot of all agents, calls, and supervisors and propagates updates to all client-side objects.

Key Benefits of CTI OS for CTI Application Developers

The CTI OS CIL provides programmers with the tools required to rapidly develop high-quality CTI-enabled applications, taking advantage of the rich features of the CTI OS Server. Every feature of CTI OS was designed with ease of integration in mind, to remove the traditional barriers to entry for CTI integrations:

- **Object-oriented interactions.** CTI OS provides an object-oriented CTI interface by defining objects for all call center interactions. Programmers interact directly with Session, Agent, SkillGroup, and Call objects to perform all functions. CIL objects are thin proxies for the server-side objects, where all the 'heavy-lifting' is done. The Session object manages all objects within the CIL. A UniqueObjectID identifies

each object. Programmers can access an object by its UniqueObjectID or by iterating through the object collections.

- **Connection and session management.** The CTI OS CIL provides out-of-the-box connection and session management with the CTI OS Server, hiding all of the details of the TCP/IP sockets connection. The CIL also provides out-of-the-box failover recovery. Upon recovery from a failure, the CIL automatically reconnects to another CTI OS Server (or reconnects to the same CTI OS Server after restart), reestablishes the session, and recovers all objects for that session.
- **All parameters are key-value pairs.** The CTI OS CIL provides helper classes to treat all event and request parameters as simply a set of key-value pairs. All properties on the CTI OS objects are accessible by name via a simple Value = GetValue(“key”) mechanism. Client programmers can add values of any type to the CTI OS Arguments structure using the enumerated CTI OS keywords or their own string keywords (for example, AddItem[“DialedNumber”, “1234”). This provides for future enhancement of the interface without requiring any changes to the method signatures.
- **Simple event subscription model.** The CTI OS CIL implements a publisher-subscriber design pattern to enable easy subscription to event interfaces. Programmers can subscribe to the event interface that suits their needs, or use the AllInOne interface to subscribe to all events. Subclassable event adapter classes enable programmers to subscribe to event interfaces and only add minimal custom code for the events they use, and no code at all for events they do not use.

System Manager Responsibilities

The remainder of this document provides step-by-step procedures for the tasks a system manager must perform to set up and configure CTI OS. These tasks include:

- Installing CTI OS Server.
- Installing CTI Toolkit Agent Desktop, Supervisor Desktop, Tools, Documentation, Win32 SDK, Java SDK, and .NET SDK.
- Enabling CTI OS security.
- Using the Windows Registry Editor (regedit.exe) to configure the required CTI OS registry keys.
- Starting CTI OS and its associated processes from Unified CCE Service Control.



Note You *must* have administrator privileges to perform the procedures discussed in this manual.

System Requirements

See the *Solution Design Guide for Cisco Unified Contact Center Enterprise* at http://www.cisco.com/en/US/products/sw/custcosw/ps1844/products_implementation_design_guides_list.html

For more information on system requirements, see the *Virtualization for Unified Contact Center Enterprise* at http://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/virtualization-unified-contact-center-enterprise.html

Set User Privileges

On supported Windows client machines, users must have privileges that enables them to run legacy applications and have read/write access to the Cisco registry keys that the desktop applications use. To set user privileges to enable users to run CTI OS Agent Desktop and CTI OS Supervisor Desktop, an administrator must perform the following steps.

Procedure

-
- Step 1** On the Microsoft Windows Start Menu, select **Start > Run**.
- Step 2** Type in **regedt32** and click **OK**.
The Microsoft Windows Registry Editor window appears.
- Step 3** Go to the following registry location:
`HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTI Desktop\Ctios`
- Step 4** Select **Security > Permissions**.
A Permissions dialog box appears.
- Step 5** If you are adding a new user, perform the following steps.
- Click **Add**.
A Select Users dialog box appears.
 - Select the user to be added from the list in the top half of the Select Users dialog box.
 - Click **Add**, then click **OK**.
You return to the Permissions dialog box; the user you just added is now on the list.
- Step 6** Click the user whose privileges you want to set.
- Step 7** Set the Full Control permissions for this user to **Allow**.
- Step 8** Click **Apply**.
- Step 9** Click **OK**.
- Step 10** Exit Registry Editor.
-

Silent monitoring

Silent monitoring is a feature that allows a supervisor to eavesdrop on a conversation between an agent and a customer without allowing the agent to detect the monitoring session. Silent monitoring functionality can be provided by Cisco Unified Communications Manager (Unified CM) or CTI OS.

You can configure each CTI OS Server for either Unified CM-based or CTI OS-based silent monitoring.

Silent Monitor Differences Between Unified CM and CTI OS

Besides the differences in implementation, CTI OS and Unified CM also differ in when they can be invoked and when they end.

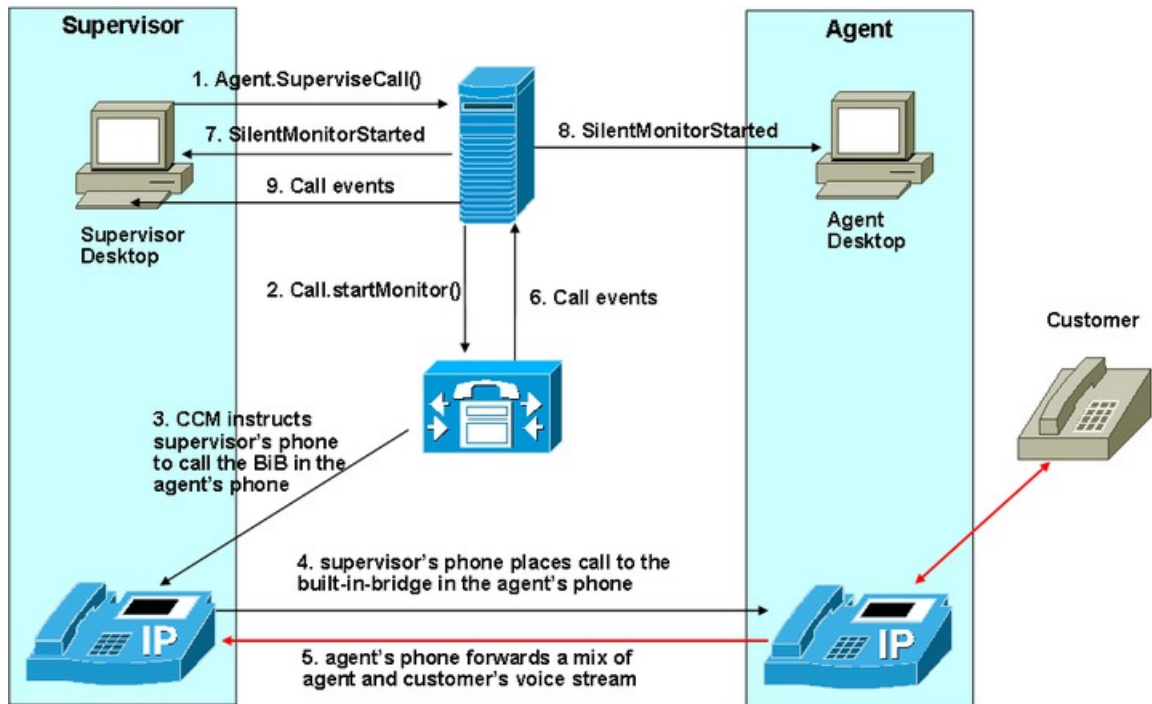
Table 1: Unified CM-Based and CTI OS-Based Silent Monitor Differences

Unified CM-Based silent monitor	CTI OS-Based silent monitor
The supervisor can only silent monitor an agent who is actively talking in a call.	The supervisor can silent monitor an agent in any state as long as the agent is logged in.
Supervisor cannot silent monitor an agent on hold.	Supervisor can silent monitor an agent on hold.
When agent consults, supervisor must stop silent monitoring held call and start silent monitoring conference.	When agent consults, supervisor automatically hears consult call.
Supervisor can only silent monitor in Not Ready state.	Supervisor can silent monitor in any state.
Supervisor must stop silent monitoring before barging in.	Supervisor can barge in while silent monitoring.
When the call that is being silent monitored ends, the silent monitor call ends. The supervisor must restart silent monitor after the agent answers another call.	When call ends, supervisor automatically silently monitors the next call as long as the supervisor has not stopped silent monitoring.

Unified CM-Based Silent Monitoring

Unified CM-based silent monitor allows a supervisor to listen in on agent calls in UCCE call centers. Supervisors can send silent monitor requests to monitor agents without the agent being aware of any monitoring activity. When the Unified CM-based approach is adopted for silent monitoring, the agent phone is used to mix the media streams of the agent call. The mix is then sent to the supervisor phone.

Figure 3: Unified CM-Based Silent Monitor



Unified CM Silent Monitor Advantages

Unified CM-based silent monitor provides the following advantages:

- No NIC card restrictions.
- Any 7.x or later version of any desktop (C++, Java, .Net) can be silent monitored provided the agent is not a mobile agent.
- Silent monitor is implemented via a call, therefore, the silent monitor call is carried on the voice LAN. With CTI OS silent monitor, the silent monitor stream is carried on the data LAN.
- Silent monitor calls are reported as agent-to-agent calls for supervisors. With CTI OS silent monitor, the time the supervisor spends silent monitoring is not tracked.

Unified Communications Manager Silent Monitor Limitations and Restrictions

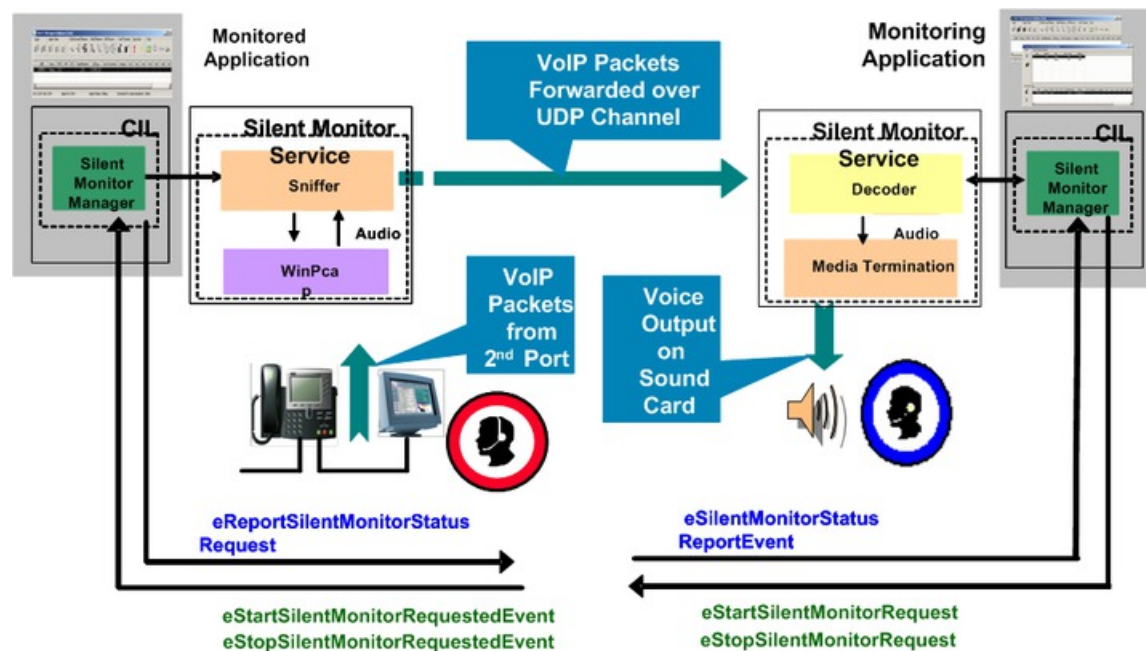
The following items prevent the use of Unified Communications Manager-based silent monitor:

- Agents using phones without a Built-In Bridge (BIB). See the Contact Center Enterprise Compatibility Matrix at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html> for a list of supported phones.
- Silent monitoring SRTP streams is not supported
- Mobile agents cannot use this method of silent monitoring

CTI OS-Based Silent Monitoring

CTI OS-based silent monitor allows a supervisor to listen in on agent calls in UCCE call centers that use CTI OS. Supervisors can send silent monitor requests to agent desktops without the agent being aware of any monitoring activity. Voice packets sent to and received by the monitored agent's IP desk phone are captured from the network and sent to the supervisor silent monitor service connected to the supervisor desktop. At the supervisor silent monitor service, these voice packets are decoded and played on the supervisor system sound card.

Figure 4: CTI OS-Based Silent Monitor



Note Silent monitor does not capture and translate DTMF digits that are selected on the CTI OS Agent Desktop or on agent desk phones.



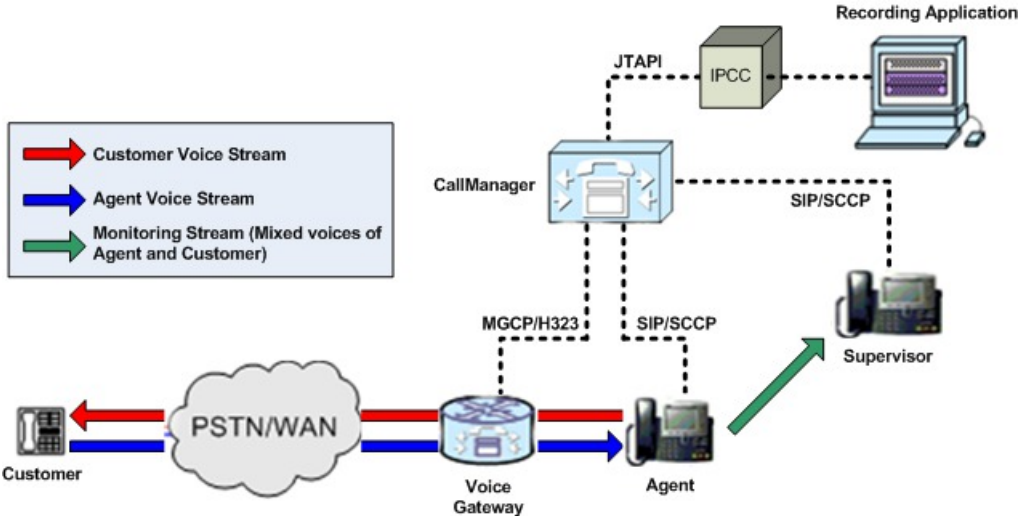
Note For the agent using the 7941, 7961, 7970, and 7971 phones, you must configure these devices on the Unified CM Administration web page with the “Span to PC Port”, “PC Voice VLAN Access” and the “PC Port” enabled. By default, the “Span to PC Port” is disabled and the “PC Voice VLAN Access” and the “PC Port” are enabled.

Network Topology for Silent Monitoring

Unified CM-Based Silent Monitoring

The following figure shows the network components and protocols involved in a Unified CM-based call monitoring session.

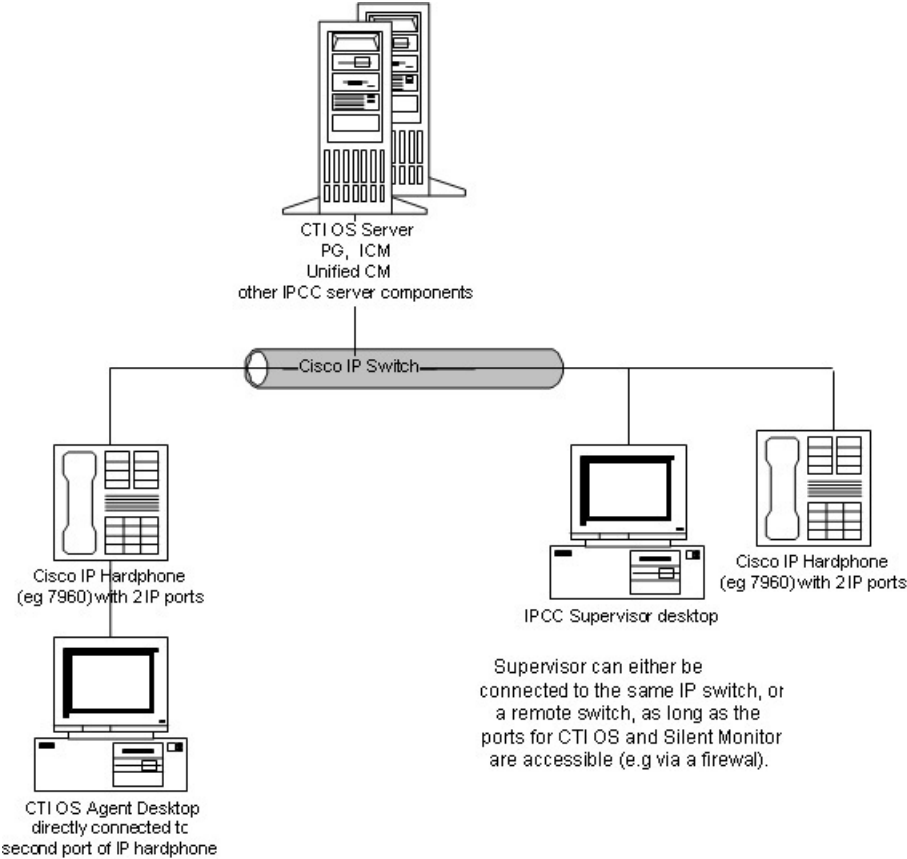
Figure 5: Unified CM-Based Silent Monitoring Network Topology



CTI OS-Based Silent Monitoring

The necessary network topology for non-mobile UCCE agents is shown in the following figure:

Figure 6: CTI OS-Based Silent Monitor Network Topology



Agents in this topology may have either an IP hardphone or IP Communicator. (The supervisor in this topology must have an IP hardphone. IP Communicator is not an option.) If the agent has an IP desk phone, it must have an agent desktop PC connected to the second IP port. If the agent has IP Communicator, you must install it on the same machine as the agent desktop.

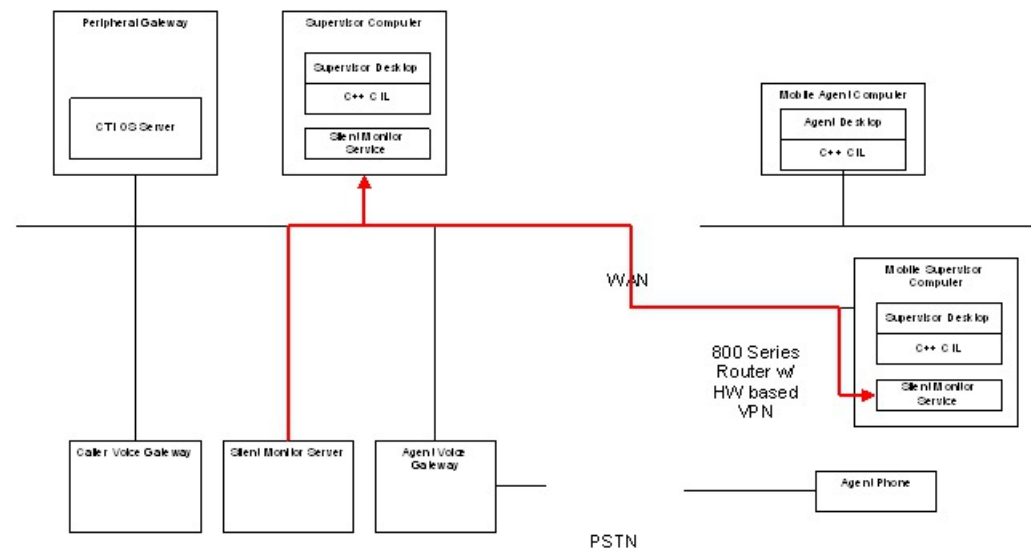
You must install a CTI OS-based desktop application that implements the CTI OS silent monitor feature on the agent desktop and supervisor desktop PCs. In addition, the components needed for an agent to be silently monitored are now automatically installed when the agent desktop is installed and those needed for a supervisor to do the silent monitoring are automatically installed when the UCCE Supervisor Desktop is installed.

Silent Monitoring and Mobile Agent Topology

You can also silently monitor mobile agents. To do this, you must manually deploy a standalone silent monitor server. This silent monitor server gains access to mobile agent voice traffic through a SPAN port that you configure to send all traffic to and from the agent gateway to the silent monitor server. The silent monitor server then filters and forwards voice traffic for the selected agent to the supervisor silent monitor server.

The necessary network topology is as follows.

Figure 7: Silent Monitoring and Mobile Agent Topology



Related Topics

[Silent Monitor Service Deployments](#), on page 42

Calculation of Additional Needed Bandwidth

Silent monitoring of an agent consumes almost the same network bandwidth as an additional voice call. If a single agent requires bandwidth for one voice call, then the same agent being silently monitored requires bandwidth for two concurrent voice calls.

For example, assume the following:

- You have 100 concurrent agents on your network.
- Up to 20% of the agents are monitored at any given time.

In this case, plan for network capacity for $100 + (20\% \text{ of } 100)$ concurrent calls, or 120 concurrent calls.

To calculate the total network bandwidth required for your call load, you would then multiply this number of calls by the per-call bandwidth figure for your particular codec and network protocol.

For example, the table on the Cisco Voice Over IP-Per Call Bandwidth Consumption website lists the per-call bandwidth on the G.711 codec (for a call with the default voice payload size) over Ethernet as 87.2 Kbps. You multiply this 87.2 Kbps by 120 calls to obtain the total required network bandwidth.

For more information about per-call bandwidths for various codecs and network protocols, see the Cisco Voice Over IP-Per Call Bandwidth Consumption website at

<http://www.cisco.com/c/en/us/support/docs/voice/voice-quality/7934-bwidth-consume.html>.

For more information about calculating bandwidth, see the Cisco Voice Codec Bandwidth Calculator at http://tools.cisco.com/Support/VBC/jsp/Codec_Calc1.jsp.



CHAPTER 5

CTI OS Component Installation

- [Silent Installation of CTI OS Components, on page 29](#)
- [Uninstall Components, on page 31](#)
- [Recover from Failed Installation of CTI OS, on page 31](#)

Silent Installation of CTI OS Components

CTI OS supports installation of some CTI OS components in unattended silent install mode. Silent install is supported for the following components:

- CTI OS Agent and Supervisor Desktops.
- CTI OS Server.



Note You must be aware of the following for CTI OS silent installation of the CTI OS agent and supervisor desktops:

- You must install .NET 4.7.1 prior to silent installation.
- You can install only CTI OS agent and supervisor desktops of CTI OS Client; you cannot silently install other CTI OS Client installation options.
- Only fresh silent installation is supported. You must uninstall all previous versions or patches of CTI OS Client prior to silently installing the CTI OS Client.
- Security is not installed when you install Client phones silently. If you need security, run **SecuritysetupPackage.exe** from the installation CD after you complete the installation.



Note Silent install is *not* supported for CTI OS Security Server and Client.

The silent installation process involves two tasks:

- Creating a response file.
- Using the response file to run CTI OS silent install on other machines.

**Warning**

Use of silent installations is discouraged, as errors encountered during the install process may go unnoticed and leave the system in an invalid state. If you choose to run installations silently, be absolutely certain that you manually perform the required pre- and post-installation instructions on the target systems.

Create a Response File

The process of creating a response file for use with the CTI OS silent install installs all CTI OS components (CTI OS Agent Desktop, CTI OS Supervisor Desktop, CTI OS Server) that exist on the machine where the response file is recorded. To create a response file for use with the CTI OS silent install, perform the following steps.

Procedure

-
- Step 1** From a command prompt, go to the directory where the CTI OS installation `setup.exe` file is located.
- Step 2** Run setup with the following option: `setup.exe /r`. This outputs a file called `setup.iss` to the `<drive>:\Windows` directory.
- Step 3** After the installation is complete, examine the setup log file to verify that installation ran to completion with no errors.
- Caution** It is critical that you verify that the installation process ran successfully and created a valid response file. Running the CTI OS silent install with an invalid response file can leave your system in an invalid state.
- Step 4** Reboot your system.
-

Run CTI OS Silent Install on Other Machines

After you create a response file on one machine, you can use that response file to run the CTI OS silent install on other machines. To do this, perform the following steps.

Procedure

-
- Step 1** On the machine or machines on which you want to run CTI OS silent install, copy the response file (`setup.iss`) to the same directory where `setup.exe` is located.
- Step 2** Run the setup with the following syntax: `setup.exe /s`.
- Step 3** When the installation is complete, examine the setup log file to verify that installation ran to completion with no errors.
- Caution** It is critical that you verify that the installation process ran successfully and created a valid response file. Running CTI OS silent install with an invalid response file can leave your system in an invalid state.

Step 4 Reboot your system.

Uninstall Components

To uninstall the CTI OS components, rerun the Setup program for the previous release and delete the Unified ICM Customer Instance that you specified during CTI OS component setup.

Recover from Failed Installation of CTI OS

If an attempted CTI OS installation fails for reasons such as power failure, disk error, or other similar circumstances, perform the following procedures to recover from the failed installation.

Procedure

- Step 1** Uninstall the release, as documented in [Uninstall Components](#).
- Step 2** Reinstall the release by performing the procedures documented in the following sections:
- [Install CTI OS Server, on page 6](#).
 - [Run Silent Monitor Service Installer, on page 36](#).
 - [Additional Configuration Steps, on page 39](#).
-



CHAPTER 6

Unified Communications Manager-Based Silent Monitor Configuration

- [Silent Monitor Service Installation and Configuration, on page 33](#)

Silent Monitor Service Installation and Configuration

This section provides an overview of the silent monitor service and discusses the tasks involved in installing and configuring the silent monitor service.



Note

- The terms silent monitor service and silent monitor server are used throughout this document.

Silent monitor service refers to a silent monitoring service running on an agent or supervisor desktop computer. This service handles silent monitoring functionality for one agent or supervisor.

Silent monitor server refers to a silent monitor service providing silent monitoring functionality for a group of mobile agents. These agents share the same gateway.

Silent Monitor Service Overview

The silent monitor functionality resides in a separate silent monitor service, rather than in the CIL. This is necessary to support the mobile agent environment. The C++ agent and supervisor desktops communicate with the silent monitor service via TCP connection. The agent desktop uses the silent monitor service to forward a voice stream to the supervisor's silent monitor service that plays the stream on the supervisor's computer speakers.

In a traditional UCCE environment, the silent monitor service runs alongside the agent and supervisor desktops on the agent's and supervisor's computer. However, the mobile agent environment does not give the CIL access to the voice packets because the agent's computer is not connected to the network through the agent's phone. These clients render the user interface for the desktops, but the actual desktop processes are running on the presentation server.

In mobile agent deployments, the voice path crosses the Public Switched Telephone Network (PSTN) and two gateways. One gateway control calls from customer phones. The other gateway controls agent calls. In this deployment, the silent monitor service is deployed from a SPAN port on the same switch as the agent gateway. This provides the silent monitor service with access to voice streams passing through the gateway.

In a mobile agent environment, the supervisor still uses a silent monitor service on the supervisor's desktop to play back the voice stream.

CTI OS Connections

To support remote silent monitoring, there can be up to nine All Event connections, including both CTI server and CTIOS server connections, subject to the following rules:

- Two CTI server All Event connections must be dedicated to CTIOS server
- Seven connections are available for other CTI server and CTIOS server All Events connections
- Of these seven connections, a maximum of five can be of the same connection type (5 CTI server connections or 5 CTIOS server connections)
- Average skills per agent should not exceed 10

How desktops Connect to Silent Monitor Services

The following is the supervisor desktop connection algorithm:

1. Connect the supervisor desktop to the silent monitor service running at port 42228 on localhost.



Note While CTI OS silent monitor clusters use port 42228 (the default), the silent monitor peers utilize port 42029 for communications purposes.

The following is the agent desktop connection algorithm:

1. If the agent desktop's connection profile specifies a silent monitor server or set of silent monitor servers, randomly choose a silent monitor server to connect to using the port present in the connection profile. For more information about how you configure a connection profile to include silent monitor services, see [CTI OS Server Installation, on page 5](#).
2. Connect the agent desktop to the silent monitor service running at port 42228 on localhost.



Note You can use a connection profile to override port 42228. In this case, desktops use the preceding algorithms to determine the address of the silent monitor service. After the address is determined, desktops connect using the determined address and the port that is present in the connection profile.

Configure ESXi Server

SPAN based silent monitoring service can be installed on UCS-C series servers version 5.1 and later for mobile and non-mobile agents. To install silent monitoring service on a virtual machine, perform the prerequisite steps in this topic and in the [Configure LAN Switch, on page 35](#) topic, and then follow the steps in [Run Silent Monitor Service Installer, on page 36](#) procedure.

Procedure

- Step 1** Configure a physical link from a switch to ESXi server.
- Step 2** Add a virtual machine port group on ESXi server for SPAN network.
- Step 3** To configure the virtual machine port group on ESXi server, perform the following steps:
- Open the ESXi where virtual machine port group is added.
 - Click on **Configuration** tab and navigate to **Networking** settings
 - In the virtual machine where port group is created. Click **Properties**.
 - In the **Ports** tab, click **Edit**.
 - Click **Security**.
 - In the **Policy Exceptions** section, from the Promiscuous Mode drop-down menu, select **Accept**.
 - Click **OK**.
 - In the **Ports** tab, highlight the virtual machine port group that you created.
 - Click **Edit**.
 - Click **Security**.
 - In the **Policy Exceptions** section, check the **Promiscuous Mode** check box.
 - Click **OK**.
 - Click **Close**.

The virtual machine port group for SPAN Port on ESXi server is configured.

What to do next

Add the created SPAN NIC to silent monitor service machine.

Configure LAN Switch

Procedure

- Step 1** In Cisco IOS LAN switches, configure the following ports.
- Configure access port that is connected to ESXi SPAN NIC using the following command:


```
#interface <interface ID>
#description CONNECTION TO ESXI SPAN PORT
#Switchport mode access
#switchport access vlan <VLAN ID>
```

where <interface ID> and <VLAN ID> variables are specific to user machine using the following command:
 - Configure access port that is connected to Gateway.


```
#interface <interface ID>
#description CONNECTION TO GATEWAY ROUTER
#Switchport mode access
#switchport access vlan <VLAN ID>
```

where <interface ID> and <VLAN ID> variables are specific to user machine.
- Step 2** Create SPAN session to monitor gateway traffic in Cisco IOS switches using the following command:

```
#monitor session 1 source interface <interface ID>both
#monitor session 1 destination interface <interface ID>
```

where <interface ID> variable is specific to user machine.

Step 3 Create SPAN session to monitor gateway traffic in Cisco CAT OS switches using the following command:

```
#set span <source port> <destination port> both session 1 inpkts enable learning enable
multicast enable
```

where <source port> and <destination port> variables are specific to user machine.

Step 4 Verify the SPAN session using the following command:

```
#Show monitor session <session ID>
```

where <session ID> variable is specific to user machine.

Upgrade Silent Monitor Service

If you are upgrading from a previous CTI OS release, you can install the next CTI OS Silent Monitor service release without uninstalling the Silent Monitor service.

Procedure

Run `setup.exe` from the `SMService` folder on the CTI OS installation media.

The upgrade of the stand-alone Silent Monitor server uninstalls the existing Silent Monitor server and installs the new version. The upgrade steps are the same as a fresh installation of Silent Monitor server.

Note This procedure only applies to upgrades of the stand-alone Silent Monitor service. The Silent Monitor service that runs on CTI OS agent and supervisor desktops upgrades silently during the CTI Toolkit Desktop Client upgrade process.

Run Silent Monitor Service Installer

The installer places two silent monitor service installers in the following directory:

```
<Install Drive>:\Program Files\Cisco Systems\CTIOS Client\CTIOS
Toolkit\Win32 CIL\Silent Monitor Files
```

The following installers are available and can be obtained from the Cisco.com:

- `SilentMonitorInstall_nogui.exe` – this executable silently installs the silent monitor service with the following settings:
 - Installed in the directory `C:\Program Files\CiscoSystems\CTIOS SilentMonitor`
 - Listens on port 42228
 - No Security

This executable runs automatically when you update from one release to another. It replaces the earlier release CIL with the newer CIL. The executable installs and starts the silent monitor service so that the agent and supervisor desktops do not lose functionality. Running this executable is sufficient only if you do not wish to override the default settings or enable Security.



Note This executable only works on the machines that have WinPCap 4.1.3 installed.

- **SMSelfExtractedInstallPackage.exe** – this executable extracts the silent monitor service setup program into the following directory:

```
<Install Drive>:\Program Files\Cisco Systems\CTIOS Client\CTIOS  
Toolkit\Win32 CIL\Silent Monitor Files\SilentMonitorServiceInstall
```

Run this executable if you wish to specify a different destination directory or port, or if you want to enable Security.

Procedure

Step 1

To run this executable silently:

- a) Open a command prompt window and navigate to the directory `<Install Drive>:\Program Files\Cisco Systems\CTIOS Client\CTIOS Toolkit\Win32 CIL\Silent Monitor Files\SilentMonitorServiceInstall`.
- b) Enter the command **setup.exe /s**.

Note This command runs the executable with the default values specified in the supplied answer file `setup.iss`. To override the default values, edit this answer file and change the values that you wish to change.

Step 2

To run the full installation program for this executable, perform the following steps:

- a) In Windows Explorer, navigate to the directory `<Install Drive>:\Program Files\Cisco Systems\CTIOS Client\CTIOS Toolkit\Win32 CIL\Silent Monitor Files\SilentMonitorServiceInstall`.
- b) Double-click the **setup.exe** file. The installation process begins.

Figure 8: Silent Monitor Service InstallShield Wizard I

You can either accept the default destination folder or click the **Browse** button and specify another directory.

- c) Click **Next**.

Specify the following information on this screen:

- **Port** – Enter the number of the port on which the silent monitor service listens for incoming connections.
- **Silent monitor Server** – Select this option to allow the silent monitor service to monitor many mobile agents simultaneously.

Note Install the silent monitor on its own VM; the silent monitor cannot be coresident with CTI OS Server or a Peripheral Gateway. For information about the silent monitor in a virtual environment, see the *Virtualization for Unified Contact Center Enterprise*

at http://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/virtualization-unified-contact-center-enterprise.html

- **Enter peer(s) information** – Select this option if this silent monitor service is part of a cluster of silent monitor services.
- **Hostname / IP address** – The hostname or IP address of the other silent monitor services in the cluster. Configure all services in a cluster to listen on the same port. For example, if you set the port to 42228 for one service, set it to 42228 for all other services in the cluster.

- d) Click **Next** to finish the installation process.
- e) Set up security.

Read the sections on using a self-signed certificate authority (CA) or a third-party CA for more information.

Sign a Silent Monitor Server Certificate Request with Self-Signed CA

Procedure

- Step 1** If the self-signed CA does not exist, then run `CreateSelfSignedCASetupPackage.exe` and store all the files that were created by the `CreateSelfSignedCASetupPackage.exe` program in a safe place. This step generates `CtiosRoot.pem` and `CtiosRootCert.pem` in the same folder from where the setup is run.
- Step 2** You must copy both `CtiosServerKey.pem` and `CtiosServerReq.pem` files from the CTI OS server machine located in `drive:\icm\Instance name\CTIOS1\Security` to the same directory as `CtiosRoot.pem` and `CtiosRootCert.pem`.
- Step 3** Run `SignCertificateSetupPackage.exe` from the same directory where `CtiosServerKey.pem`, `CtiosServerReq.pem`, `CtiosRoot.pem`, and `CtiosRootCert.pem` reside, select **CTI OS Server Certificate Request**, and enter the Ctios Certificate Authority password. This step generates `CtiosServer.pem` file if it is successful; otherwise it displays an error message.
- Step 4** Copy `CtiosServer.pem` and `CtiosRootCert.pem` back to the machine where silent monitor server resides and save them in the `C:\Cisco Systems\CTIOS\Silent Monitor\Security` directory.
- Step 5** Delete `CtiosServerkey.pem` located in `drive:\icm\Instance name\CTIOS1\Security` from the machine where CTI OS Server is installed.
- Step 6** Delete `CtiosServerKey.pem`, `CtiosServerReq.pem`, and `CtiosServer.pem` from the machine where `SignCertificateSetupPackage.exe` ran.
- Step 7** If the silent monitor server machine has a peer server, then:
 - a) Copy `CtiosClientkey.pem` and `CtiosClientreq.pem` files from the silent monitor server machine to the machine where `CtiosRoot.pem` and `CtiosRootCert.pem` reside. You must copy both `CtiosClientkey.pem` and `CtiosClientreq.pem` files to the same directory as `CtiosRoot.pem` and `CtiosRootCert.pem`.
 - b) Run `SignCertificateSetupPackage.exe` from the same directory where `CtiosClientkey.pem`, `CtiosClientreq.pem`, `CtiosRoot.pem`, and `CtiosRootCert.pem` reside, select **CTI Toolkit Desktop Client Certificate Request**, and enter the Ctios Certificate Authority password. This step generates `CtiosClient.pem` file if it is successful, otherwise it displays an error message.

- c) Copy CtiosClient.pem to the machine where silent monitor server resides and save it in the C:\Cisco Systems\CTIOS\Silent Monitor\Security directory.
 - d) Delete CtiosClientkey.pem from the machine where silent monitor server is installed.
 - e) Delete CtiosClientkey.pem, CtiosClientreq.pem, and CtiosClient.pem from the machine where SignCertificateSetupPackage.exe ran.
-

Sign a Silent Monitor Service Certificate Request with Third-Party CA

Follow these steps to sign a silent monitor service certificate request.

Procedure

- Step 1** Copy CtiosServerReq.pem file from the silent monitor service machine to the machine where the third-party CA resides.
- Step 2** Signing silent monitor service certificate request (CtiosServerReq.pem) with third-party CA generates a silent monitor service certificate. Rename it CtiosServerCert.pem.
- Step 3** The third-party CA has its certificate public information in a file. Rename this file CtiosRootCert.pem.
- Step 4** Copy CtiosServerCert.pem and CtiosRootCert.pem to the machine where the silent monitor service resides and save them in the C:\Cisco Systems\CTIOS\Silent Monitor\Security directory.
- Step 5** On the silent monitor service machine, copy the data in CtiosServerCert.pem and the data in CtiosServerkey.pem files into one file called CtiosServer.pem. The order is very important, so CtiosServer.pem must contain CtiosServerCert.pem data first and CtiosServerkey.pem data second.
- Step 6** Delete CtiosServerCert.pem and CtiosServerkey.pem from the silent monitor service machine.
- Step 7** If the silent monitor service machine has a peer server, then:
 - a) Copy CtiosClientreq.pem file from the silent monitor service machine to the machine where the third-party CA resides.
 - b) Signing CTI Toolkit Desktop Client certificate request (CtiosClientreq.pem) with third-party CA generates a CTI Toolkit Desktop Client certificate. Rename it CtiosClientCert.pem.
 - c) Copy CtiosClientCert.pem file to the machine where the silent monitor service resides and save it in the C:\Cisco Systems\CTIOS\Silent Monitor\Security directory.
 - d) On the silent monitor service machine, copy the data in CtiosClientCert.pem and the data in the CtiosClientkey.pem files into one file called CtiosClient.pem. The order is very important, so CtiosClient.pem must contain CtiosClientCert.pem data first and CtiosClientkey.pem data second.
 - e) Delete CtiosClientCert.pem and CtiosClientkey.pem from the silent monitor service machine.

If you are installing silent monitor service on a virtual machine, perform the steps listed in [Configure ESXi Server, on page 34](#) and [Configure LAN Switch, on page 35](#) to complete the installation process.

Additional Configuration Steps

This section discusses the silent monitor service configuration steps that you must perform after you install the silent monitor service. These steps are necessary to deliver silent monitor service connection information to client applications.

Rerunning CTI OS Server Setup

Rerun CTI OS Server setup to perform the following tasks:

- To configure agents to use the Silent Monitor service.
- To configure security for clients, so they can connect to Silent Monitor services that have security enabled.
- To configure mobile agents. When you rerun setup, enable mobile agent and the appropriate agent mode. This modifies the connection profile information in the registry. The **ShowFieldBitMask** is modified to display the RAS fields on the login dialog box and the **RasCallMode** registry key is added.
- To enable the default tracemark set it to 0x3.

Additional Configuration for Mobile Agent Environments

The following configuration considerations apply to environments that run mobile agent:

- In a mobile agent environment, the silent monitor service uses a Switched Port Analyzer (SPAN) port to receive the voice traffic that passes through the agent gateway. Most of the time, this requires the computer running the silent monitor service to have two NIC cards: one to handle communications with clients, and one to receive all traffic spanned from the switch.

Some switches do allow the destination port of a SPAN configuration to act as a normal network connection and in that case, only one NIC card is enough. See the "Network Traffic Restrictions" section in www.cisco.com/en/US/products/sw/custcosw/ps1001/products_tech_note09186a008010e6ba.shtml#umnic for more information on the types of catalyst switches that don't support outgoing traffic on SPAN destination port.

For example, if the agent gateway is connected to port 1 and the NIC on the silent monitor server that receives SPAN traffic is connected on port 10, the following commands are used to configure the SPAN session:

```
monitor session 1 source interface fastEthernet0/1
monitor session 1 destination interface fastEthernet0/10
```

Refer to your switch manual for details on configuring a span port. In general, traffic to and from the agent gateway's port must be forwarded to the port that is configured to receive span traffic on the silent monitor service.

- The SPAN source port should be set as the switch port into which the agent gateway (instead of the caller gateway) is plugged, or the silent monitor won't work in conference call scenarios.
- There must be two gateways: one gateway for agent traffic, and another for caller traffic. If one gateway is used for agent and caller traffic, the voice traffic does not leave the gateway and cannot be silently monitored.
- Voice traffic that does not leave the agent gateway or does not cross the agent gateway cannot be silent monitored. For example, agent-to-agent and consultation calls between mobile agents that share the same gateway cannot be silently monitored. In most mobile agent deployments, the only calls that can be reliably silent monitored are calls between agents and customers.
- All supervisors in a mobile agent environment must have the silent monitor service installed on their desktop.

- Agents do not need the silent monitor service configured on their desktops. However, you must configure the agent to use one or more silent monitor servers in the CTI OS Server setup program.
- If there are agents that can be both mobile and traditional Unified CCE, there must be at least two profiles for such agents. One profile, used when logging in as Unified CCE, does not contain any silent monitor service information. A second profile, used when logging in as a mobile agent, contains information used to connect to a silent monitor server. This enables the mobile agent to use the silent monitor service on their desktop computer and provides that mobile agent with silent monitoring functionality.

Silent Monitor Service Clusters

If more than one agent gateway is present in the call center, and an agent can use either gateway to log in, silent monitor services must be clustered to support silent monitor. You must deploy a separate silent monitor server for each gateway. You must configure a SPAN port for each silent monitor server as described in the previous section. You must then run the silent monitor server installer to install and configure the two silent monitor servers as peers. After you complete this, you must set up a connection profile to instruct the agent desktops to connect to one of the peers. (For more information on the CTI OS Server installer program, see [CTI OS Server Installation, on page 5](#).) To set up a connection profile, check the “Enter peer(s) information” checkbox and fill in the IP address of the other silent monitor service in the “Hostname/ip address” text box during silent monitor service installation (for more information, see Step 3 in the section [CTI OS Server Installation, on page 5](#)).

Installation of Silent Monitor Service with Windows Firewall Service Enabled

You must create a new port with the following parameters for any Windows server computer that has Windows Firewall Service enabled:

- Port Type: Silent Monitor Service Port
- Port Number: 42029



Note While CTI OS silent monitor clusters use port 42228 (the default), the silent monitor peers use port 42029 for communications purposes.

Harden Silent Monitor Server Security

You can run the ICM Security Hardening script only on Windows Server. To apply security hardening on a Silent Monitor Server, you must perform the following manual steps:

Procedure

- Step 1** Run the executable **SMSelfExtractedInstallPackage.exe**, which the installation process installs in the following directory:

```
<Install Drive>:\Program Files\Cisco Systems\CTIOS Client\CTIOS  
Toolkit\Win32 CIL\Silent Monitor Files
```

This executable puts a batch file named CopySecurityHardeningFiles.bat and the SecurityTemplate directory in the current directory.

- Step 2** Run **CopySecurityHardeningFiles.bat**.
This creates the directory C:\CiscoUtils and copies the corresponding files there.
- Step 3** Go to the directory C:\CiscoUtils\SecurityTemplate.
- Step 4** Run the command **cscript ICMSecurityHardening.vbe HARDEN**.
-

Add Silent Monitor Service to Windows Firewall Exceptions

The following steps describe how to add the silent monitor service as an exception if Windows Firewall is enabled on Windows Server:

Procedure

- Step 1** Go to **Windows Control Panel > System and Security > Windows Firewall**.
- Step 2** Based on your required network settings, turn on Windows Firewall.
- Step 3** Click **Allow apps to communicate through Windows Firewall**, and check the **CTIOS Silent Monitor** check box.
- Note** If you do not see the **CTIOS Silent Monitor** service on the list of programs, click **Allow another app...** button, and then click the **Browse** button. The silent monitor service executable, `SilentMonitorService.exe`, is located in the bin directory below the install directory.
-

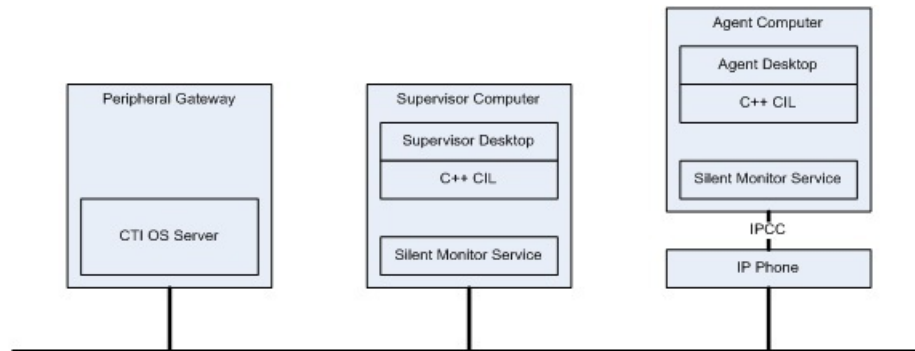
Silent Monitor Service Deployments

This section illustrates the following silent monitor service deployments:

- UCCE
- Mobile agent

Unified CCE Deployment

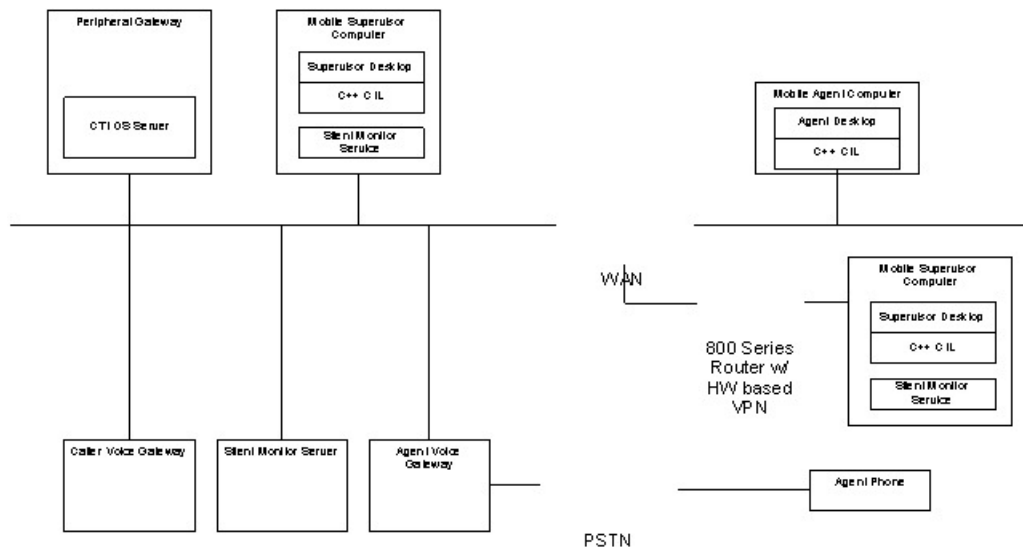
Figure 9: Unified CCE Deployment Topology



- When customers install desktops, the silent monitor service is installed on the agent desktop computer.
- The desktop is deployed behind the agent's phone. Silent monitor functionality is the same as before the upgrade. The only difference is that the service and not the CIL provides the silent monitor functionality.
- If the silent monitor service needs a different configuration than the one provided by the silent installer, then you must use `SMSelfExtractedInstallPackage.exe` to reconfigure the service.
- You can use a default Unified CCE connection profile for Unified CCE agents if no QoS is required. Otherwise you must configure a connection profile containing QoS settings. This works because CTI OS agent desktops attempt to connect to the localhost if no silent monitor services are configured using the connection profile.
- You can use a default Unified CCE connection profile for Unified CCE supervisors if no QoS is required. Otherwise, you must configure a connection profile containing QoS settings. This works because CTI OS supervisor desktops attempt to connect to localhost if no silent monitor services are configured via the connection profile.

Mobile Agent Using Analog/PSTN Phone

Figure 10: Mobile Agent Analog/PSTN Phone Topology

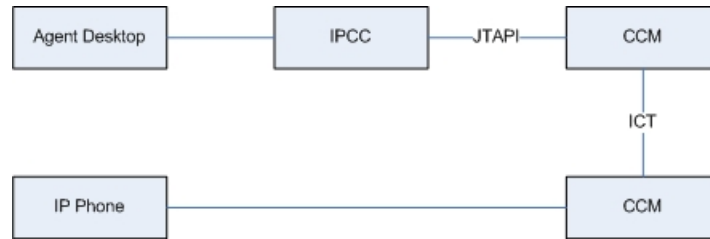


- Install a silent monitor server on a separate computer using the `SMSelfExtractedInstallPackage.exe` installer:
 - Make sure to check “Silent Monitor Server” when you install the silent monitor server.
 - This computer must have two NIC cards: one to receive SPAN port traffic and the other to receive control requests from clients and to forward monitored voice streams.
- Supervisors use the silent monitor service configured on supervisor's computer.
- Connection profiles are configured to tell mobile agents how to connect to the Silent Monitor servers.
- **SPAN port is configured on the switch.** Use the following steps to configure a SPAN port:
 - Locate the port on the switch where the agent voice gateway is connected.
 - Locate the port on the switch where the NIC card that receives SPAN traffic on the Silent Monitor server is connected.
 - Configure the switch to route SPAN traffic to the Silent Monitor server.
- The following commands are issued in global configuration mode if the voice gateway was connected to port 10 on the switch and the silent monitor service was connected to port 15.

```
no monitor session 1
monitor session 1 source interface fastEthernet0/10
monitor session 1 destination interface fastEthernet0/15
```

Mobile Agents IP Phones Topology

In some deployments, mobile agents use IP phones homed to a Unified CM other than the Unified CM used by UCCE. The following diagram illustrates the deployment of the agent phones.

Figure 11: Mobile Agents IP Phones Topology

In these cases, the silent monitor deployment is the same as the equivalent UCCE Agent deployment. The only difference is the Unified CM to which the agent's phone is homed. The following sections describe how to deploy silent monitor when mobile agents use IP phones.

Mobile Agent with IP Phone

The following Silent Monitor deployment uses mobile agents with IP phones that home to a different Unified CM from Unified CCE:

- When customers install or upgrade their desktops, the silent monitor service is silently installed on the agent desktop computer. The desktop is deployed behind the agent's phone; silent monitor functionality is the same as before the upgrade. The only difference is that the service and not the CIL provides the silent monitor functionality.
- If the silent monitor service needs a different configuration than the one provided by the silent installer, use `SMSelfExtractedInstallPackage.exe` to reconfigure the service.
- You can configure a connection profile with a registry key to allow agents and supervisors to log in as mobile agents.



CHAPTER 7

CTI OS Security

This chapter provides information about configuring the CTI OS Security Certificate and the Security Compatibility.

- [CTI OS Security Certificate Configuration, on page 47](#)
- [CTI OS Security Registry Keys, on page 51](#)
- [Security Compatibility, on page 53](#)

CTI OS Security Certificate Configuration

The CTI OS Security Certificate comprises the following:

- CTI OS Security Setup programs.
- Signing CTI Toolkit Desktop Client Certificate Request with Self-Signed Certificate Authority (CA).
- Signing CTI OS Server Certificate Request with Self-Signed CA.
- Signing CTI Toolkit Desktop Client Certificate Request with Third-Party CA.
- Signing CTI OS Server Certificate Request with Third-Party CA.

Each of these entities is detailed in this section.



Note Both Certificate Revocation List (CRL) and certificate chain are not supported in CTI OS Security.

CTI OS Security Setup Programs

To configure the CTI OS, three setup programs are implemented. These setup programs are part of the Win32 CTI OS toolkit installation, and are located in the directory `<drive>:\Program Files\Cisco Systems\CTIOS Client\CTIOS Security\Utilities`.

The first setup program, `CreateSelfSignedCASetupPackage.exe`, creates a self-signed certificate authority (CA). This must be run once if the customer wants to use a self-signed CA instead of a third party and the output of `CreateSelfSignedCASetupPackage.exe` must be saved in a secure place. This program creates CA-related files. One file, `CtiosRoot.pem`, contains the private CA information. This file must be kept in a safe place. Another file, `CtiosRootCert.pem`, contains public CA information. This setup program asks the

user to enter a password for the CA (between 8 and 30 characters), which are used when signing CTI OS certificate requests.

The second setup program, SecuritySetupPackage.exe, is used to generate certificate requests for both CTI Toolkit Desktop Client and CTI OS Server. If the certificate request is for the CTI OS Server, then it generates CtiosServerKey.pem, and CtiosServerReq.pem. These files are used when signing server certificates. If the certificate request is for the CTI Toolkit Desktop Client, then it generates CtiosClientkey.pem, and CtiosClientreq.pem. These files are used when signing client certificates.

The third setup program, SignCertificateSetupPackage.exe, is used to sign both CTI Toolkit Desktop Client and CTI OS Server certificates. This program is used only when the customer decides to sign their CTI Toolkit Desktop Client and CTI OS Server certificates with self signed CA. This program must reside in the same directory as the CtiosRootCert.pem and CtiosRoot.pem. If the certificate that is going to be signed is for the client, it generates CtiosClient.pem file. If the certificate that is going to be signed is for the server, it generates CtiosServer.pem file. This program asks the user to enter the following information:

- Ctios Certificate Authority Password. This password is the one used to create a self-signed CA.
- Select either CTI Toolkit Desktop Client Certificate Request or CTI OS Server Certificate Request.

Sign CTI Toolkit Desktop Client Certificate Request with Self-Signed CA



Note Generate CtiosRootCert.pem only once; use the same file for CTI OS server and client machines.

Follow these steps to sign a CTI Toolkit Desktop Client certificate request.

Procedure

- Step 1** If the self-signed CA does not exist, then run CreateSelfSignedCASetupPackage.exe and store all the files that were created by the CreateSelfSignedCASetupPackage.exe program in a safe place. This step generates CtiosRoot.pem and CtiosRootCert.pem in the same folder from where the setup is run.
- Step 2** Copy CtiosClientkey.pem and CtiosClientreq.pem files from the CTI Toolkit Desktop Client machine to the machine where CtiosRoot.pem and CtiosRootCert.pem reside.
- Note** You must Copy the Ctiosclientkey.pem and CtiosClientreq.pem files from the CTI Toolkit Desktop Client machine under <drive>:\Program Files\Cisco Systems\CTIOS Client\CTIOS Security to the folder where CtiosRoot.pem and CtiosRootCert.pem resides.
- Step 3** Run SignCertificateSetupPackage.exe from the same directory where CtiosClientkey.pem, CtiosClientreq.pem, CtiosRoot.pem, and CtiosRootCert.pem reside, select CTIOS Client Certificate Request, and enter the “Ctios Certificate Authority password.” This step generates the file CtiosClient.pem if it is successful; otherwise it displays an error message.
- Step 4** Copy both CtiosClient.pem and CtiosRootCert.pem back to the machine where CTI Toolkit Desktop Client is installed and save them in the <drive>:\Program Files\Cisco Systems\CTIOS Client\CTIOS Security directory.
- Step 5** Delete CtiosClientkey.pem in <drive>:\Program Files\Cisco Systems\CTIOS Client\CTIOS Security\Utilities directory from the machine where CTI Toolkit Desktop Client is installed.

- Step 6** Delete CtiosClientkey.pem, CtiosClientreq.pem, and CtiosClient.pem from the machine where SignCertificateSetupPackage.exe ran.
-

Sign CTI OS Server Certificate Request with Self-Signed CA



Note Generate CtiosRootCert.pem only once; use the same file for CTI OS server and client machines.

Follow these steps to sign a CTI OS Server certificate request.

Procedure

- Step 1** If the self-signed CA does not exist, then run CreateSelfSignedCASetupPackage.exe and store all the files that were created by the CreateSelfSignedCASetupPackage.exe program in a safe place. This step generates CtiosRoot.pem and CtiosRootCert.pem in the same folder from where the setup is run.
- Step 2** Copy CtiosServerKey.pem and CtiosServerReq.pem files from the CTI OS Server machine to the machine where CtiosRoot.pem and CtiosRootCert.pem reside.
- Note** You must copy both CtiosServerKey.pem and CtiosServerReq.pem files from the CTI OS server machine under `<drive>:\icm\Instance name\CTIOS1\Security` to the same directory as CtiosRoot.pem and CtiosRootCert.pem.
- Step 3** Run SignCertificateSetupPackage.exe from the same directory where CtiosServerKey.pem, CtiosServerReq.pem, CtiosRoot.pem, and CtiosRootCert.pem reside, select CTIOS Server Certificate Request, and enter the “Ctios Certificate Authority password.” This step generates CtiosServer.pem file if it is successful; otherwise it displays an error message.
- Step 4** Copy both CtiosServer.pem and CtiosRootCert.pem back to the machine where CTI OS Server resides and save them in the `<drive>:\icm\Instance name\CTIOS1\Security` directory.
- Step 5** Delete CtiosServerkey.pem under `<drive>:\icm\Instance name\CTIOS1\Security` from the machine where CTI OS Server is installed.
- Step 6** Delete CtiosServerKey.pem, CtiosServerReq.pem, and CtiosServer.pem from the machine where SignCertificateSetupPackage.exe ran.
- Step 7** If CTIOS Server has peer server, then:
- Copy CtiosClientkey.pem and CtiosClientreq.pem files from the CTI OS Server machine to the machine where CtiosRoot.pem and CtiosRootCert.pem reside. You must copy both CtiosClientkey.pem and CtiosClientreq.pem files to the same directory as CtiosRoot.pem and CtiosRootCert.pem.
 - Run SignCertificateSetupPackage.exe from the same directory where CtiosClientkey.pem, CtiosClientreq.pem, CtiosRoot.pem, and CtiosRootCert.pem reside, select **CTI Toolkit Desktop Client Certificate Request**, and enter the “Ctios Certificate Authority password.” This step generates CtiosClient.pem file if it is successful; otherwise it displays an error message.
 - Copy CtiosClient.pem to the machine where CTI OS Server resides and save it in `<drive>:\icm\<Instance name>\CTIOS1\Security` directory.
 - Delete CtiosClientkey.pem from the machine where CTI OS Server is installed.

- e) Delete CtiosClientkey.pem, CtiosClientreq.pem, and CtiosClient.pem from the machine where SignCertificateSetupPackage.exe ran.

Sign CTI Toolkit Desktop Client Certificate Request with Third-Party CA

Procedure

- Step 1** Copy CtiosClientreq.pem file from the CTI Toolkit Desktop Client machine to the machine where the third-party CA resides.
- Step 2** Signing CTI Toolkit Desktop Client certificate request (CtiosClientreq.pem) with third-party CA generates a CTI Toolkit Desktop Client certificate. Rename it CtiosClientCert.pem.
- Step 3** The third-party CA has its certificate public information in a file. Rename this file CtiosRootCert.pem.
- Step 4** Copy both CtiosClientCert.pem and CtiosRootCert.pem to the machine where CTI Toolkit Desktop Client resides and save them in the <drive>:\Program Files\Cisco Systems\CTIOS Client\Security directory.
- Step 5** On the CTI Toolkit Desktop Client machine, copy the data in CtiosClientCert.pem and the data in CtiosClientkey.pem files into one file called CtiosClient.pem. The order is very important, so CtiosClient.pem must contain CtiosClientCert.pem data first and then CtiosClientkey.pem data second.
- Step 6** Delete CtiosClientCert.pem and CtiosClientkey.pem from the CTI Toolkit Desktop Client machine.

Sign CTI OS Server Certificate Request with Third-Party CA

Follow these steps to sign a CTI OS Server certificate request.

Procedure

- Step 1** Copy CtiosServerReq.pem file from the CTI OS Server machine to the machine where the third-party CA resides.
- Step 2** Signing CTI OS Server certificate request (CtiosServerReq.pem) with third-party CA generates a CTI OS Server certificate. Rename it CtiosServerCert.pem.
- Step 3** The third-party CA has its certificate public information in a file. Rename this file CtiosRootCert.pem.
- Step 4** Copy both CtiosServerCert.pem and CtiosRootCert.pem to the machine where CTI OS Server resides and save them in the <drive>:\icm\<Instance name>\CTIOS1\Security directory.
- Step 5** On the CTI OS Server machine, copy the data in CtiosServerCert.pem and the data in CtiosServerkey.pem files into one file called CtiosServer.pem. The order is very important, so CtiosServer.pem must contain CtiosServerCert.pem data first and then CtiosServerkey.pem data second.
- Step 6** Delete CtiosServerCert.pem and CtiosServerkey.pem from the CTI OS Server machine.
- Step 7** If CTIOS Server has peer server, then:
 - a) Copy CtiosClientreq.pem file from the CTI OS Server machine to the machine where the third party CA resides.

- b) Signing CTI Toolkit Desktop Client certificate request (CtiosClientreq.pem) with third party CA generates a CTI Toolkit Desktop Client certificate. Rename it CtiosClientCert.pem.
- c) Copy CtiosClientCert.pem file to the machine where CTI OS Server resides and save it in the `<drive>:\icm\<Instance name>\CTIOS1\Security` directory.
- d) On the CTI OS Server machine, copy the data in CtiosClientCert.pem, and the data in CtiosClientkey.pem files into one file called CtiosClient.pem. *You must copy the files in this order*, so that CtiosClient.pem contain CtiosClientCert.pem data first and then CtiosClientkey.pem data second.
- e) Delete CtiosClientCert.pem and CtiosClientkey.pem from the CTI OS Server machine.

CTI OS Security Passwords

CTI OS Security introduces five types of passwords:

1. CTI OS Client certificate password: The administrator or installer enters this password when installing CTI OS Client security. This password is used for the CTI OS Client certificate request private key and it can be anything and the administrator or installer need not remember it.
2. CTI OS Server certificate password: The administrator or installer enters this password when installing CTI OS Server security. This password is used for the CTI OS Server certificate request private key and it can be anything and the administrator or installer need not remember it.
3. CTI OS Peer certificate password: The administrator or installer enters this password when installing CTI OS Server security. This password is used for the CTI OS Peer Server certificate request private key and it can be anything and the administrator or installer need not remember it.
4. Monitor Mode password: The administrator or installer enters this password when installing CTI OS Server security. This password is used by the agents when connecting to a secure CTI OS Server using CTI OS monitor mode applications such as AllAgents and AllCalls. This password must be the same on both CTI OS Peer Servers and the administrator or installer and whoever is using the CTI OS monitor mode applications must remember it.
5. Certificate Authority (CA) password: The administrator or installer enters this password when creating self-signed CA. The password can be anything and the administrator or installer must remember it because they must use it every time that this CA signs a certificate request.

CTI OS Security Registry Keys

The registry keys located at `[HKEY_LOCAL_MACHINE\SOFTWARE\CiscoSystems, Inc.\CTIOS\<CTIOS_Instancename>\CTIOS1\Server\Security]` define the settings for CTI OS Server Security.

Table 2: Registry Values for CTI OS Server

Registry Value Name	Value Type	Description	Default
AuthenticationEnabled	DWORD Value	For more information, see Authentication Mechanism, on page 54 .	1

Registry Value Name	Value Type	Description	Default
CAType	DWORD Value	Is created at install time. A value of 1 means the chosen CA type is self signed, and a value of 2 means the chosen CA type is third party.	1
NumBytesRenegotiation	DWORD Value	Is used for session renegotiation, which means requesting a handshake to be performed during an already established connection. This causes CTI OS Client credentials to be reevaluated and a new session to be created. It is important to replace the session key periodically for long-lasting SSL connections, because doing so makes the connection between the CTI OS Server and CTI OS Client more secure. Renegotiation happens after the CTI OS Server sends 10000000 bytes to the CTI OS Client. The minimum and the default value are 10000000.	10000000
SecurityEnabled	DWORD Value	Is created at install time. A value of 1 means CTI OS Security is enabled, and a value of 0 means CTI OS Security is disabled.	0
MonitorModeDisableThreshold	DWORD Value	Controls the number of consecutive failed attempts to access monitor mode functionality before monitor mode is disabled. Note For more information, see “Monitor Mode Security.”	3 (default)
MonitorModeDisableDuration	DWORD Value	Controls the length of time to disable monitor mode functionality after the configured number of consecutive failed attempts to access monitor mode functionality have occurred. Note For more information, see “Monitor Mode Security.”	15 minutes (default)

The registry keys located at [HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTI OS Client] define the settings for CTI OS Client Security. The following table lists the registry values for these keys.

Table 3: Registry Values for CTI OS Client

Registry Value Name	Value Type	Description	Default
CAType	DWORD Value	Is created at install time. A value of 1 means the chosen CA type is self signed, and a value of 2 means the chosen CA type is third party.	1
HandShakeTime	DWORD Value	Is created at install time. This key defines how long the CTI OS client waits during the SSL/TLS handshake phase.	5

Mode Security Monitoring

When the CTI OS Server has security enabled, the server guards itself against unlawful attempts to gain access to monitor mode functionality. It does this by tracking the number of failed attempts to access monitor mode functionality. After the configured number of consecutive failed attempts to access monitor mode functionality have occurred (3 by default), the CTI OS Server disables monitor mode functionality. When this happens, all attempts to access monitor mode functionality fail. This occurs until the configured period of time after the last failed attempt to access monitor mode functionality has passed. This time period is 15 minutes by default.

The *MonitorModeDisableThreshold* and the *MonitorModeDisableDuration* registry settings have been added to the `HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\Ctios\CTIOS<instance>\<ServerName>\Server\Security` to allow you to modify the defaults.

MonitorModeDisableThreshold

This registry field is a DWORD. It controls the number of consecutive failed attempts to access monitor mode functionality before monitor mode is disabled.

MonitorModeDisableDuration

This registry field is a DWORD. It controls the length of time to disable monitor mode functionality after the configured number of consecutive failed attempts to access monitor mode functionality have occurred.

Security Compatibility

Passing data over the network in a secure way is vital to both Cisco and the customer. CTI OS implements these features to deal with security:

Wire Level Encryption

To help secure all the traffic between the CTI OS Server and the CTI OS Client using Transport Layer Security (TLS). This protocol provides encryption and certification at the transport layer (TCP).

Authentication mechanism

For Unified CCE only, makes sure that an agent logs in successfully only if the agent supplies the correct password.

Wire Level Encryption

Wire Level Encryption provides an encryption mechanism between the latest version of CTI OS Server and CTI OS Client 11.x (y). By default, Wire Level Encryption is turned OFF. If the value of “SecurityEnabled” registry key is 0, then security is off. If the value of “SecurityEnabled” registry key is 1, then security is on. This key exists under:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<InstanceName>\CTIOS1\Server\Security
```

If the security is turned on in the CTI OS Server, then the CTI OS clients using .NET CIL, or Java CIL cannot connect to the CTI OS Server. If security is on in one CTI OS Server and this server has peers, then you must turn on security in the peers as well. The following table contains the list of CTI OS toolkits.

Table 4: Wire Level Encryption: List of CTI OS Toolkits

	C++ CIL Toolkit	COM CIL Toolkit	Java CIL Toolkit	.NET CIL Toolkit
Support Wire Level Encryption	Yes	Yes	No	No

Authentication Mechanism

The authentication mechanism is for Unified CCE only. It is on by default. If the value of “AuthenticationEnabled” registry key is 0, then authentication is off. If the value of “AuthenticationEnabled” registry key is 1, then authentication is on. This key exists under:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<InstanceName>\CTIOS1\Server\Security
```

For all peripherals other than Unified CCE, this registry key is not used.



CHAPTER 8

CTI OS Configuration

- [Use Windows Registry Editor, on page 55](#)
- [Virtual Desktop Infrastructure, on page 57](#)
- [CTI Driver Key, on page 58](#)
- [EMS Tracing Values, on page 59](#)
- [Server Registry Key, on page 60](#)
- [MainScreen Registry Key, on page 70](#)
- [Unified CCE Silent Monitor Configuration, on page 70](#)
- [ConnectionProfiles Registry Key, on page 71](#)
- [Call Appearance Grid Configuration, on page 77](#)
- [Customize Agent Statistics Grid Configuration, on page 82](#)
- [Automatic Skill Group Statistics Grid Configuration, on page 83](#)
- [Configure Additional Peripherals, on page 85](#)
- [Quality of Service/Type of Service, on page 85](#)

Use Windows Registry Editor

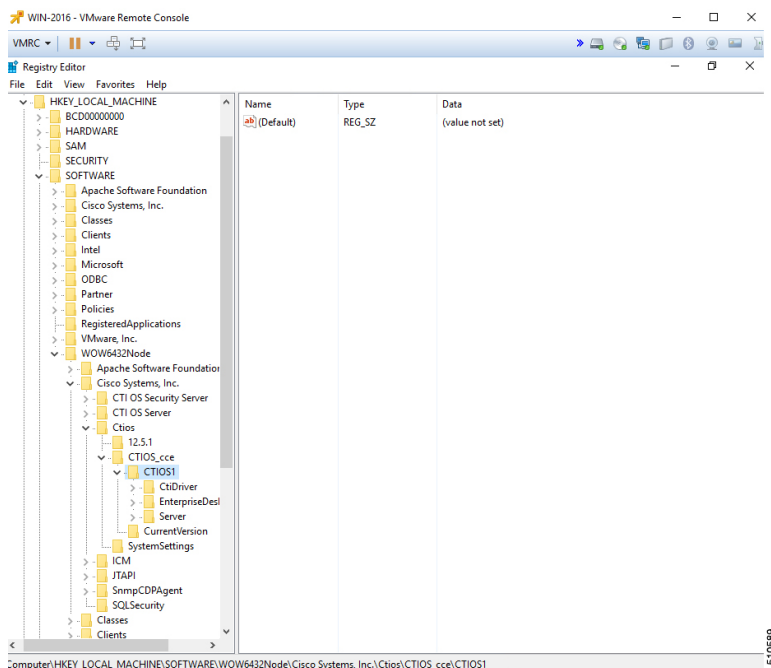
CTI OS configuration is handled through the Windows Registry Editor. Using the Editor, you can add or change registry values.



Note Except where otherwise indicated, the CTI OS registry keys discussed in this chapter are local and start at the [HKEY_LOCAL_MACHINE\SOFTWARE\ Cisco Systems, Inc.\CTIOS*<CTIOS_InstanceName>*\<CTIOSServerName>] path.

CTI OS Server installation initializes a configuration that is stored in the Windows System Registry database. You can access and edit this configuration through the Windows Registry Editor (regedit.exe). The following table shows the Registry Editor main window.

Figure 12: Windows Registry Editor Main Window



To add a key or registry value under an existing key, perform the following steps.

Procedure

-
- Step 1** Highlight the existing key in the left panel.
- Step 2** Position the cursor in the right panel and click.
- A popup menu appears.
- Step 3** From the popup menu, select **Key**, **String Value**, **Binary Value**, or **DWORD** value.
- If you select **Key**, a placeholder for the key you want to add appears highlighted in the left panel. For other items, a placeholder for the item you want to add appears highlighted in the right panel.
- Step 4** Right-click the highlighted item. A popup menu appears.
- To name the item, select **Rename** from the popup menu; then type the new name for the item.
 - To set the value data for **String**, **Binary**, and **DWORD** values, select **Modify**. A dialog box appears. Enter the value data following the **Value Data** prompt.

To edit an existing key or registry value, highlight the key or value and right-click it. Select **Modify**, **Delete**, or **Rename** from the popup menu and proceed.



Note After you change the registry, restart the CTI OS processes before the new setting can take effect.

Silent Monitor Type Configuration for CTI OS

You can configure the CTI OS can be configured to use either the Unified CM-based silent monitor or the CTI OS-based silent monitor. You accomplish this by setting the following field in the CTI OS registry:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems Inc.\CTIOS\<CTIOS InstanceName>\<CTIOSServerName>\EnterpriseDesktopSettings\All Desktops\IPCCSilentMonitor\Name\Settings\CCMBasedSilentMonitor
```

This field is a DWORD and if present and set to “1”, Unified CM-based silent monitor is used.

CTI OS-based silent monitor is used if this field is not present or if it is present and set to “0”.



Note You can also run the setup program to reconfigure the CTI OS-based silent monitor.

If the server setup program is not run, the CCMBasedSilentMonitor field is not present. As a result, CTI OS-based silent monitor is used.

Virtual Desktop Infrastructure

Virtual Desktop Infrastructure (VDI) is a server-centric computing model. It is designed to help you to host and centrally manage desktop virtual machines in the data center, while providing a full PC desktop experience.

The VMware View portfolio of products VDI lets IT run virtual desktops in the data center while giving you a single view of all your applications and data in a familiar, personalized environment on any device at any location. VDI provides greater flexibility, reliability, efficiency, and security managing desktops and applications from the datacenter.

CTI OS Desktop Installations on VDI Agent Desktops

Prerequisites

Complete functional VDI deployment as per the VDI requirements. For more information, see <http://www.vmware.com/products/view/>.

Install CTI OS Desktop on VDI Agent

Procedure

- Step 1** On any VDI agent desktop, run the CTI OS client installer and configure the desktop. For more information on the deployment, limitations, and supported features of CTI OS desktops on VDI, see *CTI OS System Manager Guide for Cisco Unified ICM*.
 - Step 2** When the installation is complete, launch the CTI OS desktop and verify basic functionality by logging in an agent, changing agent states, or making calls.
 - Step 3** After you complete the testing, follow the same steps on the other VDI agent desktops.
-

Notes and Restrictions

Silent Monitoring

CTI OS-based silent monitoring is not supported due to physical limitations. For CTI OS-based silent monitoring, you must connect the agent machine to the network via the phone hard-set. This cannot be achieved with a Virtual Machine, such as when using VDI.

ThinApp

For more information about ThinApp, see <http://www.vmware.com/products/thinapp>.

CTI Driver Key

The CTI Driver key includes registry settings required for CTI Server connection. The CTI Driver key contains one key, the Config key. The following table describes the CtiDriver/Config key registry values.

Table 5: Registry Values for [CtiDriver\Config]

Registry Value Name	Value Type	Description	Default
ClientID	String Value	The identifier of the CTI Client. This appears in the CTI Server log file to help identify which session the CTI OS Server is connected on.	CTIOSServer
ClientPassword	String Value	The password of the CTI Client. This appears in the CTI Server log file to help identify which session the CTI OS Server is connected on.	CTIOSServer
ClientSignature	String Value	The signature of the CTI Client. This appears in the CTI Server log file to help identify which session the CTI OS Server is connected on.	CTIOSServer
SideAHost	String Value	The CTI Server (sideA) IP address or hostname to which the CTI OS Server connects.	Host specified during CTI Server installation.
SideAPort	DWORD Value	The CTI Server (sideA) IP port to which the CTI OS Server connects.	Port specified during CTI Server installation.
SideBHost	String Value	The CTI Server (sideB) IP address or hostname to which the CTI OS Server connects.	Host specified during CTI Server installation.
SideBPort	DWORD Value	The CTI Server (sideB) IP port to which the CTI OS Server connects.	Port specified during CTI Server installation.
Heartbeat Interval	DWORD Value	The interval (in seconds) at which HEARTBEAT_REQ messages are sent to the CTI Server.	5

Registry Value Name	Value Type	Description	Default
ServicesMask	DWORD Value	<p>The services requested from the CTI Server and provides the functionality that the MinimizeAgentStateEvents registry value used to provide.</p> <p>To suppress multiple state events add the bit:CTI_SERVICE_IGNORE _DUPLICATE_AGENT_STATES = 0x00100000</p> <p>to the following registry key: HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\Ctios\ CTIOS_<ctios_instance>\CTIOS1\ CtiDriver\Config</p> <p>Example: Change “ServicesMask”=dword:000c0016 to: “ServicesMask”=dword:001c0016</p>	0x00000296 (52)(default)
CallMsgMask	DWORD Value	The unsolicited call events requested from the CTI Server.	0x00ffffff (16777215)
AgentStateMask	DWORD Value	The agent states requested from the CTI Server.	0x000003ff (1023)
ProtocolVersion	DWORD Value	The highest protocol version to use when connecting to the CTI Server. The highest common denominator is used when establishing the CTI Session.	15
IdleTimeout	DWORD Value	The session inactivity timeout (in seconds). The CTI Server disconnects clients after this time threshold has elapsed without other socket messages.	0x7fffffff (2147483647)
MemoryPoolSize	DWORD Value	Size of the memory pool, in bytes.	0x00000064 (100)

EMS Tracing Values

The registry keys located at [HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\ICM\<customer_instance_name>\<CTIOSComponent Name>\EMS\CurrentVersion\Library\Processes\ctios] define the settings for Event Management System (EMS) tracing. The following table lists the registry values for these keys.

Table 6: Registry Values for EMS Tracing

Registry Value Name	Value Type	Description	Default
EMSDisplayToScreen	DWORD Value	If set to 1, EMS routines attempt to write formatted messages to standard output.	0
EMSAllLogFilesMax	DWORD Value	The maximum total number of bytes that the EMS library writes to all local log files.	2000000000 (2 billion decimal)
EMSBreakOnExit	DWORD Value	If set to 1, EMS exit routines invoke the Debugger.	0
EMSBreakOnInit	DWORD Value	If set to 1, EMS initialization routines invoke the Debugger.	0
EMSDebugBreak	DWORD Value	If set to 1, EMS failure routines invoke the Debugger before exiting the process.	1
EMSLogFileCountMax	DWORD Value	The maximum number of log files that the EMS library writes.	10000 (decimal)
EMSLogFileLocation	String Value	The directory where the EMS library creates local log files.	Default directory specified at installation.
EMSLogFileMax	DWORD Value	The maximum number of bytes that the EMS library writes to a single local log file.	30000000 (30 million decimal)
EMSNTEventLogLevel	DWORD Value	The minimum severity event that EMS logs in the Application Event Log.	2
EMSTraceMask	DWORD Value	A bitmask that specifies the levels of EMS tracing that are enabled.	395791 (decimal)
EMSUserData	DWORD Value	Placeholder for arbitrary binary user data.	
EMSForwardLevel	DWORD Value	The minimum severity event that EMS forwards to the Unified ICM central controller.	0

Server Registry Key

The Server registry key contains CTI OS Server related configuration information. It contains the following subkeys:

- Agent
- CallObject
- Connections
- Device

- Peers
- Peripherals
- SkillGroup
- SilentMonitor
- Supervisor
- ThreadPoolSize
- TimerService

Agent Registry Key

The Agent key contains agent related configuration information. The following table lists the registry values for the Agent key.

Table 7: Registry Values for [ServerAgent]

Registry Value Name	Value Type	Description	Default
AgentChatLevel	string	Defines the call center personnel with whom an agent is permitted to chat. You must set this to one of the values listed in the “AgentChatLevel values” table below.	TeamSupervisors
EnableWrapupDialog	DWORD Value	When enabled (1), a Wrapup dialog box pops up at the end of the call. A value of 0 disables this feature.	1
forceLogoutOnSessionClose	DWORD Value	Set to “1” to turn on the feature to force logout an agent when their session is ended by the agent closing the window without properly logging out. Note You must manually enter this value into the registry. If the value is not entered into the registry, the effect is the same as having it set to its default (0).	0

Registry Value Name	Value Type	Description	Default
forceLogoutOnSessionCloseReason(Optional unless logout reason is required.)	DWORD Value	<p>Indicates the reason code to be used by the CTI OS Server when the agent is forced to log out.</p> <p>This need not be defined in the registry when the default value is sufficient. By setting this to a specific reason code you can easily determine when an Agent is logged out by the CTI OS Server and when the Agent logs out normally.</p> <p>Note You must set this to a non-zero value if an idle reason code reason is required. Refer to “Unified ICM Agent Desk Settings” to determine if the idle reason code is required.</p> <p>Note You must manually enter this value into the registry.</p>	0
forceNotReadyOnSessionCloseReason(Optional unless idle reason is required.)	DWORD Value	<p>Indicates the reason code to be used by the CTI OS Server when the agent is forced to the Not Ready state before being forced to log out.</p> <p>This need not be defined in the registry when the default value is sufficient. By setting this to a specific reason code you can easily determine when an Agent is logged out by the CTI OS Server and when the Agent logs out normally.</p> <p>Note You must set this to a non-zero value if an idle reason code reason is required. For more information about the required idle reason code, see “Unified ICM Agent Desk Settings”.</p> <p>Note You must manually enter this value into the registry.</p>	0
LogoutReasonRequired	DWORD Value	On all switches except UCCE, when enabled (1) a Logout Reason Code dialog box pops up when changing state to Logout. On all switches, a value of 0 disables this feature.	0
NotReadyReasonRequired	DWORD Value	On all switches except UCCE, when enabled (1) a Not Ready Reason Code dialog box pops up when changing state to NotReady. On all switches, a value of 0 disables this feature.	0

Registry Value Name	Value Type	Description	Default
PollForAgentStatsAtEndCall	DWORD Value	Controls when agent statistics are sent from CTI OS Server to CTI OS clients. A value of 0 means that agent statistics are sent at a regular interval (specified in PollingIntervalSec). A value of 1 means that agent statistics are sent only when a call ends. Note Changing the value of PollForAgentStatsAtEndCall may degrade performance.	1
PollingIntervalSec	DWORD Value	The agent statistics polling interval, in seconds.	15
WrapupDataRequired	DWORD Value	When enabled (1), wrapup data is mandatory. When disabled (0), wrapup data is not required. Not applicable to UCCE agents.	0

Table 8: AgentChatLevel Values

Value	Meaning
Disabled	All agent chat disabled.
PrimarySupervisor	Agents can chat only with primary supervisor of their team.
TeamSupervisors	Agents can chat with the primary or secondary supervisor of their team.
Team	Agents can chat with anyone in team.
Unrestricted	Agents can chat with anyone on the same peripheral.

The Agent key also contains the following subkeys:

- ReasonCodes
- WrapupStrings

ReasonCodes Registry Key

The ReasonCodes key is a site-specific key that defines the reason codes the CTI OS Agent Desktop uses. For each reason code, a string is mapped to an unsigned short value. The CTI OS Agent Desktop displays the string and sends the appropriate value to the CTI Server, which in turn passes the value along to the ACD.

The ReasonCodes key contains two subkeys:

- **Logout.** This key defines the reason codes that appear on the Select Reason: Logout screen when an agent logs out. Immediately following CTI OS Server installation, the Logout registry key contains four values that serve as placeholders for Logout reason codes (see following table).

Table 9: Initial Contents of [Server\Agent\ReasonCodes\Logout]

Registry Value Name	Value Type	Description
Insert logout reason code 1 here	DWORD Value	Placeholder for first Logout reason code.
Insert logout reason code 2 here	DWORD Value	Placeholder for second Logout reason code.
Insert logout reason code 3 here	DWORD Value	Placeholder for third Logout reason code.
Insert logout reason code 4 here	DWORD Value	Placeholder for fourth Logout reason code.

To define the text that appears for each Logout reason code in the Select Reason dialog box, set the value data associated with the reason code to the text you want to appear for that reason code. You may also add additional reason code entries as needed.

- **NotReady.** This key defines the reason codes that appear in the Select Reason: NotReady dialog box when an agent goes to NotReady state. As with the Logout key, the NotReady key initially contains four placeholder DWORD values that you can edit to define the reason codes in the Select Reason: NotReady dialog box.



Note The maximum length permitted for a reason code is 42 characters.

WrapupStrings Registry Key

The WrapupStrings key defines the predefined wrapup text strings that appear in the softphone Wrapup dialog box. The WrapupStrings key contains a subkey, Incoming, that defines the wrapup text for incoming calls. Immediately following CTI OS Server installation, the Incoming key contains the registry values listed in the following table.

Table 10: Initial Contents of [Server\Agent\WrapupStrings\Incoming]

Registry Value Name	Value Type	Description
String0	String Value	Placeholder for first wrapup text string.
String1	String Value	Placeholder for second wrapup text string.
String2	String Value	Placeholder for third wrapup text string.
String3	String Value	Placeholder for fourth wrapup text string.

To define the text that appears for each wrapup text string in the WrapUp dialog box, set the value data associated with the reason code to the text you want to appear for that wrapup string. You may also add additional wrapup string entries as desired.



Note There are no CTI OS registry keys for defining text for outgoing wrapup strings. The Unified ICM does not save any wrapup data for outgoing calls, so you need not define outgoing wrapup strings. This is applicable to transfer and conference initiated calls also. (Both transfer and conference calls are treated as outgoing calls.)

CallObject Registry Key

The CallObject key defines the values pertaining to call objects. The following table defines the CallObject key registry values.

Table 11: Registry Values for [Server\CallObject]

Registry Value Name	Value Type	Description	Default
AgentPreCallEvent Timeout	DWORD Value	Length of time, in seconds, within which an AGENT_PRE_CALL_EVENT must be followed by a BEGIN_CALL_EVENT or the call object is deleted.	30
IPCCConference_ SupportsMultipleControllers	DWORD Value	When set to 1, allows all parties of a Conference to add new parties to the conference as supported by Unified CM. If running against an earlier version of Unified CM, this registry value must be set to 0. If this value is not set to 0 when running against an earlier version of Unified CM, and a non-controller Conference party tries to make a Consult Call for a Conference, the party receives a Control Failure.	1
MinimizeEventArgs	DWORD Value	When set to 1 (optimal), minimizes the amount of nonessential call object parameters sent to the client.	1
TrashCollectionInterval Sec	DWORD value	Controls how often (in seconds) the trash collector activates and removes any stale objects from memory. A value of 0 disables the trash collector.	7200

Connections Registry Key

The Connections key defines the values for client connections to the CTI OS Server. The following table defines the Connections key registry values.

Table 12: Registry Values for [Server\Connections]

Registry Value Name	Value Type	Description	Default
ClientPoolInitialSize	DWORD Value	The number of Client objects to pre-create. Caution Leave this registry entry set to its default value.	1500
ClientPoolMinSize	DWORD Value	The minimum number of Client objects in the pool to trigger growing the pool. Caution Leave this registry entry set to its default value.	50
ClientPoolIncrement	DWORD Value	The number of Client objects to create when the pool must be grown. Caution Leave this registry entry set to its default value.	50
HeartbeatIntervalMs	DWORD Value	The number of milliseconds between heartbeats from the server to its clients.	60000
HeartbeatRetrys	DWORD Value	The number of missed heartbeats before a connection is closed for unresponsiveness.	5
ListenPort	DWORD Value	The TCP/IP port on which the CTI OS Server listens for incoming client connections.	Port specified during CTI OS Server setup.
MaxMonitorModeConnections	DWORD Value	This registry entry controls the number of monitor mode connections connected to a CTI OS Server.	7

The heartbeating mechanism uses the HeartbeatIntervalMs and HeartbeatRetrys values together to determine when a connection is stale and must be closed. The interval serves as a timeout and the retries is the number of attempts that have timed out before closing the socket.

Example with an interval of 5 seconds and three retries:

- After 5 seconds (Total time), if the server does not receive a response from the client, it sends a heartbeat request and increments the retry count to 1.
- After another 5 seconds, if the server does not receive a response from the client, it sends a heartbeat request and increments the retry count to 2.
- After another 5 seconds, if the server does not receive a response from the client, it sends a heartbeat request and increments the retry count to 3.
- After another 5 seconds, if the server does not receive a response from the client, the connection is reported failed and the socket is closed.

To disable heartbeating, set the HeartbeatIntervalMs value to 0.

A Retry value of 0 causes the connection to time out after the interval without sending any heartbeat.

Device Registry Key

The Device registry key contains one value, SnapshotDelaySec. This is a reserved value that must not be changed.

Peers Registry Key

The Peers registry key informs a CTI OS Server about other CTI OS Servers. This allows CTI OS Servers to make direct connections with one another for the purposes of routing internal messages. On startup, CTIOSServerNode reads this key and opens client connections to all peer servers.



Note You can define two CTI OS Servers as peer servers only if they are connected to the same CTI Server or CTI Server pair. You cannot define two CTI OS Servers as peer servers if they are connected to CTI Servers that reside on different PGs.

The Peers key contains the values listed in following table.

Table 13: Registry Values for [ServerPeers]

Registry Value Name	Value Type	Description	Default
HeartbeatIntervalMs	DWORD Value	Number of milliseconds between heartbeats for client connection to peer servers.	5000
HeartbeatRetrys	DWORD Value	Number of retry attempts before a connection to a peer server is determined to be down.	3

In addition, there must be a subkey for each peer server to which the current server connects. The key name is the hostname or IP address of the peer server; for example:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\CTIOS\

```

Each such subkey must contain the registry value listed in the following table.

Table 14: Registry Values for [ServerPeers] Subkeys

Registry Value Name	Value Type	Description
Port	DWORD Value	The number of the TCP/IP port on which the peer server is listening for the client connection.

Peripherals Registry Key

The Peripherals key stores the maps of valid PeripheralID and Peripheral Types. On CTI OS System startup, these mappings are read into a map, which creates the appropriate peripheral-type objects on the server.

This information must correspond to the Unified UCCE database Peripheral table Peripheral.PeripheralID and Peripheral.ClientType. While the values in ClientType are not equal to the PeripheralTypes, there is a one-to-one relationship between ClientTypes and PeripheralTypes.

The symbol PERIPHERAL_LOGICAL_NAME can be any logical name that uniquely identifies a Peripheral, such as “Phoenix ACD 1.” This is equivalent to the Peripheral.EnterpriseName logical name in the Unified UCCE database. There must be one entry for each valid Peripheral at this site.

The following lists the Peripherals key registry values.

Table 15: Registry Values for [Server\Peripherals\PERIPHERAL_LOGICAL_NAME]

Registry Value Name	Value Type	Description
PeripheralID	DWORD Value	The PeripheralID configured in the Unified UCCE database for this Peripheral.
PeripheralType	DWORD Value	The PeripheralType corresponding to this PeripheralID.

Examples:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\Ctios\

```

SkillGroup Registry Key

The SkillGroup key defines skill group configuration values. The following table lists the SkillGroup key registry values.

Table 16: Registry Values for [Server\SkillGroup]

Registry Value Name	Value Type	Description	Default
PollingInterval Sec	DWORD Value	The SkillGroup statistics polling interval, in seconds.	10

Supervisor Registry Key

The Supervisor key contains supervisor related configuration information. The following table lists the registry values for the Supervisor key.

Table 17: Registry Values for [Server\Supervisor]

Registry Value Name	Value Type	Description	Default
Supervisor ChatLevel	String Value	Defines the call center personnel with whom a supervisor is permitted to chat. This must be set to one of the values listed in the table below.	Unrestricted

Table 18: SupervisorChatLevel Values

Value	Meaning
Disabled	All supervisor chat disabled.
Team	Supervisors can chat with anyone in their primary team.
Unrestricted	Supervisors can chat with anyone on the same peripheral.

ThreadPoolSize Registry Key

ThreadPoolSize is the number of threads in the IO completion port pool.

The ThreadPoolSize registry value is found under the following registry key:

```
HKLM\Software\Cisco
Systems.Inc.\ctios\CTIOS_<instancename>\CTIOS1\Server\ThreadPool
```

Registry Value Name	Value Type	Description	Default
ThreadPoolSize	DWORD Value	If set to ≤ 0 , then the number of threads in the pool are calculated using the following formula: number of CPU's +2. Maximum threads allowed are 32.	0 for all peripheral types except Avaya where the default value is 14.



Note Balancing threads against overall performance is not a trivial task. If the ThreadPoolSize value is changed, follow up with overall performance monitoring to see whether CTI OS Server performance is affected.

TimerService Registry Key

The TimerService key specifies configuration parameters for the CTI OS Server's internal TimerService. The following table lists the registry values for the TimerService key.

Table 19: Registry Values for [Server\TimerService]

Registry Value Name	Value Type	Description	Default
ResolutionMSec	DWORD Value	The interval at which the TimerService services queued requests, expressed in milliseconds.	500

MainScreen Registry Key

The MainScreen key, located at [HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\<CTIOSInstanceName>\<CTIOSServerName>\EnterpriseDesktopSettings\All Desktops\ScreenPreferences\Name\MainScreen], includes registry values that define the behavior of softphone windows and icons in response to a BeginCallEvent. The following table lists the registry values for the MainScreen key.

Table 20: MainScreen Registry Key Values

Registry Value Name	Value Type	Description	Default
BringToFrontOnCall	DWORD Value	When enabled (1), the softphone window is raised above all other windows when a BeginCallEvent occurs.	1
FlashOnCall	DWORD Value	When enabled (1), the softphone icon on the taskbar flashes when a BeginCallEvent occurs.	0
RecordingEnabled	DWORD Value	Controls whether the Record button is enabled on the Agent and Supervisor Softphones (0 = disabled, 1 = enabled).	0
AgentStatisticsIntervalSec	DWORD Value	Controls how often (in seconds) the Agent and Supervisor Softphones update time-in-state agent statistics.	0xF

Unified CCE Silent Monitor Configuration

The IPCCSilentMonitor key contains silent monitor configuration information. The IPCCSilentMonitor key contains one subkey, named Settings.

The IPCCSilentMonitor configuration settings are declared in the registry of each server on the following location:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems Inc.\CTIOS\<CTIOSInstanceName>\<CTIOSServerName>\EnterpriseDesktopSettings\All Desktops\IPCCSilentMonitor\Name\Settings]
```

The Settings subkey contains the parameters used by the silent monitor subsystem to establish a monitoring session between a supervisor and a monitored agent. The values are listed in the following table.

Table 21: Settings Registry Subkey Values

Registry Value Name	Value Type	Description	Default
HeartbeatInterval	DWORD value	The time in seconds between consecutive heartbeats.	5

Registry Value Name	Value Type	Description	Default
HeartbeatTimeout	DWORD value	The amount of time in seconds that must elapse without receiving data before a disconnect is signaled.	15
MediaTerminationPort	DWORD value	Reserved. This is the TCP/IP port that the silent monitor subsystem uses to render monitored audio.	4000
MonitoringIPPort	DWORD value	This is the TCP/IP port on the monitoring application to which the monitored application sends monitored audio.	39200
StopSMNonACDCall	DWORD value	This stops silent monitoring of Non-ACD calls. When enabled (1) in Unified CM-based silent monitoring, the supervisor's monitor button is disabled. When enabled in desktop-based silent monitoring, the supervisor's monitor button is enabled but the supervisor can only hear ACD calls. Note You must manually enter this value into the registry. If the value has not been entered into the registry, the effect is the same as having it set to its default (0).	0

ConnectionProfiles Registry Key

The ConnectionProfiles key contains an organized list of the connection information of all configured CTI OS Servers present in the corporate network that you can access by a client application. The connection profiles are defined in the registry of each server at the following location:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems Inc.\CtiOs\CTIOS InstanceName\<CTIOSServerName>\ EnterpriseDesktopSettings\All Desktops\Login\ConnectionProfiles\Name\CtiOsProfileName
```

To create a profile for a given server, you must define a subkey under ConnectionProfiles\Name with the following format:

```
[HKEY_LOCAL_MACHINE\Software\...\ConnectionProfiles\Name\CtiOsProfileName]
"PeripheralID"=dword:5000
"Heartbeat"=dword:00000000
"MaxHeartbeats"=dword:00000005
"CtiOsA"="HostName_A"
"CtiOsB"="HostName_B"
"PortA"=dword:0000a42c
"PortB"=dword:0000a42c
"AutoLogin"=dword:00000001
"ShowFieldBitMask"=dword:00000023
"WarnIfAlreadyLoggedIn"=dword:00000001
"RejectIfAlreadyLoggedIn"=dword:00000000
```

```
"DisableSkillGroupStatistics"=dword:00000001
"DisableAgentStatistics"=dword:00000001
"UCCESilentMonitorEnabled"=dword:0x00000001
"WarnIfSilentMonitored"=0x00000000
```

The following table describes the required ConnectionProfiles key registry values.

Table 22: ConnectionProfiles Key Registry Values

SubKey/Value	Description
CtiOsProfileName	The name given to the profile. This string appears on the Login Dialog when a user is about to log in using the CTI OS Agent State Control.
PeripheralID	The numeric value of the peripheral to which the CTI OS Server connects.
Heartbeat	Time interval between heartbeat messages between the client and CTI OS Server.
MaxHeartbeats	Maximum number of heartbeats that can be missed by the CTI OS Client Session before failover occurs.
CtiOsA	DNS name of IP Address of the primary CTI OS Server to which a client application can connect.
CtiOsB	DNS name of IP Address of the secondary CTI OS Server to which a client application can connect.
PortA	TCP/IP port number assigned to the primary server.
PortB	TCP/IP port number assigned to the secondary server.
AutoLogin	Indicates if the client must automatically log in an agent or supervisor after it recovers from a system failure. For all peripherals other than UCCE you must set this field to 0x00000000. For UCCE, set this field to 0x00000001.
ShowFieldBit Mask	Indicates what fields appear in the CTI OS Login dialog box. Fields appear on the dialog box only if their corresponding bit in the mask is on. The possible fields and their corresponding masks are shown in the table “ShowBitFieldMask Fields” below. The default value at setup for ShowFieldBit Mask is 0x00000023 (AgentID, Instrument, and Password displayed).
WarnIfAlready LoggedIn	Indicates whether to display a warning but still permit login if an agent who is already logged in attempts to log in again. A value of 1 (default) enables the warning; a value of 0 disables the warning. This value is relevant only if RejectIfAlreadyLoggedIn is 0.
RejectIfAlready LoggedIn	Indicates whether or not to permit an agent who is already logged in to log in again. A value of 0 (default) permits an agent to log in again. A value of 1 prohibits an agent from logging in again.
DisableSkillGroup Statistics	Indicates whether skill group statistics are enabled for the agent using this connection profile. A value of 1 disables statistics. If this value is 0 (default) or not present, skill group statistics are enabled for this agent.

SubKey/Value	Description
DisableAgent Statistics	Indicates whether agent statistics are enabled for the agent using this connection profile. A value of 1 disables statistics. If this value is 0 (default) or not present, statistics are enabled for this agent.
IPCCSilent MonitorEnabled	Indicates whether silent monitor is enabled for the clients using this connection profile. A value of 0x00000001 (default) enables silent monitor. If this value is 0x00000000 or not present, silent monitor is disabled for this client. For all peripherals other than UCCE, you must set this field to 0x00000000.
WarnIfSilent Monitored	Indicates whether to display an indicator on the agent desktop when the agent is silent monitored by the team supervisor. A value of 0x00000001 causes a message to appear on the agent desktop when the supervisor is silent monitoring this agent. If this value is 0x00000000 (default) or not present, no message appears on the agent desktop when the supervisor is silent monitoring this agent.
RasCallMode	Indicates the agent work mode options for the mobile agent login dialog box. Valid values are 0 (agent chooses), 1 (call by call), and 2 (nailed up).

Table 23: ShowBitFieldMask Fields

Field	Mask
Instrument	0x00000001
Password	0x00000002
Work Mode	0x00000004
Position ID	0x00000008
Skillgroup	0x00000010
AgentID	0x00000020
Login Name	0x00000040
Mobile Agent	0x00000080

The heartbeating mechanism uses the MaxHeartbeats and Heartbeat values together to determine when a client must send heartbeat requests to the server and when the client must connect to the other server.

MaxHeartbeats is the max number of missed heartbeats before failover. (Default = 5)

Heartbeat is the time interval between consecutive heartbeats. (Default = 5)

This is how the heartbeating mechanism works on the CTI OS client:

- After 5 seconds, if the client does not receive a response from the server, it sends a heartbeat request 1.
- After 5 seconds, if the client does not receive a response from the server, it sends a heartbeat request 2.
- After another 5 seconds, if the client does not receive a response from the server, it sends a heartbeat request 3.
- After yet another 5 seconds, if the client does not receive any response from the server, it connects to an alternative server.



Note The amount of time it takes a client to reconnect to the other server depends on the type of failure that occurs.

The heartbeat parameters above are only a factor if the TCP/IP socket is not broken. For example, if you disconnect the network cable to the CTI OS Server, TCP/IP does not break the socket. In this case, the client uses the heartbeating mechanism listed above to detect the failure.

In a different case, however, if the CTI OS Server process crashes or the machine is turned off, the socket breaks and the client immediately knows that the connection has failed. In this case, the client directly connects to the other server without heartbeat attempts.



Note In either case, although the socket connection might get established right away, it might take a few more seconds for the agents to fully recover their previous, pre-failure state. This delay might particularly be experienced if many agents are failing over at the same time, or if the system is experiencing a heavy call load at the time of the failure.

SilentMonitorService Subkey

The ConnectionProfiles key contains a <profile_name>\SilentMonitorService subkey, which contains parameters that clients use to connect to one of a set of silent monitor services. It contains the following keywords.

Table 24: ConnectionProfiles\<profile_name>\SilentMonitorService Subkey Values

Registry Value Name	Value Type	Description
ListenPort	integer	Port on which the silent monitor service is listening for incoming connections.
TOS	integer	QOS setting for the connection.
HeartbeatInterval	integer	Amount of time in milliseconds between heartbeats.
HeartbeatRetries	integer	Number of missed heartbeats before the connection is abandoned.
Cluster		<p>A key that contains a list of silent monitor services to which the CIL tries to connect. The CIL randomly chooses one of the services in this list. This key contains the following subkeys:</p> <ul style="list-style-type: none"> • 0 – Index of the first silent monitor service • N – Index of the Nth silent monitor service <p>Both subkeys contain the following keyword.</p> <p>SilentMonitorService – hostname or IP address of a silent monitor service to which to connect.</p>

Configuration of Additional Connection Profiles

Creation of Second Profile

Use the following template to create a connection profile that includes a silent monitor server:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<instance>\CTIOS1\EnterpriseDesktopSettings\All
Desktops\Login\ConnectionProfiles\Name\<profileName>]

    "peripheralID"=dword:00001389
    "ShowFieldBitMask"=dword:000000a3
    "SwitchCapabilityBitMask"=dword:7f3f1bff
    "CtiosA"="ctios-a"
    "PortA"=dword:0000a42c
    "UCCESilentMonitorEnabled"=dword:00000001
    "WarnIfSilentMonitored"=dword:00000000
    "CtiosB"="ctios-b"
    "PortB"=dword:0000a42c
    "MaxHeartbeats"=dword:00000003
    "Heartbeat"=dword:00000005
    "AutoLogin"=dword:00000001
    "WarnIfAlreadyLoggedIn"=dword:00000000
    "RejectIfAlreadyLoggedIn"=dword:00000000
    "TOS"=dword:00000000
    "RasCallMode"=dword:00000000

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<instance>\CTIOS1\EnterpriseDesktopSettings\All
Desktops\Login\ConnectionProfiles\Name\<profileName>\SilentMonitorService]

    "HeartbeatInterval"=dword:00001388
    "HeartbeatRetries"=dword:00000005
    "ListenPort"=dword:0000a42d
    "TOS"=dword:00000000

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<instance>\CTIOS1\EnterpriseDesktopSettings\All
Desktops\Login\ConnectionProfiles\Name\<profileName>\SilentMonitorService\Cluster]

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<instance>\CTIOS1\EnterpriseDesktopSettings\All
Desktops\Login\ConnectionProfiles\Name\<profileName>\SilentMonitorService\Cluster\0]

    "SilentMonitorService"="sms-host-or-ip"
```



Note The SilentMonitorService key is not always present.

When the SilentMonitorService key is present, the agent desktop attempts to connect to the silent monitor service running on the host specified in the key.

When the SilentMonitorService key is not present, the agent desktop connects to a silent monitor service running locally (on the same computer as the agent desktop).

Two Profiles for Server- and Desktop-Based Silent Monitoring Scenario

If no silent monitor key exists in the connection profile, the profile defaults to desktop silent monitoring. The following template illustrates two connection profiles—one for desktop-based silent monitor, and one for server-based silent monitor:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<instance>\CTIOS1\EnterpriseDesktopSettings\All
Desktops\Login\ConnectionProfiles\Name\UCCE]
```

```
"peripheralID"=dword:00001388
"ShowFieldBitMask"=dword:00000023
"SwitchCapabilityBitMask"=dword:7f3f1bff
"CtiosA"="ctios-a"
"PortA"=dword:0000a42c
"UCCESilentMonitorEnabled"=dword:00000001
"WarnIfSilentMonitored"=dword:00000001
"CtiosB"="ctios-b"
"PortB"=dword:0000a42c
"MaxHeartbeats"=dword:00000003
"Heartbeat"=dword:00000005
"AutoLogin"=dword:00000001
"WarnIfAlreadyLoggedIn"=dword:00000000
"RejectIfAlreadyLoggedIn"=dword:00000000
"TOS"=dword:00000000
"SaveShowField"=dword:00000043
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<instance>\CTIOS1\EnterpriseDesktopSettings\All
Desktops\Login\ConnectionProfiles\Name\Mobile Agent]
```

```
"peripheralID"=dword:00001388
"ShowFieldBitMask"=dword:000000a3
"SwitchCapabilityBitMask"=dword:7f3f1bff
"CtiosA"="ctios-a"
"PortA"=dword:0000a42c
"UCCESilentMonitorEnabled"=dword:00000001
"WarnIfSilentMonitored"=dword:00000000
"CtiosB"="ctios-b"
"PortB"=dword:0000a42c
"MaxHeartbeats"=dword:00000003
"Heartbeat"=dword:00000005
"AutoLogin"=dword:00000001
"WarnIfAlreadyLoggedIn"=dword:00000000
"RejectIfAlreadyLoggedIn"=dword:00000000
"TOS"=dword:00000000
"RasCallMode"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<instance>\CTIOS1\EnterpriseDesktopSettings\All
Desktops\Login\ConnectionProfiles\Name\Mobile Agent\SilentMonitorService]
```

```
"HeartbeatInterval"=dword:00001388
"HeartbeatRetries"=dword:00000005
"ListenPort"=dword:0000a42d
"TOS"=dword:00000000
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<instance>\CTIOS1\EnterpriseDesktopSettings\All
```



```
Desktops\Login\ConnectionProfiles\Name\Mobile
Agent\SilentMonitorService\Cluster]

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\CTIOS_<instance>\CTIOS1\EnterpriseDesktopSettings\All
Desktops\Login\ConnectionProfiles\Name\Mobile
Agent\SilentMonitorService\Cluster\0]

"SilentMonitorService"="sms-host-or-ip"
```

Call Appearance Grid Configuration

The CallAppearance key contains a list of all the columns that appear on the softphone Call Appearance grid.

The columns are declared in the registry of each server on the following location:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\Ctios\<CTIOS
InstanceName>\<CTIOSServerName>\EnterpriseDesktopSettings\ All
Desktops\Grid\CallAppearance\Columns\Number\Position]
```

Position represents the actual location in the grid where the column appears. For example for the first column Position is “1” and for the fifth column it is “5”.

The following table lists the attributes that a column declaration can contain.

Table 25: Column Declaration Attributes

Attribute	Type	Description
Type	String Value	Assigns a column to display the Call information identified by the value of this attribute. The “Type values” table below lists the possible values.
Header	String Value	Contains the text string that appears on the header of the column. If not specified, the Type appears instead.
Width	DWORD value	Column width expressed in pixels. If the Auto Resize Columns property is set on the Call Appearance Grid, this attribute has no effect. The column is automatically sized to match the column header or column cell content, whichever is longer. If the Auto Resize Columns property is not set, one of the following occurs: <ul style="list-style-type: none"> • If Width is specified, the column sizes to match it. • If Width is <i>not</i> specified, the column sizes to a default length.
MaxChars	String Value	Maximum number of characters that can appear in the column.

Attribute	Type	Description
Name	String Value	Used only when the Type is ECC; contains the name of a given ECC variable. The name in this attribute must be entered without the prefix “ user .” For the standard Outbound Option ECC variables, use the prefix BA without any dots following it; for example, BAResponse .
Alignment	String Value	Defines the alignment of the information on the columns. Possible values are “left”, “right” or “centered.”
NumericOnly	String Value	If “true” the column accepts only numeric values for display. If “false” alphanumeric values may appear.
editable	String Value	Indicates if the user can modify the cells on the column at runtime.

The following table lists the Type values.

Table 26: Type Values

Type	Description
CallID	Associates the column with the unique call ID.
CallStatus or Status	Associates the column with Call Status.
DNIS	Associates the column with DNIS.
ANI	Associates the column with ANI.
CED	Associates the column with the caller entered digits.
DialedNumber or DN	Associates the column with the dialed number.
UserToUserInfo or UserToUser	Associates the column with user to user information.
WrapUp	Associates the column with the call wrap up data.
Var1, Var2, ..., Var10	Associates the column with a call variable.
NAMEDVARIABLE, ECCVariable, ECCVar, ECC, or ECCNAME	Associates the column with an scalar ECC Variable.
NAMEDARRAY or ECCARRAY	Associates the column with a Named Array ECC variable.
CampaignID	Campaign ID for value appears in the Agent Real Time table. Set to zero if not used. <i>Applicable only to Outbound Option systems.</i>
QueryRuleID	Query rule ID for value appears in the Agent Real Time table. Set to zero if not used. <i>Applicable only to Outbound Option systems.</i>

The following are examples of column declarations:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CtiOs\

```

```
"Type"="CallID"
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CtiOs\

```

```
"Type"="Var2"
"editable"="true"
```

The following is an example of associating a column with an ECC variable:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CtiOs\

```

```
"Type"="ECC"
"Name"="bobc"
"Header"="ECC Bobc"
"Maxchars"="8"
"editable"="true"
```

The following is an example of associating a column with an ECC array variable. Note that the “Name” key must contain both the array name and the subscript/index:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CtiOs\

```

```
"Type"="ECCARRAY"
"Name"="bobc[0]"
"Header"="ECCARRAY Bobc"
"Maxchars"="8"
"editable"="true"
```

Configure Automatic Call Appearance Grid

The CTIOSServer directory contains a file, `callappearance.default.reg.txt`, which provides the following default definition for Call Appearance grid columns 1 to 18:

```
REGEDIT4
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\Columns]
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number]
```

```

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\Columns\ Number\1]
"Type"="CallID"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\Columns\ Number\10]
"Type"="Var2"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\Columns\ Number\11]
"Type"="Var3"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\12]
"Type"="Var4"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\Columns\ Number\13]
"Type"="Var5"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\14]
"Type"="Var6"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\15]
"Type"="Var7"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\16]
"Type"="Var8"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\17]
"Type"="Var9"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\

```

```
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\18]
"Type"="Var10"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\2]
"Type"="CallStatus"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\3]
"Type"="DNIS"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\4]
"Type"="ANI"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\5]
"Type"="CED"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\6]
"Type"="DialedNumber"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\7]
"Type"="UserToUserInfo"
"maxchars"="129"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\8]
"Type"="WrapUp"
"maxchars"="40"
"editable"="true"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\CallAppearance\ Columns\Number\9]
"Type"="Var1"
"maxchars"="40"
"editable"="true"
```

To import this default definition into your registry, perform the following steps.

Procedure

-
- Step 1** Choose **Start > Run** dialog box.
- Step 2** Rename the callappearance.default.reg.txt file to callappearance.default.reg.
- Step 3** Enter **regedit filename**
- where *filename* is the *full pathname* of the callappearance.default.reg file.

Step 4 Cycle your CTI OS Server process.

Related Topics

[Unified CCE Service Control](#), on page 89

Customize Agent Statistics Grid Configuration

The CTIOSServer directory contains a file, agentstatistics.default.reg.txt, that contains the default definition for the Agent Statistics grid. The following is an example agentstatistics.default.reg.txt file that defines Agent Statistic grid columns 1 and 2:

```
REGEDIT4

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid]

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\AgentStatistics]

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\AgentStatistics\Columns]

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\AgentStatistics\ Columns\Number]
"DisableStatsMinimization"=dword:00000000

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\AgentStatistics\ Columns\Number\1]
"Type"="CallsHandledToday"
"Header"="CallsHandledToday"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\AgentStatistics\Columns\Number\2]
"Type"="TimeLoggedInToday"
"Header"="TimeLoggedInToday"
```

The DisableStatsMinimization registry value controls the quantity of agent statistics that are sent from the CTI OS Server to CTI OS clients. Possible values are 0 (only those agent statistics that are configured to be displayed on the agent statistics grid are sent to the client) and 1 (all agent statistics are sent to the client); default is 0.

To customize the Agent Statistics grid, perform the following steps.

Procedure

- Step 1** Make a copy of the agentstatistics.default.reg.txt file.
- Step 2** Rename the copied agentstatistics.default.reg.txt file to agentstatistics.default.reg.
- Step 3** Add, remove, and renumber column definitions *in the copied file* as desired.

- Step 4** Choose **Start > Run** dialog box.
- Step 5** Enter **regedit filename**
where *filename* is the *full pathname* of the edited copy of the agentstatistics.default.reg file.
- Step 6** Cycle your CTI OS Server process.

Related Topics

[Unified CCE Service Control](#), on page 89

Automatic Skill Group Statistics Grid Configuration

The CTIOSServer directory contains a file, skillgroupstatistics.default.reg.txt, that contains the default definition for the Skill Group Statistics grid. The following is an example skillgroupstatistics.default.reg.txt file that defines columns 1 through 4:



Note The first column of the Skill Group Statistics window should be SkillGroupName.

```
REGEDIT4

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid]

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\SkillGroupStatistics]

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\SkillGroupStatistics\Columns]

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\SkillGroupStatistics\Columns\Number]
"DisableStatsMinimization"=dword:00000000
"DisableMonitorModeStatsMinimization"=dword:00000000

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\SkillGroupStatistics\Columns\Number\1]
"Type"="SkillGroupName"
"header"="SkillGroupName"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\SkillGroupStatistics\Columns\Number\2]
"Type"="AgentsAvail"

[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\
<CTIOS InstanceName>\<CTIOSServerName>\
EnterpriseDesktopSettings\All Desktops\Grid\SkillGroupStatistics\Columns\Number\3]
"Type"="AgentsNotReady"
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\  
<CTIOS_InstanceName>\<CTIOSServerName>\  
EnterpriseDesktopSettings\All Desktops\Grid\SkillGroupStatistics\Columns\Number\4]  
"Type"="AgentsReady"
```

The DisableStatsMinimization registry value controls the quantity of skill group statistics that are sent from the CTI OS Server to CTI OS agent mode clients. Possible values are 0 (only those skill group statistics that are configured to appear on the skill group statistics grid are sent to the client) and 1 (all skill group statistics are sent to the client); default is 0.

The DisableMonitorModeStatsMinimization registry value controls the quantity of skill group statistics that are sent from the CTI OS Server to CTI OS monitor mode clients. Possible values are 0 (only those skill group statistics that are configured to appear on the skill group statistics grid are sent to the client) and 1 (all skill group statistics are sent to the client); default is 0.



Note When viewing CTIOS with the Supervisors, the default skill group shows up on the CTIOS Agent Skill Group stats. This default skill group reports all voice calls not routed by a Unified UCCE script.



Note While you can customize columns in the **Skill Group Statistics** grid, you should retain the following registry settings:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\CTIOS\  
InstanceName>\<CTIOSServerName>\EnterpriseDesktopSettings\All  
Desktops\Grid\SkillGroupStatistics\Columns\Number\1]
```

```
"Type"="SkillGroupNumber"  
"header"="SkillGroupNumber"
```

The header can vary depending on the language you use. To customize the Skill Group Statistics grid, perform the following steps.

Procedure

- Step 1** Make a copy of the skillgroupstatistics.default.reg.txt file.
 - Step 2** Rename the copied skillgroupstatistics.default.reg.txt file to skillgroupstatistics.default.reg.
 - Step 3** Add, remove, and renumber column definitions *in the copied file* as desired.
 - Step 4** Open the Windows **Start > Run** dialog box.
 - Step 5** Enter **regedit filename**
where *filename* is the full pathname of the edited copy of the skillgroupstatistics.default.reg file.
 - Step 6** Cycle your CTI OS Server process.
-

Related Topics

[Unified CCE Service Control](#), on page 89

Configure Additional Peripherals

The Peripheral Identifier screen in CTI OS Server setup lets you supply peripheral information for a single peripheral only. To configure additional peripherals, perform the following steps.

Procedure

-
- Step 1** Define a registry key for the peripheral in [Server\Peripherals\ PERIPHERAL_LOGICAL_NAME].
- Step 2** Create a connection profile for the peripheral.
-



Note The value that you specify for Peripheral ID in the Peripherals registry key definition *must* match the value that you specify for Peripheral ID in the connection profile definition.

Related Topics

- [Peripherals Registry Key](#), on page 67
- [ConnectionProfiles Registry Key](#), on page 71

Quality of Service/Type of Service

CTI OS supports “Type of Service” ToS.

The following connections/components support Qos/ToS:

- CTI OS Server to CTI OS Client.
- CTI OS Client (C++ CIL only) to CTI OS Server.

For CTI OS, TOS tagging is not implemented in the Java or .NET (C#) CILs. As stated above, a system using these could support one-way tagging from server to client, but traffic from the client to the server is sent on a best-effort basis.

CTI OS supports the marking of TCP/IP packets with ToS. This allows for preferential treatment (for example, class AF31 for assured forwarding) of CTI signaling traffic if the network is configured to support this QoS scheme.

By default, CTI OS does not mark packets, which means that the traffic is sent with "best effort" (ToS = 0).

To turn on the ToS markings, you must configure certain registry keys. In general, ToS effects only outgoing packets. For example, the CTI OS Server can send packets with ToS markings for assured forwarding to CTI OS clients. However, that does not imply that CTI OS clients must also send their network traffic with the same ToS value to the CTI OS Server. CTI OS clients could in fact send their traffic on a best-effort basis, which would mean that ToS is only active one way. Most likely, though, ToS is configured the same for both directions.

Basic Configuration

To turn on ToS with AF31 for bidirectional communications, add/modify some registry keys for CTI OS Server.

1. The following key turns on marking of packets CTI OS Server sends to CTI OS clients:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\\CTIOS1\Server\Connections
"TOS"=dword:00000068
```



Note The dword value above is listed in hexadecimal format (decimal 104).

2. This registry key turns on markings of packets sent from the client to the server:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\\CTIOS1\EnterpriseDesktopSettings\All
Desktops\Login\ConnectionProfiles\Name\UCCE<or other profile name>
"TOS"=dword:00000068
```

3. This key turns on TOS marking for Silent Monitor packets.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems,
Inc.\Ctios\\CTIOS1\EnterpriseDesktopSettings\All
Desktops\UCCESilentMonitor\Name\Settings "TOS"=dword:000000B8
```



Note Use a different class (real-time/voice) with a different TOS value (Hex B8) for a silent monitor stream.

Important Additional Configuration Information

Ensure the following registry key in Windows Server is as follows:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\TcpIp\Parameters
"DisableUserTOSSetting"=dword:00000000
```

Caveats

- For the ToS to become effective, you must configure the network (specifically the routers) to treat packets with ToS markings preferentially.
- The traffic between the CTI OS Server and CTI OS clients may include types of data that do not qualify for AF31 type of service. In general, use AF31 for signaling traffic. For example, a call delivered event sent from the CTI OS Server is time critical, as is a potential Answer request in response sent from the client in order to answer an alerting call. However, the CTI OS Server can also send statistics to clients. AF31 is not appropriate for this type of data. However, because CTI OS sends all traffic on the same connection, either all packets are marked or none. Therefore, you must turn off CTI OS Skillgroup statistics with TOS enabled.

- When hardphones are used with silent monitoring, the switch in the phone overrides the TOS marking to 0. This affects both silent monitor and CTI OS client to CTI OS Server traffic. (It does not affect CTI OS Server to CTI OS client traffic.) To correct this problem, write ACL to classify traffic based on TCP/UDP port number from the endpoint.



CHAPTER 9

Startup, Shutdown, and Failover

- [Unified CCE Service Control, on page 89](#)
- [CTI OS Failover, on page 90](#)

Unified CCE Service Control

The Unified CCE Service Control application is an interface into the Windows platform's service control manager, which starts and stops services.

To start, stop, or cycle the processes in the CTI OS Server, use the appropriate tabs from the Unified CCE Service Control window. To set CTI OS to start automatically on Windows startup, select the service name and click the Automatic button.



Note CTI OS is not displayed in the Service Manager in the ICM Websetup page.

When the CTI OS service starts, it launches processes listed in the following table.

Table 27: CTI OS System Processes

Process Name	Process Description	Runs In Console Window
CtiosServerNode	The main CTI OS Server process. This process manages all CTI OS objects and listens for and manages client connections.	Yes
CTIOSTrace	The CTI OS tracing utility. This process uses the Unified ICM Event Management System (EMS) to trace server messages to local log files in EMS format.	No
NM	The Unified ICM NodeManager (fault tolerance manager). Each Unified ICM service is started by NodeManager, and NodeManager restarts any abnormally terminated processes.	No

Process Name	Process Description	Runs In Console Window
NMM	The Unified ICM NodeManagerManager (system fault tolerance). Each Unified ICM Node (e.g. CTI OS) starts up a NMM process to handle system-level faults. In the event of a unrecoverable system fault, NMM restarts the host computer.	No

CTI OS Failover

The server processes are managed by a fault tolerance/recovery platform called NodeManager. NodeManager creates and monitors each process running as part of the CTI OS service, and automatically restarts abnormally terminated processes.

Failover of CTI OS Related Components

CTI OS handles failover of related components as described in the following sections.



Caution

The CTI OS desktop can buffer actions if an agent clicks buttons during a failover. Those actions can then take effect when the failover completes. You should warn agents not to click desktop buttons during a failover.

IP Phones

If an IP phone goes out of service, CTI OS sends an event to all soft phones associated with the IP phone that their IP phone is out of service. In addition, the affected softphones display the message “Offline.” When the IP phone is back in service, agents must manually log in.

Switches

If a switch goes out of service, CTI OS sends an event to all softphones associated with the switch that the switch is offline. In addition, the affected softphones display the message “Offline.” When the switch is back in service, agents must manually sign in.

Peripheral Gateway

Because the Peripheral Gateway (PG) is a fault-tolerant process pair, CTI OS is not affected if the PG merely switches active sides. If the PG goes offline, CTI OS sends an “Offline” message to each softphone client.

CTI Server Failure

On a CTI Server failure, the CTI OS Server usually reconnects almost immediately to the redundant CTI Server. If reconnection to the redundant CTI Server is not possible, the CTI OS Server sends a failure response to any requests made to the CTI Server.

In addition, CTI OS sends an event message to all softphone clients. On receipt of this message, the softphone clients display an “Offline” message.

When the CTI Server comes back online, CTI OS performs a snapshot of all agents, devices, and calls to reestablish state information.

CTI OS Server Failure

On a CTI OS Server failure, CTI OS disconnects all softphones from the failed CTI OS Server. These softphones attempt to reconnect automatically to another CTI OS Server; if reconnection is not possible, CTI OS sends an event message to all softphone clients. On receipt of this message, the softphone clients display an “Offline” message.

NodeManager restarts the CTI OS Server. When the CTI OS Server process comes back online, CTI OS performs a snapshot of all agents, devices, and calls to reestablish state information.



CHAPTER 10

Peripheral-Specific Support

- [TDM peripherals, on page 93](#)
- [General Unified ICM Support, on page 93](#)
- [CTI OS Support, on page 95](#)

TDM peripherals

Different peripheral manufacturers provide varying levels of support for CTI specific features. You must take these differences into account when writing a CTI OS client application. As far as possible, the CTI OS Server and agent desktop simulate the hardphone behavior of the peripheral in question. The CTI OS Supervisor Desktop for Unified CCE is specific to Unified CCE and is currently not supported on the TDM switches because they do not, in general, provide the Supervisory features that Unified CCE provides.



Note The peripherals mentioned in this chapter are the ones that CTI OS supports. Please contact Cisco CTI Product Management if you are interested in CTI OS support for a peripheral not mentioned here.

This chapter provides the following information:

- Peripheral-specific equivalents for some common Unified ICM terms
- A list of Unified ICM features that some peripherals do not support
- A table of CTI call event types that are unavailable for different peripheral types
- A table of CTI OS client control requests that are unsupported by different peripheral types
- Differences and limitations in the level of CTI support provided by various peripherals—including a list of CTI Server agent states and the corresponding terminology/functionality associated with the various peripherals

General Unified ICM Support

This section describes differences in how various peripherals implement Unified ICM functionality.

Peripheral-Specific Terminology

Different peripheral manufacturers use different terminology for Unified ICM terms such as agents, skill groups, and services. For example, other manufacturers might call a service an application, a split, or a gate. The following table lists several Unified ICM terms and provides peripheral-specific equivalents.

Table 28: Unified ICM and Peripheral-Specific Terminology

Unified ICM Term	Peripheral-Specific Equivalent
Agent	Agent
Peripheral target	Unified CCE: Agent Target Others: Trunk group and DNIS ¹
Service	Aspect Contact Server: Application Avaya DEFINITY ECS: Vector Directory Number (VDN) Avaya Aura CC (Symposium): Application
Skill group	Aspect Contact Server: Agent group Avaya DEFINITY ECS: Skill group or hunt group ² Avaya Aura CC (Symposium): Skill Set Others: Skill group
Trunk	Aspect Contact Server: Instrument ³ Avaya Aura CC (Symposium): None Others: Trunk
Trunk group	Avaya Aura CC (Symposium): Route Others: Trunk group

¹ The Aspect Contact Server maps a trunk group and DNIS to a Call Control Table (CCT). The DEFINITY ECS uses the trunk group and DNIS for incoming calls.

² If an ECS is running in Expert Agent Selection (EAS) mode, a skill group maps to an ECS skill group; otherwise, it maps to a hunt group.

³ A CallCenter instrument can be a trunk, a teleset, or a workstation.

In some cases, the Unified ICM concept is very close to the corresponding ACD feature. For example, the Unified ICM concept of a service is very similar to the Aspect concept of an application. In other cases, the ACD does not have a feature that maps exactly to the Unified ICM feature. In these cases, you might choose a different mapping than shown in the table above. For example, although it might make sense to associate each VDN on a DEFINITY ECS with an Unified ICM service, you could also map each hunt group to a service.

On an Avaya DEFINITY ECS running in EAS mode, each skill group may have multiple subgroups depending on the switch configuration. Unified ICM emulates this by automatically creating additional skill groups for these peripheral types.

Unified ICM Feature Limitations

Some ACDs have limitations that prevent them from making full use of specific features of Unified ICM. The following table summarizes these limitations for those ACDs.

Table 29: Unified ICM Features Not Supported for Specific Peripherals

Peripheral Type	Restrictions
Aspect Contact Server	Only one skill group assignment per agent
Avaya DEFINITY ECS	None
Unified CCE System PG	Does not support Trunks or Trunk Groups
Avaya Aura CC (Symposium)	No Peripheral Service Level reporting No Trunk Group Real Time or Trunk Group Half Hour data elements

CTI OS Support

This section describes how different peripheral types implement and support CTI OS functionality. It includes the following information:

- A table of call event types that are unavailable for different peripheral types
- A table of client control requests that are unsupported by different peripheral types
- A list of other peripheral-specific differences and limitations
- A table of agent states

Call Events

The following table lists the call events that are not available from different peripheral types:

- The entry “none” indicates that the event is available from all supported peripherals.
- A single asterisk (*) indicates that the event is available from the starred peripheral, subject to the restrictions/limitations listed in the [Peripheral-Specific Limitations and Differences, on page 97](#).
- A double asterisk (**) indicates that the event is available from Aspect when the PG is configured to use the Aspect Event Link.

Table 30: Call Events Not Available to Specific Peripherals

Unavailable Event	Peripherals
AGENT_PRE_CALL	Aspect, DEFINITY, Avaya Aura CC (Symposium), IVR
AGENT_PRE_CALL_ABORT	Aspect, DEFINITY, Avaya Aura CC (Symposium), IVR

Unavailable Event	Peripherals
AGENT_STATE	None
BEGIN_CALL	None
CALL_CLEARED	Aspect*
CALL_CONFERENCED	Aspect**,IVR
CALL_CONNECTION_CLEARED	None
CALL_DATA_UPDATE	None
CALL_DELIVERED	Aspect*
CALL_DEQUEUED	DEFINITY, Avaya Aura CC (Symposium), Unified CCE, IVR
CALL_DIVERTED	Aspect, Unified CCE, Avaya Aura CC (Symposium)
CALL_ESTABLISHED	IVR
CALL_FAILED	Aspect, Avaya Aura CC (Symposium), IVR
CALL_HELD	Aspect**, IVR
CALL_ORIGINATED	Aspect, DEFINITY*, Avaya Aura CC (Symposium)
CALL_QUEUED	Unified CCE, IVR
CALL_REACHED_NETWORK	Aspect, Avaya Aura CC (Symposium), IVR
CALL_RETRIEVED	Aspect**, IVR
CALL_SERVICE_INITIATED	Aspect**, DEFINITY*, IVR
CALL_TRANSFERRED	IVR
CALL_TRANSLATION_ROUTE	Unified CCE
END_CALL	None
RTP_STARTED_EVENT	Aspect, Avaya Aura CC (Symposium), IVR
RTP_STOPPED_EVENT	Aspect, Avaya Aura CC (Symposium), IVR
SYSTEM	None

Client Control Requests

The following table lists the client control requests that are not supported by the different peripheral types.

Table 31: Client Control Requests Not Available to Specific Peripherals

Unavailable Request	Peripherals
ALTERNATE_CALL	Avaya Aura CC (Symposium)
ANSWER_CALL	IVR
CLEAR_CALL	IVR
CLEAR_CONNECTION	IVR
CONFERENCE_CALL	IVR
CONSULTATION_CALL	IVR
DEFLECT_CALL	Aspect, Avaya Aura CC (Symposium), IVR
HOLD_CALL	IVR
MAKE_CALL	IVR
MAKE_PREDICTIVE_CALL	IVR
QUERY_AGENT_STATE	IVR
QUERY_DEVICE_INFO	IVR
RECONNECT_CALL	IVR
RETRIEVE_CALL	IVR
SEND_DTMF_SIGNAL	Aspect, Avaya Aura CC (Symposium), IVR
SET_AGENT_STATE	IVR
SNAPSHOT_CALL	IVR
SNAPSHOT_DEVICE	IVR
TRANSFER_CALL	IVR

Peripheral-Specific Limitations and Differences

This section lists CTI OS-related restrictions and implementation differences for various peripherals.



Note

- MAKE_CALL is only supported when the agent is in the NotReady state for an UCCE peripheral.
- MAKE_CALL is not supported for the remaining peripherals supported by CTI OS.
- The call continues to be active even after a party is released from the conference.

Aspect Contact Server

- AgentExtension and AgentInstrument are defined as the port number that the teleset is connected to.
- Events marked by an asterisk (*) are available when the PG is configured to use the Aspect EventLink.
- Call Alerting (Call Delivered, LocalConnectionState = LCS_ALERTING) is available when the EventLink is used.
- Outbound calls on some trunk types do not always provide Call Cleared events. Interflow calls that are accepted, but handled by the originating site, sometimes also do not provide Call Cleared events.
- Outbound calls require that you specify the CallPlacementType in an outbound request.
- Conference calls can have a maximum of three parties.
- In a single-step/blind transfer of a call, the initial call must come in over a trunk (be a CCT call) and the dialed number must go to a CCT.
- In a regular call transfer, the consult call can be either a CCT call or an agent_inside call.
- Alternate call operations require that the initial call is a CCT call. The second call (consult call) can be either a CCT call or an agent_inside call.
- In the MAKE_PREDICTIVE_CALL_REQ message, the AnswerDetectControl1 field must contain the binary value of the Application Bridge AD_PARAM setting, and the AnswerDetectControl2 field must contain the binary value of the Application Bridge ANS_MAP setting.
- Transfer and Conference behavior is modeled after hardphone behavior. To initiate a Transfer or a Conference, use the MakeCall control (Transfer Init and Conference Init buttons are unavailable) to make a second (consult) call. After you make this call, the Transfer Complete and Conference Complete buttons are available to complete the desired action.

Avaya DEFINITY ECS

- AgentExtension and AgentInstrument are defined as the station extension.
- DEFINITY ECS events are the same with or without EAS (Expert Agent Selection).
- Both EAS and non-EAS versions maintain a list of preconfigured agent groups. When you log in with EAS, the agent is automatically logged in to all preconfigured Agent groups. When you log in without EAS, the agent is logged in to only those groups that you specify in the login request.
- The Cisco Peripheral Interface Module (PIM)—the Cisco proprietary interface between a peripheral and the Peripheral Gateway (PG)—does support call events on inside calls only when Unified ICM monitors the agent's station (agent station appears in the Unified ICM Peripheral Monitor Table), when the call goes through a monitored VDN, or when the call is originated by a CTI MakeCallReq. An agent on the switch originates Inside calls. Inside calls include consult calls before a transfer or conference. After the transfer or conference completes, you can see call events for the merged ACD call.
- Auto Answer agents must have the phone off the hook or you cannot log in to the agent. Manual Answer agents must leave the phone on the hook.
- Applications must wait a time interval of three times the refresh rate (defined in the Avaya Call Management System) between login or logout attempts. Failure to do so may cause the PIM to miss the login event and result in a failed call request.

- If a third-party action fails, an ASAI cause value returns for CTI OS clients that access a DEFINITY ECS switch. If you have a copy of the *DEFINITY Technical Reference Manual*, you can determine the actual cause of the failure by performing the following steps:
 - Refer to the following table of “DEFINITY Cause Values” to obtain the DEFINITY ECS value that corresponds to the returned ASAI cause value.
 - Refer to the following table “Third-party request/section in DEFINITY manual” to find the chapter of the *DEFINITY Technical Reference Manual* that discusses the third-party action that you attempted.
 - Refer to the chapter specified in the table “Third-party request/section in DEFINITY manual” for an explanation of the DEFINITY ECS cause value.

Table 32: DEFINITY Cause Values

ASAI Value	DEFINITY ECS Value	Cause Value	Description
-MAX_LONG	none	*C_NUSE_LONG	The ECS does not return a value.
0	CS0/28	*C_INVLDNUM	Invalid origination or destination address.
1	CS0/111	*C_PROTERR	Capability sequence was violated or underlying protocol error was detected; the ECS returned an unrecognized value.
2	CS3/40	*C_RESUNAVL	Resources to fulfill service are not available.
3	CS0/50	*C_FACUNSUB	Capability is implemented but not subscribed to by requester.
4	CS3/79	*C_SER_UNIMP	Incompatible options selected.
5	CS0/96	*C_MAND_INFO	One of the required parameters is missing.
6	CS0/100	*C_INVLDIE	Value specified in parameter is not allowed or defined.
7	CS3/63	*C_SERV_UNAVIL	Domain or call is being monitored by another adjunct.
8	CS3/86	*C_CALLID_TERM	Call is no longer in active state.
9	CS0/98	*C_INCOM_ST	Message not compatible with call state.
10	CS0/81	*C_INVALID_CRV	Invalid call identifier (sao_id) also known as cluster_id is used or call does not exist.

ASAI Value	DEFINITY ECS Value	Cause Value	Description
11	CS3/80	*C_INCOM_OPT	Incompatible options used to establish the call.
12	CS0/102	*C_REC_TIMER	Timer expired.
13	CS3/15	*C_NOLOGIN	Agent not logged in to split.
14	CS3/11	*C_NOSPLIT_MEM	Agent not member of specified split or split number specified incorrectly.
15	CS0/17	*C_USER_BUSY	Domain or call is being monitored by another adjunct.
16	CS0/18	*C_NOUSE_RESP	Originating address does not respond to service.
17	CS3/43	*C_PERM_DENIED	Permission checks for service have failed.
18	CS3/87	*C_CLUST_TERM	Association terminated because service is not active.
19	CS3/27	*C_OUT_OF_SERV	Domain was removed by administration.
20	CS3/12	*C_INCS_AGT_ST	Agent not in compatible state.
21	CS3/13	*C_MAXLOGIN	Agent logged in to maximum number of splits.
22	CS3/14	*C_INC_PASWD	Invalid login password.
23	CS3/16	*C_AGT_STATE	Request to put agent in the state that the agent is already in.
24	CS3/41	*C_BAD_ADMIN	ACD not provisioned or optioned.
25	CS0/16	*C_NORMAL	Normal termination; call routed successfully.
26	CS0/42	*C_NETCONJ	Association terminated because of network congestion.
27	CS0/99	*C_BAD_IE	Unknown information element detected.
28	CS3/22	*C_QUEFULL	Queue is full.
29	CS3/42	C_REORDER_DENIAL	Reorder/Denial.
30	CS3/46	C_ADMIN_PROGRESS	Administration is in progress; request cannot be serviced.

ASAI Value	DEFINITY ECS Value	Cause Value	Description
31	CS3/53	C_FEATURE_REJECTED	The ECS has rejected a request from the adjunct.
32	CS0/1	C_UNASSIGNED_NUM	Unassigned number.
33	CS0/21	C_CALL_REJECTED	Call rejected.
34	CS0/22	C_NUM_CHANGED	Number changed.
35	CS0/31	C_NORMAL_UNSPECIF	Normal, unspecified.
36	CS0/34	C_NO_CIRCUIT	No circuit or channel available.
37	CS0/41	C_TEMP_FAILURE	Temporary Failure.
38	CS0/58	C_BEARER_CAP_UNAVAIL	Bearer capability not presently available.
39	CS0/88	C_INCOMPAT_DESTINATION	Incompatible destination.
40	CS0/95	C_INVALID_MESSAGE	Invalid message, unspecified (backward compatibility).
41	CS0/97	C_NON_EXIST_MESSAGE	Message nonexistent/ not implemented.
42	CS0/127	C_UNSPECIFIED	Unspecified.
43	CS3/19	C_NO_ANSWER	No answer.
44	CS3/20	C_NO_TRUNKS	Trunks not available.
45	CS3/21	C_NO_CLASSIFIERS	Classifiers not available.
46	CS3/30	C_REDIRECT	Redirected.
47	CS3/38	C_NETWORK_OUT_OF_ORDER	Network out of order.
48	Undefined	*C_CAUSE_UNKNOWN	Undefined value returned from the ECS.
49	CS0/52	*C_OUT_CALL_BARRED	Outgoing call was barred.
50	CS3/23	C_REMAINS_IN_Q	Call remains in queue.
51	CS0/65	C_BEARER_SVC_NOT_IMPL	Bearer service not implemented.

ASAI Value	DEFINITY ECS Value	Cause Value	Description
52	CS3/17	C_TIMED_ANSWER	Assumed answer based on internal timer.
53	CS3/18	C_VOICE_ENERGY_ANSWER	Voice energy detected by the ECS.
54	CS0/82	C_NO_TONE_CHANNEL	Channel or tone do not exist (no tone connected to the specified call).
55	CS3/24	C_ANSWERING_MACHINE	Answering machine detected.
56	CS0/29	C_FACILITY_REJECTED	Facility rejected.
57	CS3/25	C_FORWARD_BUSY	Redirection cause.
58	CS3/26	C_COVER_BUSY	Redirection cause.
59	CS3/28	C_COV_DONT_ANS	Redirection cause.
60	CS3/31	C_FORWARD_ALL	Redirection cause.
61	CS3/8	C_LISTEN_ONLY	Single-Step Conference listen only.
62	CS3/9	C_LISTEN_TALK	Single-Step Conference listen-talk.

For example, an ASAI value of 15 corresponds to the DEFINITY ECS value of CS0/17 (C_USER_BUSY).

Table 33: Third-Party Request/Section in DEFINITY Manual

Third-party Action or Request	Chapter in Manual
Third-party actions via Call Control: Auto Dial (3PAD), Clear (3PCC), Deflect (Redirect) (3PREDIR), Drop (Selective Drop) (3PSD), Listen-Disconnect, Listen-Reconnect, Selective Hold (3PSH), Make Call (3PMC) (or Predictive Call), Relinquish Control (3PRC), Reconnect (Retrieve) (3PR), Send DTMF (3PSDS), Take Control (3PTC)	Chapter 4: ASAI and Call Control
Third-Party actions via Domain Control: Auto Dial (3PAD), Domain Control (3PDC), Answer (3PANS), Merge (Transfer/Conference) (3PM)	Chapter 5: ASAI and Domain Control
Call Routing (RT_REQ, RT_SEL, RT_END)	Chapter 7: ASAI and Call Routing
Agent State change: Login, Logout, Change Workmode: NotReady (AUX), Ready (AVAIL), WorkReady (ACW), and so forth.) Activating/Canceling Call Forwarding Activating/Canceling Send All Calls	Chapter 8: ASAI and Request Feature Capabilities
Value Queries	Chapter 9: ASAI and Value Query Capabilities

Third-party Action or Request	Chapter in Manual
Set Value: Message Waiting Indicator (MWI) Set Billing Type	Chapter 10: ASAI and Set Value Capabilities

For example, Chapter 8, “ASAI and Request Feature Capabilities” discusses third-party login requests.

When TSAPI interface of Avaya peripheral is used, PIM maps CSTA error is returned by TSAPI CSTA APIs to ASAI error codes. The following tables display the mapping API by API bases.

Table 34: Third Party Answer - cstaAnswerCall

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
INVALID_CSTA_DEVICE_IDENTIFIER (12) - An invalid device identifier or extension is specified in the alerting call.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the parameters was invalid]
INVALID_CSTA_CONNECTION_IDENTIFIER (13) - An incorrect callID or an incorrect deviceID is specified.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the parameters was invalid]
GENERIC_STATE_INCOMPATIBILITY (21) - The station user did not go off-hook within five seconds and cannot be forced off-hook.	GENERIC_OPERATION_REJECTION (71)	C_NOUSE_RESP(16) [Originating address does not respond to service]
INVALID_OBJECT_STATE (22) - The specified connection at the station is not in alerting, connected, held, or bridged state.	GENERIC_OPERATION_REJECTION (71)	C_INCOM_ST(9) [Message not compatible with call state]
NO_CALL_TO_ANSWER (28) - The call was redirected to coverage within the five-second interval.	GENERIC_OPERATION_REJECTION (71)	C_INVALID_CRV(10) [Invalid call identifier (sao_id), also known as cluster_id is used or call does not exist]
GENERIC_SYSTEM_RESOURCE_AVAILABILITY (31) - The client attempted to add a seventh party to a call with six active parties.	GENERIC_OPERATION_REJECTION (71)	C_RESUNAVL(40) [Resources to fulfill service are not available]
RESOURCE_BUSY (33) - The user at the station is busy on a call or there is no idle appearance available.	GENERIC_OPERATION_REJECTION (71)	C_USER_BUSY(15) [Domain or call is being monitored by another adjunct]

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
GENERIC_SUBSCRIBED_RESOURCE_AVAILABILITY (41) - The device identifier specified for alerting a call corresponds to a SIP station and the "Type of 3PCC Enabled" for the station is not set to "Avaya".	GENERIC_OPERATION_REJECTION (71)	C_FACUNSUB(3) [Capability is implemented but not subscribed to by requester]
MISTYPED_ARGUMENT_REJECTION (74) - DYNAMIC_ID is specified in alerting call.	GENERIC_OPERATION_REJECTION (71)	C_SER_UNIMP(4) [Incompatible options selected]

Table 35: Third Party Drop - cstaClearConnection

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
GENERIC_UNSPECIFIED (0) - The specified data provided for the userInfo parameter exceeds the maximum size. For private data versions 2-5, the maximum length for userInfo is 32 bytes. Beginning with private data version 6, the maximum length was increased to 96 bytes.	GENERIC_OPERATION_REJECTION (71)	C_INVLDIE(6) [Value specified in parameter is not allowed or defined]
INVALID_OBJECT_STATE (22) - The specified connection at the station is not currently active (is either in alerting or held state) so it cannot be dropped.	GENERIC_OPERATION_REJECTION (71)	C_INCOM_ST(9) [Message not compatible with call state]
NO_ACTIVE_CALL (24) - The connectionID contained in the request is invalid. CallID may be incorrect too.	GENERIC_OPERATION_REJECTION (71)	C_INVALID_CRV(10) [Invalid call identifier (sao_id), also known as cluster_id is used or call does not exist]
NO_CONNECTION_TO_CLEAR (27) - The connectionID contained in the request is invalid. CallID may be correct, but deviceID is wrong.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
RESOURCE_BUSY (33) - The switch is busy with another CSTA request. This happens when two AE Services servers are issuing requests (Hold Call, Retrieve Call, Clear Connection, and so on) to the same device.	GENERIC_OPERATION_REJECTION (71)	C_USER_BUSY(15) [Domain or call is being monitored by another adjunct]

Table 36: Third Party Merge - cstaConferenceCall

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
INVALID_CSTA_DEVICE_IDENTIFIER (12) - An invalid device identifier or extension is specified in heldCall or activeCall.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
INVALID_CSTA_CONNECTION_IDENTIFIER (13) - The controlling deviceID, in heldCall, or activeCall has not been specified correctly.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
GENERIC_STATE_INCOMPATIBILITY (21) - Both calls are alerting, both calls are being service-observed, or an active call is in a vector processing stage.	GENERIC_OPERATION_REJECTION (71)	C_REORDER_ENIAL(29) [Reorder/Denial]
INVALID_OBJECT_STATE (22) - The connections specified in the request are not in valid states for the operation to take place. For example, it does not have one call active and one call in the held state as required.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
RESOURCE_BUSY (33) - The switch is busy with another CSTA request. This can happen when two AE Services servers are issuing requests (Hold Call, Retrieve Call, Clear Connection, Conference Call, and so on) to the same device.	GENERIC_OPERATION_REJECTION (71)	C_USER_BUSY(15) [Domain or call is being monitored by another adjunct]
CONFERENCE_MEMBER_LIMIT_EXCEEDED (38) - The request attempted to add a seventh party to an existing six-party conference call. If a station places a six-party conference call on hold and another party adds another station (so that there are again six active parties on the call which is the limit of the Communication Manager), then the station with the call on hold will not be able to retrieve the call.	GENERIC_OPERATION_REJECTION (71)	C_REORDER_DENIAL (29) [Reorder/Denial]
GENERIC_SUBSCRIBED_RESOURCE_AVAILABILITY (41) - The device identifier specified in activeCall and heldCall corresponds to a SIP station and the "Type of 3PCC Enabled" for the station is not set to "Avaya".	GENERIC_OPERATION_REJECTION (71)	C_FACUNSUB(3) [Capability is implemented but not subscribed to by requester]

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
MISTYPED_ARGUMENT_REJECTION (74) - DYNAMIC_ID is specified in heldCall or activeCall.	GENERIC_OPERATION_REJECTION (71)	C_SER_UNIMP(4) [Incompatible options selected]

Table 37: Third Party Hold - cstaHoldCall

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
INVALID_CSTA_DEVICE_IDENTIFIER (12) - An invalid device identifier or extension is specified in activeCall.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
INVALID_CSTA_CONNECTION_IDENTIFIER (13) - The connection identifier contained in the request is invalid or does not correspond to a station.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
NO_ACTIVE_CALL (24) - The party to be put on hold is not currently active (for example, in the alerting state) so it cannot be put on hold.	GENERIC_OPERATION_REJECTION (71)	C_INCOM_ST(9) [Message not compatible with call state]
RESOURCE_BUSY (33) - The switch is busy with another CSTA request. This can happen when two AEI Services servers are issuing requests (Hold Call, Retrieve Call, Clear Connection, and so on) for the same device.	GENERIC_OPERATION_REJECTION (71)	C_USER_BUSY(15) [Domain or call is being monitored by another adjunct]
GENERIC_SUBSCRIBED_RESOURCE_AVAILABILITY (41) - The device identifier specified in activeCall corresponds to a SIP station and the "Type of 3PCC Enabled" administered for the station is not set to "Avaya".	GENERIC_OPERATION_REJECTION (71)	C_FACUNSUB(3) [Capability is implemented but not subscribed to by requester]
OUTSTANDING_REQUEST_LIMIT_EXCEEDED (44) - The client attempted to put a third party on hold while two parties are on hold already, on an analog station.	GENERIC_OPERATION_REJECTION (71)	C_INCOM_ST(9) [Message not compatible with call state]
MISTYPED_ARGUMENT_REJECTION (74) - DYNAMIC_ID is specified in inactiveCall.	GENERIC_OPERATION_REJECTION (71)	C_SER_UNIMP(4) [Incompatible options selected]

Table 38: Third Party Make - cstaMakeCall

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
GENERIC_UNSPECIFIED (0) - The specified data provided for the userInfo parameter exceeds the maximum allowable size. For private data versions 2-5, the maximum length of userInfo is 32 bytes. Beginning with private data version 6, the maximum length of userInfo is 96 bytes.	GENERIC_OPERATION_REJECTION (71)	C_INVLDIE(6) [Value specified in parameter is not allowed or defined]
INVALID_CALLING_DEVICE (5) - The callingDevice is out of service or not administered correctly in the switch.	GENERIC_OPERATION_REJECTION (71)	C_OUT_OF_SERV(19) [Domain was removed by Administration]
INVALID_CSTA_DEVICE_IDENTIFIER (12) - An invalid device identifier or extension is specified in callingDevice.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
GENERIC_STATE_INCOMPATIBILITY (21) - The originator does not go off-hook within five seconds after originating the call and cannot be forced off-hook.	GENERIC_OPERATION_REJECTION (71)	C_NOUSE_RESP(16) [Originating address does not respond to service]
RESOURCE_BUSY (33) - The user is busy on another call and cannot originate this call, or the switch is busy with another CSTA request. This can happen when two AE Services servers are issuing requests (Hold Call, Retrieve Call, Clear Connection, Make Call, and so on) for the same device.	GENERIC_OPERATION_REJECTION (71)	C_USER_BUSY(15) [Domain or call is being monitored by another adjunct]
GENERIC_SUBSCRIBED_RESOURCE_AVAILABILITY (41) - The device identifier specified in callingDevice corresponds to a SIP station and the "Type of 3PCC Enabled" administered for the station is not set to "Avaya".	GENERIC_OPERATION_REJECTION (71)	C_FACUNSUB(3) [Capability is implemented but not subscribed to by requester]

Table 39: Third Party Retrieve - cstaRetrieveCall

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
INVALID_CSTA_DEVICE_IDENTIFIER (12) - An invalid device identifier or extension is specified in heldCall.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [Invalid origination or destination address]

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
INVALID_CSTA_CONNECTION_IDENTIFIER (13) - The connectionID contained in the request is invalid.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [Invalid origination or destination address]
GENERIC_STATE_INCOMPATIBILITY (21) - The user was on-hook when the request was made and did not go off-hook within five seconds (call remains on hold).	GENERIC_OPERATION_REJECTION (71)	C_NOUSE_RESP(16) [Originating address does not respond to service]
NO_ACTIVE_CALL (24) - The specified call at the station is cleared and so it cannot be retrieved.	GENERIC_OPERATION_REJECTION (71)	C_INVALID_CRV(10) [Invalid call identifier (sao_id) also known as cluster_id is used or call does not exist]
NO_HELD_CALL (25) - The specified connection at the station is not in the held state (for example, in the alerting state) and so it cannot be retrieved.	GENERIC_OPERATION_REJECTION (71)	C_INCOM_ST(9) [Message not compatible with call state]
RESOURCE_BUSY (33) - The switch is busy with another CSTA request. This can happen when two AE Services servers are issuing requests (Hold Call, Retrieve Call, Clear Connection, Conference Call, and so on) for the same device.	GENERIC_OPERATION_REJECTION (71)	C_USER_BUSY(15) [Domain or call is being monitored by another adjunct]
CONFERENCE_MEMBER_LIMIT_EXCEEDED (38) - The client attempted to add a seventh party to a six-party conference call.	GENERIC_OPERATION_REJECTION (71)	C_RESUNAVL(40) [Resources to fulfill service are not available]
GENERIC_SUBSCRIBED_RESOURCE_AVAILABILITY (41) - The device identifier specified in heldCall corresponds to a SIP station and the "Type of 3PCC Enabled" administered for the station is not set to "Avaya".	GENERIC_OPERATION_REJECTION (71)	C_FACUNSUB(3) [Capability is implemented but not subscribed to by requester]
MISTYPED_ARGUMENT_REJECTION (74) - DYNAMIC_ID is specified in heldCall.	GENERIC_OPERATION_REJECTION (71)	C_SER_UNIMP(4) [Incompatible options selected]

Table 40: Third Party Transfer - cstaTransferCall

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
INVALID_CSTA_DEVICE_IDENTIFIER (12) - An invalid device identifier or extension was specified in heldCall or activeCall.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
INVALID_CSTA_CONNECTION_IDENTIFIER (13) - The controllingdeviceID in activeCall or heldCall has not been specified correctly.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
GENERIC_STATE_INCOMPATIBILITY (21) - The request failed due to one of the following reasons: <ul style="list-style-type: none"> • Both calls are alerting • Both calls are being service-observed • An active call is in a vector-processing stage • The Trunk-to-Trunk Transfer feature is not enabled on Avaya Communication Manager 	GENERIC_OPERATION_REJECTION (71)	C_REORDER_DENIAL (29) [Reorder/Denial]
INVALID_OBJECT_STATE (22) - The connections specified in the request are not in valid states for the operation to take place. For example, the transferring device does not have one active call and one held call as required.	GENERIC_OPERATION_REJECTION (71)	C_INCOM_ST(9) [Message not compatible with call state]
INVALID_CONNECTION_ID_FOR_ACTIVE_CALL (23) - The callID inactiveCall or heldCall has not been specified correctly.	GENERIC_OPERATION_REJECTION (71)	C_INVLDNUM(0) [One of the entered parameters was invalid]
RESOURCE_BUSY (33) - The switch is busy with another CSTA request. This can happen when two AE Services servers are issuing requests (Hold Call, Retrieve Call, Clear Connection, Transfer Call, and so on) for the same device.	GENERIC_OPERATION_REJECTION (71)	C_REORDER_DENIAL(29) [Reorder/Denial]
GENERIC_SUBSCRIBED_RESOURCE_AVAILABILITY (41) - The device identifier specified in activeCall and heldCall corresponds to a SIP station and the "Type of 3PCC Enabled" administered for the station is not set to "Avaya".	GENERIC_OPERATION_REJECTION (71)	C_FACUNSUB(3) [Capability is implemented but not subscribed to by requester]

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
MISTYPED_ARGUMENT_REJECTION (74) - DYNAMIC_ID is specified in heldCall or activeCall.	GENERIC_OPERATION_REJECTION (71)	C_SER_UNIMP(4) [Incompatible options selected]

Table 41: Third Party Clear - cstaClearCall

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
NO_ACTIVE_CALL (24) - The callID of the connectionID specified in the request is invalid.	GENERIC_OPERATION_REJECTION (71)	C_INVALID_CRV(10) [Invalid call identifier (sao_id) also known as cluster_id is used or call does not exist]

Table 42: Third Party Set Agent State - cstaSetAgentState

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
<p>GENERIC_UNSPECIFIED (0) - The request failed due to one of the following reasons:</p> <ul style="list-style-type: none"> • The request attempted to log out an ACD agent who is already logged out • The request attempted to log an ACD agent into a split of which they are not a member • The request attempted to log in an ACD agent with an incorrect password • The request attempted to log in an ACD agent at a station where the Auto Answer feature is enabled, but the station is not off-hook. 	GENERIC_UNSPECIFIED (0)	<p>C_NOLOGIN [for agent logout request when agent not logged in]</p> <p>C_INC_PASWD(22) [for agent login request]</p> <p>C_INCS_AGT_ST [for any other set agent state requests]</p>

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
<p>GENERIC_OPERATION (1) - The request attempted to log in an ACD agent that is already logged in.</p>	<p>For TP login Request:</p> <ul style="list-style-type: none"> • SPECIFIED_EXTENSION_ALREADY_IN_USE(283) [Other Agent is already logged in on device] • SPECIFIED_AGENT_ALREADY_SIGNED_ON(259) [Same Agent is logged on same device] • GENERIC_OPERATION_REJECTION (71) [station not in service] <p>For non TP login Request:</p> <ul style="list-style-type: none"> • GENERIC_OPERATION (1) 	<p>For TP login Request:</p> <ul style="list-style-type: none"> • C_CAUSE_UNKNOWN [Other Agent is already logged in on device/Same Agent is logged on same device - not to be used] • C_OUT_OF_SERV [Station not in service] • C_AGT_STATE (23) [for all requests other than login]
<p>VALUE_OUT_OF_RANGE (3)</p> <p>The request failed due to one of the following reasons:</p> <ul style="list-style-type: none"> • The workMode private parameter is not valid for the agentMode • The reason code is outside of the acceptable range (1- 9 or 1-99). (CS0/100) 	<p>INVALID_AGENT_WORKMODE [for work-mode change requests]</p> <p>INVALID_AGENT_REASON_CODE [for logout requests]</p>	<p>C_CAUSE_UNKNOWN (Not to be used)</p>
<p>OBJECT_NOT_KNOWN (4)</p> <p>The request failed due to one of the following reasons:</p> <ul style="list-style-type: none"> • service request did not specify a valid on-PBX station for the ACD agent in device • agentGroup or device parameters were NULL • agentID parameter was NULL when agentMode was set to AM_LOG_IN 	<p>GENERIC_UNSPECIFIED_REJECTION (70)</p>	<p>C_INVLDNUM(0) [One of the entered parameters was invalid]</p>
<p>INVALID_CSTA_DEVICE_IDENTIFIER (12) - An invalid device identifier has been specified in the device.</p>	<p>GENERIC_UNSPECIFIED_REJECTION (70)</p>	<p>C_INVLDNUM(0) [One of the entered parameters was invalid]</p>

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
INVALID_FEATURE (15) - The feature is not available for the agentGroup or the enablePending feature is not available for the switch version.	GENERIC_UNSPECIFIED_REJECTION (70)	C_SERV_UNAVIL(7) [Domain or call is being monitored by another adjunct]
INVALID_OBJECT_TYPE (18) (CS3/80) - A reason code was specified, but the specified workMode was not WM_AUX_WORK or AM_LOG_OUT.	GENERIC_UNSPECIFIED_REJECTION (70)	C_INCOM_OPT(11) [Incompatible options used to establish the call]
GENERIC_STATE_INCOMPATIBILITY (21) <ul style="list-style-type: none"> • A work mode change was requested for a non-ACD agent • The Agent station is maintenance busy or out of service 	GENERIC_UNSPECIFIED_REJECTION (70)	C_MAXLOGIN(21) [for login requests - Agent logged in to maximum number of splits] C_INCS_AGT_ST [for all other requests other than login]
GENERIC_SYSTEM_RESOURCE_AVAILABILITY (31) - The request cannot complete due to lack of available switch resources.	GENERIC_UNSPECIFIED_REJECTION (70)	C_RESUNAVL(2) [Resources to fulfill service are not available]
RESOURCE_BUSY (33) - The service attempted to change the state of an ACD agent that is currently on a call.	GENERIC_UNSPECIFIED_REJECTION (70)	C_USER_BUSY(17) [Domain or call is being monitored by another adjunct]
GENERIC_SUBSCRIBED_RESOURCE_AVAILABILITY (41) - The device identifier specified in device corresponds to a SIP station and the "Type of 3PCC Enabled" administered for the station is not set to "Avaya".	GENERIC_UNSPECIFIED_REJECTION (70)	C_FACUNSUB(3) [Capability is implemented but not subscribed to by requester]

The following errors apply to every CSTA Service that is supported by the TSAPI Service.

Table 43: Common Switch-related CSTA Service Errors

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
GENERIC_UNSPECIFIED (0) - An error has occurred. The TSAPI Service could not provide a specific error value.	GENERIC_UNSPECIFIED (0)	C_CAUSE_UNKNOWN

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
GENERIC_OPERATION (1) - The CTI protocol is broken d or the service invoked is not consistent with a CTI application association.	GENERIC_OPERATION (1)	C_PROTERR(1) [Capability sequence was violated or underlying protocol error was detected; an unrecognized value was returned by the ECS]
REQUEST_INCOMPATIBLE_WITH_OBJECT (2) - The service request does not correspond to a CTI application association.	GENERIC_UNSPECIFIED_REJECTION (70)	C_FEATURE_REJECTED (31) [The ECS has rejected a request from the adjunct]
VALUE_OUT_OF_RANGE (3) - Communication Manager detects that a required parameter is missing from the request or an out-of-range value has been specified.	GENERIC_UNSPECIFIED_REJECTION (70)	C_MAND_INFO(5) [One of the required parameters is missing]
OBJECT_NOT_KNOWN (4) - The TSAPI Service detects that a required parameter is missing in the request. For example, the deviceIDof a connectionID is not specified in a service request.	GENERIC_UNSPECIFIED_REJECTION (70)	C_MAND_INFO(5) [One of the required parameters is missing]
INVALID_FEATURE (15) - The TSAPI Service detects a CSTA Service request that is not supported by Communication Manager.	GENERIC_UNSPECIFIED_REJECTION (70)	C_SERV_UNAVIL(7) [Domain or call is being monitored by another adjunct]
GENERIC_SYSTEM_RESOURCE_AVAILABILITY (31) - The request cannot be completed due to lack of available switch resources.	GENERIC_UNSPECIFIED_REJECTION (70)	C_RESUNAVL(2) [Resources to fulfill service are not available]
RESOURCE_OUT_OF_SERVICE (34) - An application can receive this error code when a single CSTA Service request is ending abnormally due to protocol error.	GENERIC_UNSPECIFIED_REJECTION (70)	C_PROTERR(1) [Capability sequence was violated or underlying protocol error was detected; an unrecognized value was returned by the ECS]

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
NETWORK_BUSY (35) - Communication Manager is not accepting the request at this time because of processor overload. The application may wish to retry the request but should not do so immediately.	GENERIC_UNSPECIFIED_REJECTION (70)	C_NETCONJ
GENERIC_SUBSCRIBED_RESOURCE_AVAILABILITY (41) - The TSAPI Service could not acquire the license(s) needed to satisfy the request.	GENERIC_OPERATION_REJECTION (71)	C_FACUNSUB(3) [Capability is implemented but not subscribed to by requester]
OUTSTANDING_REQUEST_LIMIT_EXCEEDED (44) - The given request cannot be processed due to a system resource limit on the device.	GENERIC_OPERATION_REJECTION (71)	C_RESUNAVL(2) [Resources to fulfill service are not available]
GENERIC_UNSPECIFIED_REJECTION (70) - This is a TSAPI Service internal error, but it cannot be more specific. The system administrator should check the AE Services OAM error logs for more information about this error.	GENERIC_UNSPECIFIED_REJECTION (70)	C_CAUSE_UNKNOWN
GENERIC_OPERATION_REJECTION (71) - This is a TSAPI Service internal error, but not a defined error. The system administrator should check the TSAPI Service error logs for more information about this error.	GENERIC_OPERATION_REJECTION (71)	C_CAUSE_UNKNOWN
DUPLICATE_INVOCATION_REJECTION (72) - The TSAPI Service detects that the invokeID in the service request is being used by another outstanding service request. This service request is rejected. The outstanding service request with the same invokeID is still valid.	GENERIC_OPERATION_REJECTION (71)	C_SER_UNIMP(4) [Incompatible options selected]
UNRECOGNIZED_OPERATION_REJECTION (73) - The TSAPI Service detects that the service request from a client application is not defined in the API. A CSTA request with a 0 or negative invokeID will receive this error.	GENERIC_OPERATION_REJECTION (71)	C_SER_UNIMP(4) [Incompatible options selected]

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
<p>RESOURCE_LIMITATION_REJECTION (75) - The TSAPI Service detects that it lacks internal resources such as the memory or data records to process a service request. A system administrator should check the TSAPI Service error logs for more detailed information about this error. This failure may reflect a temporary situation. The application should retry the request.</p>	<p>GENERIC_OPERATION_REJECTION (71)</p>	<p>C_TEMP_FAILURE(37) [Temporary Failure]</p>
<p>ACS_HANDLE_TERMINATION_REJECTION (76) - The TSAPI Service detects that anacsOpenStream session is terminating. The TSAPI Service sends this error for every outstanding CSTA request of this ACS Handle.</p> <p>For example, a user may power off the PC before the application issues anacsCloseStream request and waits for the confirmation event. In this case, the acsCloseStream is issued by the TSAPI Service on behalf of the application and there is no application to receive this error. If an application issues anacsCloseStream request and waits for its confirmation event, the application will receive this error for every outstanding request.</p>	<p>GENERIC_OPERATION_REJECTION (71)</p>	<p>C_CAUSE_UNKNOWN</p>
<p>SERVICE_TERMINATION_REJECTION (77) - The TSAPI Service detects that it cannot provide the service due to the failure or shutting down of the communication link between the Telephony Server and Communication Manager. The TSAPI Service sends this error for every outstanding CSTA request that effects every ACS Handle. Although the link is down or Communication Manager is out of service, the TSAPI Service remains loaded and advertised. When the TSAPI Service is in this state, all CSTA Service requests from a client will receive a negative acknowledgment with this error code.</p>	<p>GENERIC_OPERATION_REJECTION (71)</p>	<p>C_CAUSE_UNKNOWN</p>

CSTA Error Code Returned by TSAPI	ICM CSTA error code	Mapped Cause ASAI Value
<p>REQUEST_TIMEOUT_REJECTION (78) - The TSAPI Service did not receive the response of a service request sent to Communication Manager more than 30 seconds ago. The request is canceled and negatively acknowledged with this error code. When this occurs, the communication link between the TSAPI Service and Communication Manager may be out of service or congested. Congestion may occur when TSAPI applications exceed the capacity of the TSAPI Service.</p>	<p>GENERIC_OPERATION_REJECTION (71)</p>	<p>C_REC_TIMER(12) [Timer expired]</p>
<p>REQUESTS_ON_DEVICE_EXCEEDED_REJECTION (79) - The TSAPI Service processes one service request at a time for every device. The TSAPI Service queues CSTA requests for a device. Only a limited number of CSTA requests are queued on a device. If this number is exceeded, the incoming client request is negatively acknowledged with this error code. Usually an application sends one request and waits for its completion before it makes another request. The MAX_REQS_QUEUED_PER_DEVICE parameter has no effect on this class of applications.</p> <p>Situations of sending a sequence of requests without waiting for their completion are rare. However, if this is the case, set the MAX_REQS_QUEUED_PER_DEVICE parameter to a proper value. The default value for MAX_REQS_QUEUED_PER_DEVICE is 4.</p>	<p>GENERIC_OPERATION_REJECTION (71)</p>	<p>C_QUEFULL(28) [Queue is full]</p>

Unified CCE System PG

- MAKE_CALL is only supported when the agent is in the NotReady state. An agent cannot make new calls when in wrapup mode.
- Consult and blind transfers are supported. However, placing a call on hold, making a new call, and then completing the transfer is not supported.

- The consult call must be in the Talking state before the Transfer/Conference can be completed. Therefore, if an Alternate is done in the middle of a Transfer/Conference, the operation can only be completed after a second Alternate is done to restore status quo.
- Completing a conference or a transfer to a consulted agent on hold is not supported.
- Transferring conferences to an unobserved party is not supported.
- Overlapping transfer and conference consult operations on the same parties are not supported. For example, Agent A calls Agent B. During the conversation, Agent A must conference consult Agent C. Agent B feels that Agent D has more information, so Agent B then transfer consults to Agent D. To end the call, Agent A completes the conference and Agent B completes the transfer. This would fail.
- Only the conference initiator can add parties to the conference.
- Calls do not get queued at the Unified CM but instead at some queue point. Because of this, skill group queue statistics are not available via the `QUERY_SKILL_GROUP_STATISTICS_REQ`. CTI can monitor service controlled VRUs to get queued and dequeued events, as well as established events.
- `RTP_STARTED_EVENT` and `RTP_STOPPED_EVENT` are particular to Unified CCE to support recording vendors.
- `AGENT_PRECALL_EVENT` and `AGENT_PRECALL_ABORT_EVENT` are particular to Unified CCE. They provide call context data before the routed call arrives.
- A `CALL_CONNECTION_CLEARED_EVENT` may be received with a cause of `CEC_REDIRECTED` for the following cases:
 - Agent calls a CTI Route Point and call is directed to another resource
 - Agent calls an VRU and the VRU redirects the call
 - Agent calls a number with a forwarding option turned on
- You can only monitor devices that have agents logged in via CTI OS. The Unified ICM Peripheral Monitor Table is not supported for the Unified CCE PG.
- The Unified CM Shared line feature (agents share the same extension) is not supported.
- Agent Desk Settings control some agent behaviors. These are configured in Unified ICM and downloaded by the agent desktop upon startup. `WrapupInMode` is the wrapup mode variable for incoming calls and `WrapupOutMode` is the wrapup mode variable for outgoing calls. The valid values for these parameters are:
 - **REQUIRED**

For either incoming or outgoing calls, the agent has no option but to go to the Wrapup state when a call ends. While the agent is on the call, all agent state buttons are disabled. While the agent is in the wrapup state, the Ready and NotReady buttons must be enabled.

Clicking either the Ready or NotReady buttons must dismiss the Wrapup dialog box and put the agent in the state that was chosen. However, if the wrapup timer was enabled in the PG configuration and timeout occurs before an agent state is chosen, the agent state automatically changes as follows:

 - If the timeout occurred at the end of an incoming call, the agent state changes to Ready.
 - If the timeout occurred at the end of an outgoing call, the agent state changes to NotReady.

- **REQUIRED_WITH_DATA**

The same as **REQUIRED**, but the agent must input some data into the Wrapup dialog box before exiting the dialog box and going to a Ready or NotReady state. This applies only to WrapupInMode.

- **OPTIONAL**

For either incoming or outgoing calls, the agent can only enter any after call state—Wrapup, Ready or NotReady—by clicking the appropriate button.

- **NOT_ALLOWED**

For either incoming or outgoing calls, the agent is only able to enter the Ready or NotReady states. The wrapup button is disabled.

Points of note for API users:

- If the wrapup mode is **REQUIRED_WITH_DATA**, SetAgentState for returning to ready or not ready fails with an error code of **CF_WRAPUP_DATA_REQUIRED** (280) if there is no wrapup data entered into a call.
- If Logout Reason or NotReady Reasons are required, an error of **CF_REASON_CODE_REQUIRED** (281) is received if the reasons are not assigned in set agent state request. You must also create Logout Reason and NotReady Reason dialog boxes in the Reason Code if you require these properties.

For more information about reason code and wrapup modes, see the *Administration Guide for Cisco Unified Contact Center Enterprise*.

- The PG also uses the Supervisor Interface periodically to interrogate the switch to examine agent configuration change. The period interval is controlled by the Windows Registry entry “MonitorGroupTimerQuery”. If there is an agent skill group assignment change, the PG knows only when it next interrogates the switch.

UCCE Error Codes

The following table provides a brief description of the error message and what they indicate.

Table 44: Error Code Indicator

Error	Indicates
PERERR_TELDRIVE	The telephony driver layer generated the error.
PERERR_JTCLIENT	The JTAPI client generated the error.
PERERR_JTAPPLAY	The JTAPI application layer generated the error.
PERERR_GW_E	The JTAPI gateway generated the error.
PERERR_CM	Cisco Unified Communications Manager generated the error.

The following table lists error codes and their descriptions.



Note Some of these values appear over two lines due to space limitations.

Table 45: Error Code Description

Return Value/ Code	Error Message	Description
-1 PERERR_UNKNOWN	Unknown Peripheral Error.	The Peripheral error specified does not exist.
10001 PERERR_TELDRIVE_LOCKTPSERVICES	A logic error occurred prior to Locking TP Services.	The TP Services cannot be locked by the thread because they are already locked. This is a serious logic condition and should be reported/resolved.
10002 PERERR_TELDRIVE_LOCKINSTANCE	A logic error occurred prior to Locking the Client Instance.	The Client Instance cannot be locked by the thread because it is already locked. This is a serious logic condition and should be reported/resolved.
10003 PERERR_TELDRIVE_LOCKTELDRIVELAYER	A logic error occurred prior to Locking the Telephony Driver Layer.	The Telephony Driver Layer cannot be locked by the thread because it is already locked. This is a serious logic condition and should be reported/resolved.
10004 PERERR_TELDRIVE_NOINSTRUMENTFOR EXTENSION	The extension number specified is not associated with any known instrument.	An instrument with the number specified cannot be found for any instrument. Perhaps an invalid extension was specified.
10101 PERERR_TELDRIVE_AGENTALREADYLOGGEDOUT	The agent is already LOGGED out.	An attempt was made to log out an agent that is already logged out. This attempt failed.
10102 PERERR_TELDRIVE_AGENTALREADYSIGNEDON	The agent is already LOGGED ON.	An attempt was made to log in an agent that is already logged in. This attempt failed.
10103 PERERR_TELDRIVE_AGENTAVAILORWORK	The requested function cannot be performed since the agent is AVAILABLE or in a CALL WORK State.	This can occur when an agent tries to make a call from an AVAILABLE, or WORK state.
10104 PERERR_TELDRIVE_AGENTCANTGOUNAVAILABLE	The Agent cannot go UNAVAILABLE due to possible calls.	When this error occurs, the ROUTER did not approve the agent going unavailable. Typically retrying this makes it succeed.
10105 PERERR_TELDRIVE_AGENTNOTINATEAM	Agent is not a TEAM member– cannot make supervisor call.	The agent is trying to make a supervisor assist call but is not a member of a team.

Return Value/ Code	Error Message	Description
10106 PERERR_TELDRIVE_AGENTRESERVED	Agent is RESERVED – cannot make call.	This error occurs when the agent is trying to make a call or consult call but is currently RESERVED for an incoming call.
10107 PERERR_TELDRIVE_AGENTTEAMNOTFOUND	Internal Logic Error – Agent Team not found.	The agent team specified in the agent object cannot be found. This indicates an internal error that should be reported and resolved.
10108 PERERR_TELDRIVE_BADSTATETRANSITION	The state transition is invalid from the current state.	The routine ValidateAgentPrevalentStateTransition determined that the desired transition was illegal from the current state.
10109 PERERR_TELDRIVE_CALLTYPENOTVALIDFOR DIALPLAN	The agent is attempting to make a call that is not valid for their defined call plan.	The call type that the call was classified into is not allowed for the dialed Number Plan used.
10111 PERERR_TELDRIVE_CANTGOREADYFROM CURRENTSTATE	Cannot transition to READY from current state.	Based upon transition rules, the agent cannot go READY. Examples: You cannot go READY from TALKING.
10112 PERERR_TELDRIVE_CANTLOGOUTFROM CURRENTSTATE	The agent cannot log out from the current state.	The agent must be NOT READY in order to log out.
13042 PERERR_GW_E_THREADCLEARCALL_DROP_EXCEPTION	JTAPI Gateway – Error on CLEAR CALL operation – Exception.	The routine run in object ThreadClearCall got an exception (not of type CiscoJtapiException) on a call to "drop".
13044 PERERR_GW_E_THREADCLEARCONNECTION_UNKNOWN_CONNECTION	JTAPI Gateway – Error on CLEARCONNECTION operation – Unknown connection ID.	
13045 PERERR_GW_E_THREADCONFERENCECALL_ACTIVE_CONN_NOT_TALKING	JTAPI Gateway – Error on CONFERENCE operation – ACTIVE connection not in proper state.	The connection specified in the active connection is not in the TALKING state.
13046 PERERR_GW_E_THREADCONFERENCECALL_BAD_ACTIVE_CONNECTION	JTAPI Gateway – Error on CONFERENCE operation – ACTIVE connection not found.	
13047 PERERR_GW_E_THREADCONFERENCECALL_BAD_HELD_CONNECTION	JTAPI Gateway – Error on CONFERENCE operation – HELD connection not found.	
13048 PERERR_GW_E_THREADCONFERENCECALL_CREATECALL_NULL_CALL	JTAPI Gateway – Error on CONFERENCE operation.	The routine run in object ThreadConferenceCall got a null call returned from "createcall".

Return Value/ Code	Error Message	Description
13049 PERERR_GW_E_THREAD CONFERENCECALL_ EXCEPTION_ADDPARTY	JTAPI Gateway – Error on CONFERENCE operation.	The routine run in object ThreadConferenceCall got an exception (not of type CiscoJTapiException) on a call to "addparty".
13050 PERERR_GW_E_THREAD CONFERENCECALL_ EXCEPTION_CONFERENCE_NEW	JTAPI Gateway – Error on CONFERENCE operation.	The routine run in object ThreadConferenceCall got an exception (not of type CiscoJTapiException) on a call to "conference" for the NEW call.
13051 PERERR_GW_E_ THREADCONFERENCECALL_ EXCEPTION_CONFERENCE_HELD	JTAPI Gateway – Error on CONFERENCE operation.	The routine run in object ThreadConferenceCall got an exception (not of type CiscoJTapiException) on a call to "conference" for the HELD call.
13052 PERERR_GW_E_ THREADCONFERENCECALL_ EXCEPTION_CONSULT	JTAPI Gateway – Error on CONFERENCE operation.	The routine run in object ThreadConferenceCall got an exception (not of type CiscoJTapiException) on a call to "consult".
13053 PERERR_GW_E_ THREADCONFERENCECALL_ EXCEPTION_CREATECALL	JTAPI Gateway – Error on CONFERENCE operation.	The routine run in object ThreadConferenceCall got an exception (not of type CiscoJTapiException) on a call to "consult".
13054 PERERR_GW_E_ THREADCONFERENCE CALL_EXCEPTION_ SETCONFERENCEENABLE	JTAPI Gateway – Error on CONFERENCE operation.	The routine run in object ThreadConferenceCall got an exception (not of type CiscoJTapiException) on a call to "setconferenceenable".
13055 PERERR_GW_E_ THREADCONFERENCE CALL_EXCEPTION_ SETTRANSFERCONTROLLER	JTAPI Gateway – Error on CONFERENCE operation.	The routine run in object ThreadConferenceCall got an exception (not of type CiscoJTapiException) on a call to "settransfercontroller".
13056 PERERR_GW_E_THREAD CONFERENCECALL_ HELD_CONN_NOT_HELD	JTAPI Gateway – Error on CONFERENCE operation – HELD connection not HELD	The connection passed for the held connection is not in the HELD state.
13057 PERERR_GW_E_THREAD CONFERENCECALL_ NULL_DIALED_NUMBER	JTAPI Gateway – Error on CONFERENCE operation – Invalid Dialed Number.	A NULL dialed number was specified for the consultation number.
13058 PERERR_GW_E_THREAD CONSULTATIONCALL_ CREATECALL_NULL_CALL	JTAPI Gateway – Operation error on CONSULT operation.	The routine run in object ThreadConsultationCall got a null call returned from "createCall".

Return Value/ Code	Error Message	Description
13059 PERERR_GW_E_THREAD CONSULTATIONCALL_ EXCEPTION_CONSULT	JTAPI Gateway – Error on CONSULT operation.	The routine run in object ThreadConsultationCall got an exception on a call to "settransfercontroller".
13060 PERERR_GW_E_THREAD CONSULTATIONCALL_ EXCEPTION_CREATECALL	JTAPI Gateway – Error on CONSULT operation.	The routine run in object ThreadConsultationCall got an exception on a call to "createCall".
13061 PERERR_GW_E_THREAD CONSULTATIONCALL_ EXCEPTION_SET CONFERENCEENABLE	JTAPI Gateway – Error on CONSULT operation.	The routine run in object ThreadConsultationCall got an exception on a call to "setConferenceEnable".
13062 PERERR_GW_E_THREAD CONSULTATIONCALL_ INVALID_CONSULT_TYPE	JTAPI Gateway – Error on CONSULT operation – Invalid Consult type.	The type specified is not TRANSFER or CONFERENCE.
13063 PERERR_GW_E_THREAD CONSULTATIONCALL_ NO_ACTIVE_CONNECTION	JTAPI Gateway – Error on CONSULT operation – No Active Connection.	The ACTIVE connection specified in the request does not exist.
13064 PERERR_GW_E_THREAD ESCAPESEERVICE_ CREATECALL_NULL_CALL1	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got a NULL call returned from "createCall" (method "CreateNewCall" in class ThreadEscapeService).
13065 PERERR_GW_E_THREAD ESCAPESEERVICE_ CREATECALL_NULL_CALL2	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got a NULL call returned from "createCall" (method "CreateConsultCall" in class ThreadEscapeService).
13066 PERERR_GW_E_THREAD ESCAPESEERVICE_ CREATECALL_NULL_CALL3	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got a NULL call returned from "createCall" (method "CreateBlindConferenceCall" in class ThreadEscapeService).
13067 PERERR_GW_E_THREAD ESCAPESEERVICE_ EXCEPTION_CONFERENCE	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got an exception on a call to "conference" (method "CreateBlindConferenceCall" in class ThreadEscapeService).
13068 PERERR_GW_E_THREAD ESCAPESEERVICE_ EXCEPTION_CONNECT	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got an exception on a call to "connect" (method "CreateNewCall" in class ThreadEscapeService).
13069 PERERR_GW_E_THREAD ESCAPESEERVICE_ EXCEPTION_CONSULT1	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got an exception on a call to "consult" (method "CreateConsultCall" in class ThreadEscapeService).
13070 PERERR_GW_E_THREAD ESCAPESEERVICE_ EXCEPTION_CONSULT2	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got an exception on a call to "consult" (method "CreateBlindConferenceCall" in class ThreadEscapeService).

Return Value/ Code	Error Message	Description
13071 PERERR_GW_E_THREAD ESCAPESERVICE_ EXCEPTION_CREATECALL1	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got an exception on a call to "createCall" (method "CreateNewCall" in class ThreadEscapeService).
13072 PERERR_GW_E_THREAD ESCAPESERVICE_ EXCEPTION_CREATECALL2	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got an exception on a call to "createCall" (method "CreateConsultCall" in class ThreadEscapeService).
13073 PERERR_GW_E_THREAD ESCAPESERVICE_ EXCEPTION_CREATECALL3	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got an exception on a call to "createCall" (method "CreateBlindConferenceCall" in class ThreadEscapeService).
13074 PERERR_GW_E_THREAD ESCAPESERVICE_ EXCEPTION_GETADDRESS	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got an exception on a call to "getAddress" (method "CreateNewCall" in class ThreadEscapeService).
13075 PERERR_GW_E_THREAD ESCAPESERVICE_ EXCEPTION_GETTERMINALS	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got an exception on a call to "getTerminals" (method "CreateNewCall" in class ThreadEscapeService).
13076 PERERR_GW_E_THREAD ESCAPESERVICE_ EXCEPTION_SETCONFERENCEENABLE1	JTAPI Gateway – Error on SUPERVISOR (escape) operation	Got an exception on a call to "setConferenceEnable" (method "CreateConsultCall" in class ThreadEscapeService).
13077 PERERR_GW_E_THREAD ESCAPESERVICE_ EXCEPTION_SETCONFERENCEENABLE2	JTAPI Gateway – Error on SUPERVISOR (escape) operation	Got an exception on a call to "setConferenceEnable" (method "CreateBlindConference" in class ThreadEscapeService).
13078 PERERR_GW_E_THREAD ESCAPESERVICE_ INVALID_EMERGENCY_ ALERT_TYPE	JTAPI Gateway – Error on SUPERVISOR (escape) operation – Invalid Alert Type.	The Alert type specified was not CONSULT or BLIND_CONFERENCE.
13079 PERERR_GW_E_THREAD ESCAPESERVICE_ INVALID_SUPERVISOR_ ASSIST_TYPE	JTAPI Gateway – Error on SUPERVISOR (escape) operation – Invalid Alert Type.	The Alert type specified was not CONSULT or BLIND_CONFERENCE.
13080 PERERR_GW_E_THREAD ESCAPESERVICE_ NO_TERMINAL_LIST	JTAPI Gateway – Error on SUPERVISOR (escape) operation.	Got a NULL terminal list from "getTerminals" (method "CreateNewCall" in class ThreadEscapeService).
13081 PERERR_GW_E_THREAD HOLDCALL_ CALL_NOT_CONTROLLED	JTAPI Gateway – Error on HOLD operation – Uncontrolled Call.	The call specified is not a controlled call.

Return Value/ Code	Error Message	Description
13082 PERERR_GW_E_THREAD HOLDCALL_EXCEPTION_HOLD	JTAPI Gateway – Error on HOLD operation – Exception.	Got an exception on a call to "hold" (method "run" in class ThreadHoldCall).
13083 PERERR_GW_E_THREAD MAKECALL_CREATECALL_NULL_CALL	JTAPI Gateway – Error on MAKE CALL operation – Can't create call.	Got a NULL call returned from "createCall" (method "run" in class ThreadMakeCall).
13084 PERERR_GW_E_THREAD MAKECALL_CREATE_CALL_FAILURE	JTAPI Gateway – Error on MAKE CALL operation – Can't create call.	Got an exception on a call to "createCall" (method "run" in class ThreadMakeCall).
13085 PERERR_GW_E_THREAD MAKECALL_GENERIC_CM_ERROR	JTAPI Gateway – Error on MAKE CALL operation – Exception.	Got an exception on a call to "connect" (method "run" in class ThreadMakeCall).
13086 PERERR_GW_E_THREAD MAKECALL_NULL_TERMINAL_LIST	JTAPI Gateway – Error on MAKE CALL operation.	Got a NULL terminal list returned from "getTerminals" (method "run" in class ThreadMakeCall).
13087 PERERR_GW_E_THREAD MAKECALL_PROVIDER_GETADDRESS	JTAPI Gateway – Error on MAKE CALL operation.	Got an exception on a call to "getAddress" (method "run" in class ThreadMakeCall).
13088 PERERR_GW_E_THREAD MAKECALL_PROVIDER_GETTERMINAL	JTAPI Gateway – Error on MAKE CALL operation.	Got an exception on a call to "getTerminals" (method "run" in class ThreadMakeCall).
13089 PERERR_GW_E_THREAD REDIRECTCALL_EXCEPTION_REDIRECT	JTAPI Gateway – Error on REDIRECT operation – Exception.	Got an exception on a call to "redirect" (method "run" in class ThreadRedirectCall).
13090 PERERR_GW_E_THREAD RETRIEVECALL_CALL_NOT_CONTROLLED	JTAPI Gateway – Error on RETRIEVE operation – Uncontrolled Call.	The call specified is not a controlled call.
13091 PERERR_GW_E_THREAD RETRIEVECALL_EXCEPTION_UNHOLD	JTAPI Gateway – Error on RETRIEVE operation – Exception.	Got an exception on a call to "unhold" (method "run" in class ThreadRetrieveCall).
13092 PERERR_GW_E_THREAD SENDDTMF_EXCEPTION_GENERATEDDTMF	JTAPI Gateway – Error on SEND DTMF operation – Exception.	Got an exception on a call to "generateDTMF" (method "run" in class ThreadSendDTMF).
13093 PERERR_GW_E_THREAD SENDDTMF_INVALID_CONNECTION	JTAPI Gateway – Error on SEND DTMF operation – Invalid Connection ID.	The method "run" in class ThreadSendDTMF got a null connection from a call to "findTerminalConnection".
13094 PERERR_GW_E_THREAD SENDDTMF_NOT_MEDIATERMINAL_CONNECTION	JTAPI Gateway – Error on SEND DTMF operation – No Media.	

Return Value/ Code	Error Message	Description
13095 PERERR_GW_E_THREAD SUPERVISECALL_ACTIVE_CONN_NOT_TALKING	JTAPI Gateway – Error on SUPERVISE operation – ACTIVE connection not in proper state.	The connection specified in the active connection is not in the TALKING state.
13096 PERERR_GW_E_THREAD SUPERVISECALL_ALREADY_BARGED_IN	JTAPI Gateway – Error on SUPERVISE operation – Cannot Barge in, already barged into.	The call specified on the barge in request has already been barged into.
13097 PERERR_GW_E_THREAD SUPERVISECALL_CREATECALL_NULL_CALL	JTAPI Gateway – Error on SUPERVISE operation – Can't create call.	The routine run in object ThreadSuperviseCall got a null call returned from "createcall".
13098 PERERR_GW_E_THREAD SUPERVISECALL_EXCEPTION_ANSWER1	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "answer" (method "DirectSupervisorBargeIn" in class ThreadSuperviseCall).
13099 PERERR_GW_E_THREAD SUPERVISECALL_EXCEPTION_ANSWER2	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "answer" (method "BargeInBlindConferenceCall" in class ThreadSuperviseCall).
13100 PERERR_GW_E_THREAD SUPERVISECALL_EXCEPTION_CONFERENCE1	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "conference" (method "SupervisorBargeInCall" in class ThreadSuperviseCall).
13101 PERERR_GW_E_THREAD SUPERVISECALL_EXCEPTION_CONFERENCE2	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "conference" (method "DirectSupervisorBargeIn" in class ThreadSuperviseCall).
13102 PERERR_GW_E_THREAD SUPERVISECALL_EXCEPTION_CONSULT	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "conference" (method "DirectSupervisorBargeIn" in class ThreadSuperviseCall).
13103 PERERR_GW_E_THREAD SUPERVISECALL_EXCEPTION_CREATECALL	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "createCall" (method "DirectSupervisorBargeIn" in class ThreadSuperviseCall).
13104 PERERR_GW_E_THREAD SUPERVISECALL_EXCEPTION_DISCONNECT1	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "disconnect" (method "DropSupervisorCall" in class ThreadSuperviseCall).
13105 PERERR_GW_E_THREAD SUPERVISECALL_EXCEPTION_DISCONNECT2	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "disconnect" (method "InterceptCall" in class ThreadSuperviseCall).

Return Value/ Code	Error Message	Description
13106 PERERR_GW_E_THREAD SUPERVISECALL_EXCEPTION_SET CONFERENCEENABLE	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "disconnect" (method "DirectSupervisorBargeIn" in class ThreadSuperviseCall).
13107 PERERR_GW_E_THREAD SUPERVISECALL_HELD_CONN_NOT_HELD1	JTAPI Gateway – Error on SUPERVISE operation – HELD connection is not HELD.	The connection specified for the HELD call is not in the held state (method "BargInCall" class ThreadSuperviseCall).
13108 PERERR_GW_E_THREAD SUPERVISECALL_HELD_CONN_NOT_HELD2	JTAPI Gateway – Error on SUPERVISE operation – HELD connection is not HELD.	The connection specified for the HELD call is not in the held state (method "DirectSupervisorBargeIn" class ThreadSuperviseCall).
13109 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_ACTION	JTAPI Gateway – Error on SUPERVISE operation – Invalid action. The action specified was not CLEAR, BARGE_IN or INTERCEPT.	
13110 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_ACTIVE_CONNECTION	JTAPI Gateway – Error on SUPERVISE operation – No ACTIVE connection.	The connection specified in the active connection does not exist.
13111 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_AGENT_CALLID1	JTAPI Gateway – Error on SUPERVISE operation – Bad Call ID.	The call ID in the agent object is invalid (method "BargInCall" class ThreadSuperviseCall).
13112 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_AGENT_CALLID2	JTAPI Gateway – Error on SUPERVISE operation – Bad Call ID.	The call ID in the agent object is invalid (method "DirectSupervisorBargeIn" class ThreadSuperviseCall).
13113 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_AGENT_CONNECTION1	JTAPI Gateway – Error on SUPERVISE operation – Bad Connection ID.	The connection ID in the agent object is invalid (method "BargInCall" class ThreadSuperviseCall).
13114 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_AGENT_CONNECTION2	JTAPI Gateway – Error on SUPERVISE operation – Bad Connection ID.	The connection ID in the agent object is invalid (method "InterceptCall" class ThreadSuperviseCall).
13115 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_HELD_CONNECTION	JTAPI Gateway – Error on SUPERVISE operation – Invalid HELD connection.	The connection ID in the agent object is invalid (method "BargInCall" class ThreadSuperviseCall).
13116 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_SUPERVISOR_CONNECTION1	JTAPI Gateway – Error on SUPERVISE operation – Invalid Supervisor connection.	The connection ID in the agent object is invalid (method "DropSupervisorCall" class ThreadSuperviseCall).

Return Value/ Code	Error Message	Description
13117 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_SUPERVISOR_CONNECTION2	JTAPI Gateway – Error on SUPERVISE operation – Invalid Supervisor connection.	The connection ID in the agent object is invalid (method "BargeInCall" class ThreadSuperviseCall).
13118 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_SUPERVISOR_CONNECTION3	JTAPI Gateway – Error on SUPERVISE operation – Invalid Supervisor connection.	The connection ID in the agent object is invalid (method "DirectSupervisorBargeIn" class ThreadSuperviseCall).
13119 PERERR_GW_E_THREAD SUPERVISECALL_INVALID_SUPERVISOR_CONNECTION4	JTAPI Gateway – Error on SUPERVISE operation – Invalid Supervisor connection.	The connection ID in the agent object is invalid (method "BargeInBlindTransferCall" class ThreadSuperviseCall).
13120 PERERR_GW_E_THREAD SUPERVISECALL_SUPERVISOR_NOT_TALKING	JTAPI Gateway – Error on SUPERVISE operation – Supervisor Connection not TALKING.	The supervisor's connection is not in the talking state (method "DirectSupervisorBargeIn" class ThreadSuperviseCall).
13121 PERERR_GW_E_THREAD TRANSFERCALL_ACTIVE_CONN_NOT_TALKING	JTAPI Gateway – Error on SUPERVISE operation – Connection not TALKING.	The connection is not in the talking state (method "BargeInCall" class ThreadSuperviseCall).
13122 PERERR_GW_E_THREAD TRANSFERCALL_EXCEPTION_SETTRANSFER_CONTROLLER	JTAPI Gateway – Error on SUPERVISE operation – Exception.	The method "run" in class ThreadTransferCall got an exception on a call to "setTransferController".
13123 PERERR_GW_E_THREAD TRANSFERCALL_EXCEPTION_TRANSFER1	JTAPI Gateway – Error on SUPERVISE operation – Exception.	The method "run" in class ThreadTransferCall got an exception on a call to "transfer" with the HELD call specified.
13124 PERERR_GW_E_THREAD TRANSFERCALL_EXCEPTION_TRANSFER2	JTAPI Gateway – Error on SUPERVISE operation – Exception.	Got an exception on a call to "transfer" with the ACTIVE call specified (method "run" in class ThreadTransferCall).
13125 PERERR_GW_E_THREAD TRANSFERCALL_HELD_CONN_NOT_HELD	JTAPI Gateway – Error on TRANSFER operation HELD connection not HELD.	The connection passed for the held connection is not in the HELD state.
13126 PERERR_GW_E_THREAD TRANSFERCALL_INVALID_ACTIVE_CONNECTION	JTAPI Gateway – Error on TRANSFER operation – No ACTIVE.	The connection specified in the active connection does not exist.
13127 PERERR_GW_E_THREAD TRANSFERCALL_INVALID_HELD_CONNECTION	JTAPI Gateway – Error on TRANSFER operation Invalid HELD connection.	The connection ID in the agent object is invalid.

Return Value/ Code	Error Message	Description
20000 PERERR_CM_UNSPECIFIED	An unspecified Call Manager – error occurred on the operation.	
20001 PERERR_CM_TIMEOUT	A time-out Call Manager – occurred on the operation.	An operation exceeded the time limit that was configured/allocated for that operation.
20002 PERERR_CM_NO_ACTIVE_DEVICE_FOR_THIRDPARTY	Call Manager – Undescribed Error.	
20003 PERERR_CM_EXISTING_FIRSTPARTY	Call Manager – Line was specified that was not found.	
20004 PERERR_CM_ILLEGAL_HANDLE	Call Manager – Handle is unknown to the system.	
20005 PERERR_CM_UNDEFINED_LINE	Call Manager – Undescribed Error.	
20006 PERERR_CM_ILLEGAL_CALLINGPARTY	Call Manager – Attempt to originate call using a calling party that is not on the device.	
20007 PERERR_CM_CALL_ALREADY_EXISTS	Call Manager – Another call already exists on the line.	
20008 PERERR_CM_LINECONTROL_FAILURE	Call Manager – Line control refuses to let a new call because of its state (probably bug).	
20009 PERERR_CM_ILLEGAL_CALLSTATE	Call Manager – Line is not in a legal state to invoke the command.	
20010 PERERR_CM_CALLHANDLE_NOTINCOMINGCALL – Call Manager	Attempt to answer a call that either does not exist or is not in the correct state.	
20011 PERERR_CM_TRANSFERFAILED_DESTINATION_UNALLOCATED	Call Manager – Attempt to transfer to a directory number that is not registered.	
20013 PERERR_CM_TRANSFERFAILED_DESTINATION_BUSY	Call Manager – Attempt to transfer to a busy destination.	
20014 PERERR_CM_TRANSFERFAILED	Call Manager – Transfer failed.	Probable cause is one of the call legs was hung up or disconnected from the far end.

Return Value/ Code	Error Message	Description
20015 PERERR_CM_HOLDFAILED	CallManager – Hold was rejected by line control or call control.	
20017 PERERR_CM_RETRIEVE_FAILED	CallManager – Retrieve was rejected by line control or call control.	
20018 PERERR_CM_DB_NO_MORE_DEVICES	CallManager – Error No longer used.	
20020 PERERR_CM_DB_ILLEGAL_DEVICE_TYPE	CallManager – Error No longer used.	
20021 PERERR_CM_DB_ERROR	CallManager – Device query contained an illegal device type.	
20022 PERERR_CM_CANNOT_TERMINATE_MEDIA_ON_PHONE	CallManager – Media cannot be terminated by an application when the device has a physical phone (the phone always terminates the media).	
20025 PERERR_CM_UNKNOWN_GLOBAL_CALL_HANDLE	CallManager – Error no longer used.	
20026 PERERR_CM_DEVICE_NOT_OPEN	CallManager – Command issued on a line that must be open.	
20027 PERERR_CM_ASSOCIATED_LINE_NOT_OPEN	CallManager – Undescribed Error.	
20028 PERERR_CM_SSAPI_NOT_REGISTERED	CallManager – Redirect command was issued when the internal supporting interface was not initialized.	
20029 PERERR_CM_REDIRECT_CALL_DOES_NOT_EXIST	CallManager – Attempt to redirect a call that does not exist or is no longer active.	
20048 PERERR_CM_REDIRECT_CALLINFO_ERR	CallManager – Internal error returned from call control.	
20049 PERERR_CM_REDIRECT_ERR	CallManager – Internal error returned from call control.	
20050 PERERR_CM_REDIRECT_CALL_CALL_TABLE_FULL	CallManager – Internal error returned from call control.	
20051 PERERR_CM_REDIRECT_CALL_PROTOCOL_ERROR	CallManager – Internal error returned from call control.	

Return Value/ Code	Error Message	Description
20052 PERERR_CM_REDIRECT_CALL_UNKNOWN_DESTINATION	CallManager – Attempt to redirect to an unknown destination.	
20053 PERERR_CM_REDIRECT_CALL_DIGIT_ANALYSIS_TIMEOUT	CallManager – Internal error returned from call control	
20054 PERERR_CM_REDIRECT_CALL_MEDIA_CONNECTION_FAILED	CallManager – Internal error returned from call control.	
20055 PERERR_CM_REDIRECT_CALL_PARTY_TABLE_FULL	CallManager – Internal error returned from call control.	
20056 PERERR_CM_REDIRECT_CALL_ORIGINATOR_ABANDONED	CallManager – Far end hung up on the call being redirected.	
20057 PERERR_CM_REDIRECT_CALL_UNKNOWN_PARTY	CallManager – Internal error returned from call control.	
20058 PERERR_CM_REDIRECT_CALL_INCOMPATIBLE_STATE	CallManager – Internal error returned from call control.	
20059 PERERR_CM_REDIRECT_CALL_PENDING_REDIRECT_TRANSACTION	CallManager – Internal error returned from call control.	
20060 PERERR_CM_REDIRECT_CALL_UNKNOWN_ERROR	CallManager – Internal error returned from call control.	
20061 PERERR_CM_REDIRECT_CALL_NORMAL_CLEARING	CallManager – Internal error returned from call control.	
20062 PERERR_CM_REDIRECT_CALL_UNRECOGNIZED_MANAGER	CallManager – Internal error returned from call control.	
20063 PERERR_CM_REDIRECT_CALL_DESTINATION_BUSY	CallManager – Redirect destination is busy.	
20064 PERERR_CM_REDIRECT_CALL_DESTINATION_OUT_OF_ORDER	CallManager – Redirect destination is out of order.	
20065 PERERR_CM_CANNOT_OPEN_DEVICE	CallManager – Device open failed because the associated device is shutting down (unregistering).	
20066 PERERR_CM_TRANSFER_FAILED_OUTSTANDING_TRANSFER	CallManager – Existing transfer still in progress.	

Return Value/ Code	Error Message	Description
20067 PERERR_CM_TRANSFER_FAILED_CALLCONTROL_TIMEOUT	CallManager – Expected response from call control not received during a transfer.	
20068 PERERR_CM_CALLHANDLE_UNKNOWN_TO_LINECONTROL	CallManager – Attempt to redirect call that was unknown to line control.	
20069 PERERR_CM_OPERATION_NOT_AVAILABLE_IN_CURRENT_STATE	CallManager – Undescribed Error.	
20070 PERERR_CM_CONFERENCE_FULL	CallManager – Undescribed Error.	
20071 PERERR_CM_MAX_NUMBER_OF_CTI_CONNECTIONS_REACHED	CallManager – Undescribed Error.	
20080 PERERR_CM_INCOMPATIBLE_PROTOCOL_VERSION	CallManager – Undescribed Error.	
20081 PERERR_CM_UNRECOGNIZABLE_PDU	CallManager – QBE protocol error (bug).	
20082 PERERR_CM_ILLEGAL_MESSAGE_FORMAT	CallManager – QBE protocol error (bug).	
20094 PERERR_CM_DIRECTORY_TEMPORARY_UNAVAILABLE	CallManager – Undescribed Error.	
20095 PERERR_CM_DIRECTORY_LOGIN_NOT_ALLOWED	CallManager – Undescribed Error.	
20096 PERERR_CM_DIRECTORY_LOGIN_FAILED	CallManager – Login to the directory server failed when opening the provider.	
20097 PERERR_CM_PROVIDER_NOT_OPEN	CallManager – Attempt to issue a CTI command before the provider was open.	
20098 PERERR_CM_PROVIDER_ALREADY_OPEN	CallManager – Attempt to reopen a provider.	
20099 PERERR_CM_NOT_INITIALIZED	CallManager – Attempt to open a provider before CTI initialization completes.	
20100 PERERR_CM_CLUSTER_LINK_FAILURE	CallManager – Link failed to one of the call managers in the cluster (network error).	

Return Value/ Code	Error Message	Description
20101 PERERR_CM_LINE_INFO_DOES_NOT_EXIST	CallManager – Undescribed Error.	
20102 PERERR_CM_DIGIT_GENERATION_ALREADY_IN_PROGRESS	CallManager – Undescribed Error.	
20103 PERERR_CM_DIGIT_GENERATION_WRONG_CALL_HANDLE	CallManager – Undescribed Error.	
20104 PERERR_CM_DIGIT_GENERATION_WRONG_CALL_STATE	CallManager – Undescribed Error.	
20105 PERERR_CM_DIGIT_GENERATION_CALLSTATE_CHANGED	CallManager – Undescribed Error.	
20112 PERERR_CM_RETRIEVE_FAILED_ACTIVE_CALL_ON_LINE	CallManager – Undescribed Error.	
20113 PERERR_CM_INVALID_LINE_HANDLE	CallManager – Undescribed Error.	
20114 PERERR_CM_LINE_NOT_PRIMARY	CallManager – Undescribed Error.	
20115 PERERR_CM_CFWDALL_ALREADY_SET	CallManager – Undescribed Error.	
20116 PERERR_CM_CFWDALL_DESTN_INVALID	CallManager – Undescribed Error.	
20117 PERERR_CM_CFWDALL_ALREADY_OFF	CallManager – Undescribed Error.	
20119 PERERR_CM_DEVICE_OUT_OF_SERVICE	CallManager – Undescribed Error.	
20120 PERERR_CM_MSGWAITING_DESTN_INVALID	CallManager – Undescribed Error.	
20121 PERERR_CM_DARES_INVALID_REQ_TYPE	CallManager – Undescribed Error.	
20122 PERERR_CM_CONFERENCE_FAILED	CallManager – Undescribed Error.	
20123 PERERR_CM_CONFERENCE_INVALID_PARTICIPANT	CallManager – Undescribed Error.	

Return Value/ Code	Error Message	Description
20124 PERERR_CM_CONFERENCE_ALREADY_PRESENT	CallManager – Undescribed Error.	
20125 PERERR_CM_CONFERENCE_INACTIVE	CallManager – Undescribed Error.	
20126 PERERR_CM_TRANSFER_INACTIVE	CallManager – Undescribed Error.	
20153 PERERR_CM_COMMAND_NOT_IMPLEMENTED_ON_DEVICE	CallManager – Device does not support the command.	Undescribed Error.
20512 PERERR_CM_PROVIDER_CLOSED	CallManager – Undescribed Error.	
20513 PERERR_CM_PROTOCOL_TIMEOUT	CallManager – Undescribed Error.	
24095 PERERR_CM_GENERAL	CallManager – Unknown CallManager Failure on Operation.	An error response was received for a request issued to the call manager, but no error code could be extracted. This is always the case in the Encore Release. Please refer to the JTAPI log for more information.

Avaya Aura CC (Symposium)

- The Peripheral Gateway (and thus CTI OS clients) do not receive a CallEstablished Event for an off-switch call. As a result of this limitation, some features—such as blind conference or transfer operation off-switch—are not supported. The soft phone receives no notification that the call has been connected off-switch, and thus the application requires manual intervention from the agent (who heard a dial-tone, a ring, or an answer, and so forth) before completing the conference or transfer operation.
- The Transfer button is not enabled after an off-switch consult.
- Single-step/blind transfer or conference is not supported. Transfer and conference calls must be consultative.
- Consultative Transfer to a supervisor is not supported.
- Users cannot transfer to an AgentID.
- Users cannot put a conference or consultative call on hold, therefore the button is disabled.
- There is a delay when switching from the NotReady state to the Ready state.
- There is no equivalent to the AACC state WalkAway. The ACD gives a NOT_READY state to Unified ICM, but the switch rejects a request to set WalkAway to Not_Ready.
- Third-party call control and agent control requests issued through the CTI Server interface sometimes return a Peripheral error code in the failure indication message if the request fails. For the Avaya Aura CC (Symposium), this Peripheral error code is either a Status value or a Cause value. Generally, Status values are returned for call requests such as MakeCall and Cause values are returned for agent control

requests such as SetAgentState. The Avaya Aura CC (Symposium) Status and Cause values are defined in the two following tables.

- The ALTERNATE_CALL request is not supported with the Avaya Aura CC (Symposium) (for more information, see [Call Events, on page 95](#)).

Table 46: Avaya Status Values

Status Value (hex/dec)	Description
Invalid Parameters	
0A00 / 2560	Invalid calling TN
0A01 / 2561	Invalid calling DN; wrong DN specified
0A02 / 2562	Incomplete calling DN
0A03 / 2563	Invalid called DN
0A04 / 2564	Incomplete called DN
0A05 / 2565	Invalid called TN
0A06 / 2566	Invalid origination manner
0A07 / 2567	Invalid destination manner
0A08 / 2568	Invalid origination user type
0A09 / 2569	Invalid customer number
0A0A / 2570	System or data base error
Unsuccessful Call Origination	
0B00 / 2816	Origination party busy
0B01 / 2817	Origination resource blocking
0B02 / 2818	Origination set is maintenance busy
0B03 / 2819	500/2500 set is onhook
0B04 / 2820	Origination DN busy
0B05 / 2821	Origination is ringing
0B06 / 2822	Unable to disconnect origination (that is, already disconnected)
0B07 / 2823	Origination access restriction blocking
0B08 / 2824	Origination call on permanent hold
0B0A / 2826	System or data base error

Status Value (hex/dec)	Description
0B0B / 2827	Origination receiving end to end signaling
0B0C / 2828	The call is currently in an ACD queue
0B0E / 2830	Origination set invoked hold
0B14 / 2836	Transfer key not configured
0B15 / 2837	Transfer key not idle
0B16 / 2838	Set active in conference call
0B17 / 2839	Transfer or MPO/TSA class of service not configured
0B18 / 2840	Cannot put call on hold
0B1D / 2845	No active call exists on set
0B1E / 2846	No held call exists on set
Unsuccessful Call Termination	
0C00 / 3072	Terminating party is busy
0C01 / 3073	Destination resource blocking
0C02 / 3074	Destination in invalid state
0C07 / 3079	Destination access restriction blocking
0D0A / 3338	System or database error
Network Interceptions	
0C08 / 3080	Unassigned number
0C09 / 3081	No route to destination
0C0A / 3082	No user responding
0C0B / 3083	Number changed
0C0C / 3084	Destination out of service
0C0D / 3085	Invalid number format
0C0E / 3086	No circuit available
0C0F / 3087	Network out of order
0C10 / 3088	Temporary failure
0C11 / 3089	Equipment congestion
Network Interceptions with In-Band Information	

Status Value (hex/dec)	Description
0C19 / 3097	Terminating party is busy
0C1A / 3098	Unassigned number
0C1B / 3099	No route to destination
0C1C / 3100	No user responding
0C1D / 3101	Number changed
0C1E / 3102	Destination out of service
0C1F / 3103	Invalid number format
0C20 / 3104	No circuit available
0C21 / 3105	Network out of order
0C22 / 3106	Temporary failure
0C23 / 3107	Equipment congestion
0C24 / 3108	Interworking, unspecified
0CFE / 3326	Other cause
Unsuccessful Conference or Transfer Operation	
0D00 / 3328	Cannot complete conference
0D01 / 3329	Cannot initiate transfer
0D02 / 3330	Cannot complete transfer
0D03 / 3331	Cannot retrieve original call
0D04 / 3332	Fast Transfer initiation failed
0D05 / 3333	Fast Transfer completion failed
0D0B / 3339	Hold Request failed

Table 47: Avaya Cause Values

Cause Value (hex/dec)	Description
1002 / 4098	Access restricted
1003 / 4099	Resource unavailable
1004 / 4100	Invalid customer number
1005 / 4101	Invalid origination address

Cause Value (hex/dec)	Description
1006 / 4102	Invalid destination address
1007 / 4103	Invalid manner
1008 / 4104	Unsuccessful retrieve original
1009 / 4105	Unsuccessful transfer
100A / 4106	Unsuccessful conference
100B / 4107	Unsuccessful answer request
100C / 4108	Unsuccessful release request
1070 / 4208	Refer to Connection Status IE
2004 / 8196	The target DN is invalid
2005 / 8197	The target DN is not AST
2006 / 8198	The Customer Number is invalid
2007 / 8199	The feature could not be invoked
2008 / 8200	The feature is not configured on the set
2009 / 8201	The requested feature is out of valid range
200A / 8202	The target set is not ACD agent
200B / 8203	The target set is a Virtual Agent
200C / 8204	The set is maintenance busy
200D / 8205	Set is in wrong state for invocation
200E / 8206	Set is in target state
200F / 8207	No NRDY/RDY while ACD set is logged out
2010 / 8208	Package C customer cannot use NRDY with IDN call
2011 / 8209	Feature IE is missing or invalid
2012 / 8210	DN IE is missing or invalid
2013 / 8211	Agent ID IE is missing or invalid
2014 / 8212	Agent ID is invalid
2015 / 8213	CFW DN IE is invalid
2016 / 8214	The Call Forward DN is too long
2017 / 8215	The Call Forward DN is invalid

Cause Value (hex/dec)	Description
2018 / 8216	User is invoking Call Forward
2019 / 8217	MSB/MSI not supported for 500/2500 sets
201A / 8218	500/2500 ACD agent already changed status
201B / 8219	500/2500 ACD agent set is being rung
201C / 8220	User is manually logging in 500 /2500 ACD set

Swap Feature in Avaya Aura Contact Center (Symposium) ACD

The Swap feature enables the agents to swap or alternate between customer calls and consult calls, both from hardphones as well as softphones.

The Swap feature deploys a CTI toolbar with Unified ICM, offering most of the phone set functionalities. One of the most important functionalities is that it allows the agent to swap or alternate between primary and consult calls during a Consultation Call.

The agent performing the transfer must carry out a swap, or alternate between the primary key (ACD or DN) and the secondary key of transfer. On the phone set, an agent can perform a swap by using the transfer or primary key of the used line (ACD or DN).



Note The Swap feature is not supported when CTI OS is used with the Avaya Aura Contact Center (Symposium).

Dependencies and Patches for Swap Feature Support in Softphones and Hardphones

The following patches are required for Swap feature support.

Symposium SCCS 5.0:

- SU 05
- SUS0501/02/03
- NN_SCCS_5.0_DP_050302_S (mandatory)
- NN_SCCS_5.0_DP_050301_S (optional)

NCCM 6.0:

- SU03
- SUS0301
- PEP_030130_RU

Nortel CS1000 Succession 4.0 or 4.5:

- MPLR20429
- MPLR21764

Enabling Swap Feature on Unified ICM

You can enable the Swap feature with the help of Config REGISTRY Key called NortelSwapPatchInstalled. This key is created when you install the patch. Set the value of this registry key to 1 before starting the PG.

If there are multiple instances of the Avaya Aura Contact Center PG in the same box, you must set the registry NortelSwapPatchInstalled to 1 for all the PG instances. This allows the CTI OS Server to enable the alternate button on the client desktop.

Agent States

This section presents the agent-state terminology and functionality used by CTI OS Server and how it corresponds to the terminology and functionality of various call center peripherals.

Table 48: Agent State Functionality and Call Center Terminology

State	Peripheral-Specific Equivalent
<p>Available</p> <p>The agent is ready to accept a call.</p>	<p>Aspect Contact Server: Avail</p> <p>Avaya DEFINITY ECS: AVAIL</p> <p>Avaya Aura CC (Symposium): Idle</p>
<p>BusyOther</p> <p>The agent is busy performing a task associated with another active Skill Group.</p>	<p>Aspect Contact Server: MSG (if Aspect Event Link is not being used)</p> <p>Avaya DEFINITY ECS: OTHER</p> <p>Avaya Aura CC (Symposium): No equivalent</p>
<p>Hold</p> <p>The agent currently has all calls on hold.</p>	<p>Aspect Contact Server: HOLD</p> <p>Avaya DEFINITY ECS: No equivalent</p> <p>Avaya Aura CC (Symposium): On Hold, On Hold Walkaway</p>
<p>Login</p> <p>The agent has logged in to the ACD. It does not necessarily indicate that the agent is ready to accept calls.</p>	<p>Although viewed as a state by CTI Server, this state is more of an event, and is not treated as a state by the switches.</p>
<p>Logout</p> <p>The agent has logged out of the ACD and cannot accept any additional calls.</p>	<p>Aspect Contact Server: Signed Off</p> <p>Avaya DEFINITY ECS: No equivalent</p> <p>Avaya Aura CC (Symposium): Logout</p>
<p>NotReady</p> <p>The agent is logged in but is unavailable for any call work.</p>	<p>Aspect Contact Server: Idle</p> <p>Avaya DEFINITY ECS: AUX</p> <p>Avaya Aura CC (Symposium): Not Ready Walkaway (however, this state requires the agent to click Hold and physically unplug the headset – because a physical act is involved, a software request to set the agent state to NotReady fails), Emergency</p>

State	Peripheral-Specific Equivalent
<p>Reserved</p> <p>The agent is reserved for a call that arrives at the ACD shortly.</p>	<p>Aspect Contact Server: RSVD</p> <p>Avaya DEFINITY ECS: No equivalent</p> <p>Avaya Aura CC (Symposium): Call Presented</p>
<p>Talking</p> <p>The agent is currently talking on a call (inbound, outbound, or inside).</p>	<p>Aspect Contact Server: Talking ACD1, Talking ACD2, Talking ACT1, Talking ACT2, Talking Out1, Talking Out2, Talking Inside, Supervisor Line, MSG, HELP (MSG and HELP correspond to Talking only if Aspect Event Link is being used.)</p> <p>Avaya DEFINITY ECS: AUX-IN, AUX-OUT, ACD-IN, ACD-OUT, ACW-IN, ACW-OUT, DACD</p> <p>Avaya Aura CC (Symposium): Active, Consultation</p>
<p>Unknown</p> <p>The agent state is currently unknown.</p>	<p>Aspect Contact Server: No equivalent</p> <p>Avaya DEFINITY ECS: UNKNOWN</p> <p>Avaya Aura CC (Symposium): No equivalent</p>
<p>WorkNotReady</p> <p>The agent is performing after-call work and is not ready to receive a call after the work is complete.</p>	<p>Aspect Contact Server: No equivalent</p> <p>Avaya DEFINITY ECS: No equivalent</p> <p>Avaya Aura CC (Symposium): No equivalent</p>
<p>WorkReady</p> <p>The agent is performing after-call work and is ready to receive a call after the work is complete.</p>	<p>Aspect Contact Server: Wrap-up</p> <p>Avaya DEFINITY ECS: ACW, DACW</p> <p>Avaya Aura CC (Symposium): Not Ready, Break, Busy</p>



CHAPTER 11

Cisco Unified Mobile Agent

- [Log in to CTI OS Agent Desktop, on page 141](#)
- [Verify Login, on page 142](#)
- [Enable Ready State, on page 142](#)
- [Transfer a Call, on page 142](#)

Log in to CTI OS Agent Desktop

Perform the following steps to log in to the CTI OS Agent Desktop.

Procedure

Step 1 From the desktop, click **Login**.
The CTI Login dialog box appears.

Step 2 In the **CTI Login** dialog box, enter the following information in the corresponding fields:

Option	Description
Mobile Agent	You must select this check box to log in as a Mobile Agent.
Phone Number	The dial number for the phone the agent intends to use. Note The format for the phone number must follow the dial plan, for example, 91201-123-xxxx.
Call Mode	Select nailed connection.

Step 3 Click **OK**.

The desktop automatically enters the state that is configured on the switch (either Ready or Not Ready) and the buttons for actions for that state are enabled.

Note For a nailed connection, the desktop must receive and answer a setup call before agent login is complete.

Verify Login

Perform the following procedure to verify your login.

Procedure

Step 1 Check to be sure that your desktop is in the Ready or Not Ready state.

Step 2 Check to be sure the status bar of your Unified Mobile Agent Desktop displays the following:

- Agent ID for the logged-in agent
- Agent Extension
- Agent Instrument
- Current Agent Status
- The server that the desktop is connected to

Step 3 Check to be sure the action buttons that are allowed for your current agent state are enabled.

Note If you log in as a Mobile Agent and want to make a phone call from CTI OS Agent Desktop, you must use the CTI Dialing Pad on the desktop.

Enable Ready State

Procedure

If you are in the Not Ready state and the **Ready** button is enabled, click the **Ready** button.

Transfer a Call

Procedure

Step 1 Click the **Transfer** button.
The CTI Dialing Pad dialog box appears.

Step 2 Enter the phone number to be dialed in the Dialed Number field or select a destination from the pull-down menu.

The pull-down menu contains the last six numbers dialed from this desktop.

Step 3 Optionally, click the **More** button to display the Call Data tab, where you can optionally enter data associated with the call.

Step 4 Do one of the following:

Option	Description
If you <i>do want to speak</i> with the consulted agent, click the Transfer Init button.	When you press the Transfer Init button, the call is put on hold. The agent has an opportunity to speak to the consulted agent before completing the transfer. When the consult call is answered, the button changes to Transfer Complete. To complete the transfer, click Transfer Complete .
If you <i>do not want to speak</i> with the consulted agent, click Single Step .	The call automatically transfers.



APPENDIX **A**

Ethernet Card Testing

- [Ethernet Cards for Silent Monitor, on page 145](#)
- [Test Procedure, on page 145](#)

Ethernet Cards for Silent Monitor

On a site with IP telephony, the Unified CM and the IP Phones normally use a Virtual Local Area Network (VLAN) that logically separates voice from data. Although both traffic types are on the same physical channel, they are sent on different VLANs, one for voice and other for data. This configuration enables you to send voice with higher priority than data.

In a call center with silent monitoring, the agent desktop system uses one single physical channel to interact with two different VLANs. You connect the agent desktop system to the PC port on the back of the IP phone. Then, the silent monitor subsystem can collect the voice packets reaching the phone and forward the packets to the supervisor workstation.

The agent desktop system accesses the physical channel through an Ethernet Network Interface Controller (NIC). The NIC monitors the channel and collects Ethernet frames addressed to the agent's computer. The NIC runs a preprocessing step to extract IP packets from the Ethernet frames and deliver them to the TCP/IP stack on the operating system.

During internal testing, Cisco identified that some Ethernet NIC card drivers cannot preprocess Ethernet frames that have an IP packet encapsulated in a VLAN frame. The NIC card driver discards the Ethernet frame if the IP packet is encapsulated in an 802.1Q frame. Some vendors can provide a configuration setting that allows their NIC card driver to forward VLAN traffic to the TCP/IP stack.

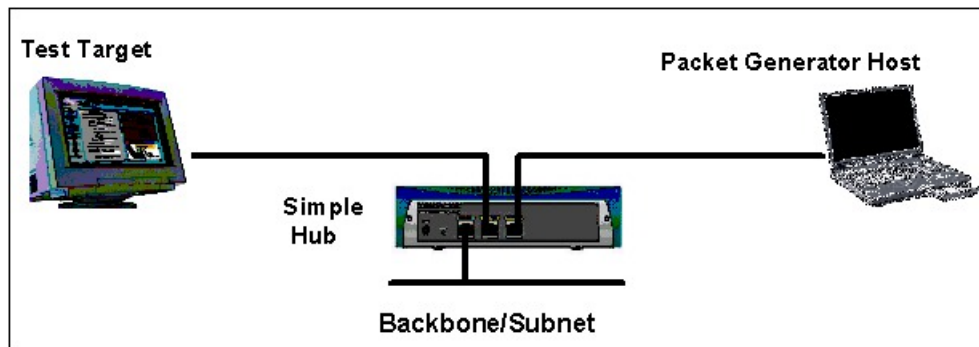
If an agent desktop's NIC card driver discards VLAN traffic, then the silent monitor subsystem on that desktop cannot collect and forward voice packets. Silent monitor cannot function properly on such a NIC. Cisco developed a procedure to determine if a particular Ethernet NIC card driver works with the CTI OS silent monitor. The procedure is described in the following sections.

Test Procedure

The test involves sending sample VLAN packets to a *Test Target NIC* card and verifying that the packets are not discarded by the pre-processing step but are passed onto the TCP/IP stack on the operating system at the computer hosting the NIC card.

The test requires a configuration as shown in the following diagram.

Figure 13: Silent Monitor Ethernet Card Test Configuration



The Test Target NIC is connected to one port of a simple Hub. The Hub is connected to the network backbone or subnet. You also need a *Packet Generator Host* capable of generating Ethernet traffic. You must connect the *Packet Generator Host* to another port on the Hub.

The *Packet Generator Host* equipment can be either a dedicated packet analyzer or a computer with a software-based packet analyzer with capabilities to generate Ethernet traffic.

You can use several available software packet analyzers that can be used for this purpose. For more information about reliable analyzers, visit the *Cooperative Association for Internet Data Analysis* website at <http://www.caida.org/tools/taxonomy/workload.xml>.

The following sections demonstrate the use of Sniffer Pro.

After you set up the environment as described above you must load the software tools on the *Test Target* and *Packet Generator Host* as follows.

Prepare Test Target

Procedure

- Step 1** Install the *WinPcap* utility. The WinPcap installation program is located at the root directory on the Cisco Computer Telephony Integration CTI Object Server CD.
- Step 2** Create a directory on the *Test Target* computer named "VLANTest".
- Step 3** From the Cisco Computer Telephony Integration CTI Object Server CD, copy WinDump.exe and place it in the directory you created in Step 2. (*WinDump* is located on the CD under *CtiOS\Tools\VLANTest\WinDump*.)
- Step 4** Open a console window. Go to the directory where you copied WinDump.exe.
- Step 5** Determine the MAC address of the *Test Target* NIC by executing `ipconfig /all` at the command prompt. Write down the number that appears for the Physical Address. For example, the "Intel Pro/100" NIC card has a MAC address of **00D059d8f7d9**.

Figure 14: Determining the Test Target NIC MAC Address

```

C:\Select C:\WINNT\system32\cmd.exe
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
DNS Suffix Search list. . . . : cisco.com

Ethernet adapter Local Area Connection 2:

    Connection-specific DNS Suffix . : cisco.com
    Description . . . . . : Cisco Systems 350 Series PCMCIA Wir
    Physical Address. . . . . : 00-09-43-74-55-94
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . : Yes
    IP Address. . . . . : 10.86.165.239
    Subnet Mask . . . . . : 255.255.254.0
    Default Gateway . . . . . : 10.86.164.1
    DHCP Server . . . . . : 161.44.124.23
    DNS Servers . . . . . : 161.44.124.122
    . . . . . : 64.102.6.247
    . . . . . : 171.68.226.120
    Primary WINS Server . . . . . : 161.44.122.10
    Secondary WINS Server . . . . . : 64.102.2.51
    Lease Obtained. . . . . : Friday, August 08, 2003 5:39:41 PM
    Lease Expires . . . . . : Saturday, August 09, 2003 1:39:41 P

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix . : cisco.com
    Description . . . . . : Intel(R) PRO/100 VE Network Connect
    Physical Address. . . . . : 00-09-43-74-55-94
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . : Yes
    IP Address. . . . . : 10.86.139.153
    Subnet Mask . . . . . : 255.255.255.128
    Default Gateway . . . . . : 10.86.139.129
  
```

Step 6 Determine the device interface number of the *Test Target NIC*. Execute **windump -D** and write down the number of the NIC you want to test. In this example, you would choose interface number 1, which corresponds to the “Intel Pro/100” NIC card.

Note If you are not sure which number to pick, repeat the test for each card until the test succeeds for one (sufficient to pass) or this fails for all cards.

Step 7 Start WinDump to monitor the *Test Target NIC* for incoming VLAN packets. To do this execute **windump -i <device_number> vlan**. In the following example the *device_number* is 1.

Figure 15: Monitoring the Test Target NIC for Incoming VLAN Packets

```

C:\WINNT\system32\cmd.exe - windump -i 1 vlan
D:\Development\VLAN Testing\WinDump>windump -i 1 vlan
windump: listening on \Device\NPF_{5E18F304-4257-46C3-9ADD-A39EDC591C3C}
  
```

Prepare Packet Generator Host

Perform the following steps to prepare the packet generator host.

Procedure

Step 1 Load the packet analyzer software onto your *Packet Generator Host*.

Step 2 Load the sample capture file provided in the Cisco Computer Telephony Integration CTI Object Server CD (C:\Tools\VLANTest\VLANCapture\VLANSamplePackets.cap). The capture file was generated in a format that is used by most dedicated and software packet analyzers.

Step 3 Select the Decode view from the tab at the bottom of the screen.

Executing a Test

The test involves sending sample VLAN packets to a *Test Target NIC* card and verifying that the packet is not discarded by the pre-processing step but is passed onto the TCP/IP stack on the computer hosting the NIC card.

The test case to determine whether or not the *Test Target NIC* is qualified to work with CTI OS silent monitor is as follows. (In the test case nomenclature, PA stands for Packet Analyzer and WD stands for WinDump.)

Table 49: SMNIC- 1 Send Sample VLAN Packets to Test Target NIC Card

Objective	Verify that the Test Target NIC can pre-process VLAN packets and forward them to the TCP/IP stack on the Test Target Host.	
Steps	Party	Action
1	PA	Select one of the loaded sample VLAN Packets.
2	PA	Select or right-click “Send Current Frame”.
3	PA	Modify the destination MAC address to use the MAC address of the Test target NIC (for more information, see the figure “Modifying the destination MAC address” below).
4	PA	Send the new frame to the Test Target NIC five times.
5	WD	Verify that there is activity reported on the Test Target NIC.
Expected Result	At the <i>Test Target</i> computer <i>windump</i> displays five packets for VLAN ID = 85 (for more information, see the figure “Sample output showing successful packet capture” below). If the test fails, no packets appear.	

Figure 16: Modifying the Destination MAC Address

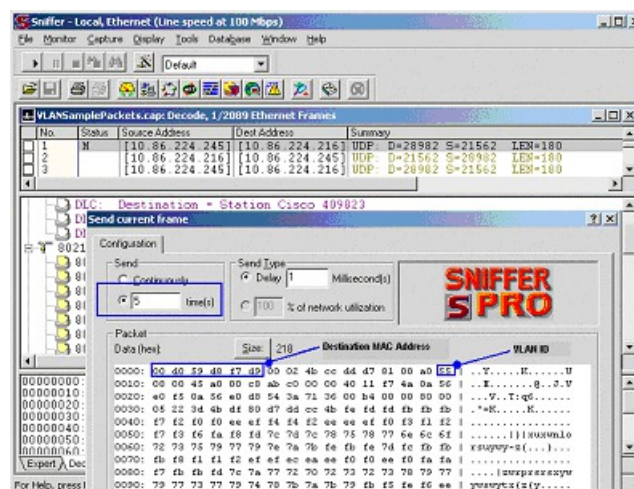
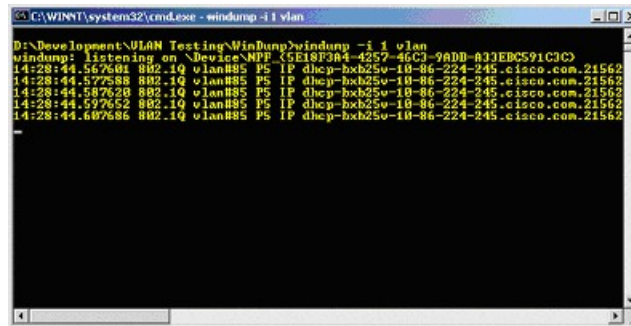


Figure 17: Sample Output Showing Successful Packet Capture



```
C:\WINNT\system32\cmd.exe - windump -i 1 vlan
D:\Development\DLAN_Testing\WinDump>windump -i 1 vlan
windump: listening on device\NDP {5E187D84-4277-46C3-9ADD-813EDC591C3C}
14:28:44.567601 802.1Q vlan885 PS IP dhcp-bxb250-10-86-224-245.cisco.com.21562
14:28:44.577688 802.1Q vlan885 PS IP dhcp-bxb250-10-86-224-245.cisco.com.21562
14:28:44.587628 802.1Q vlan885 PS IP dhcp-bxb250-10-86-224-245.cisco.com.21562
14:28:44.597652 802.1Q vlan885 PS IP dhcp-bxb250-10-86-224-245.cisco.com.21562
14:28:44.607686 802.1Q vlan885 PS IP dhcp-bxb250-10-86-224-245.cisco.com.21562
```

If the outcome of this test is successful, then your *Test Target NIC* works with the CTI OS silent monitor. Otherwise, contact your NIC card provider and ask what settings are necessary to allow your NIC card driver to forward all packets including VLAN packets to the TCP/IP stack on the computer so that your packet analyzer tool can capture and display them. Then apply the appropriate adjustments and rerun this test procedure.

