

# External Calls Appear as Internal in Cisco IPCC Statistics

Document ID: 67988

---

**Introduction**

**Prerequisites**

Requirements

Components Used

Conventions

**Problem**

**Cause**

**Solution**

**NetPro Discussion Forums – Featured Conversations**

**Related Information**

---

## Introduction

This document describes one reason why outbound calls appear as inbound or internal calls in the statistics, and provides a workaround in a Cisco IP Contact Center (IPCC) Enterprise environment.

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of these topics:

- Cisco Intelligent Contact Management (ICM)
- Cisco CallManager
- Cisco IPCC

### Components Used

The information in this document is based on these software and hardware versions:

- Cisco ICM version 5.x and later
- Cisco CallManager version 4.x

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

### Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

## Problem

When an agent makes an outbound call, the call appears as an internal or inbound call in reports and statistics.

## Cause

This problem occurs due to a change in Cisco JTAPI version. In earlier versions of Cisco JTAPI, when a caller calls an address outside the cluster, Cisco CallManager delivers the CallCtlConnNetworkReachedEv and CallCtlConnNetworkAlertingEv events for the far-end address.

Cisco CallManager version 4.0 and later do not deliver these events. In these versions, the CallCtlConnection for the far-end address goes to the ESTABLISHED state from the OFFERED state. The application receives the CallCtlConnOfferedEv and CallCtlConnEstablishedEv events for the far-end address. The application does not receive the CallCtlConnNetworkReachedEv and CallCtlConnNetworkAlertingEv events.

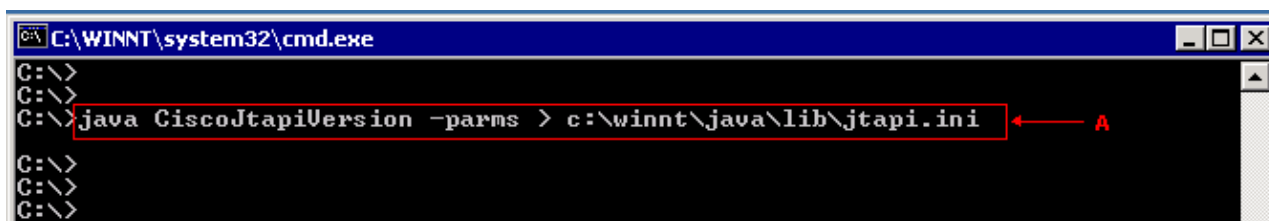
In order to receive the network events, you must turn on the "Allow overlap sending" flag on the route pattern configured for the gateway. A new jtapi.ini parameter called "AllowNetworkEventsAfterOffered" is introduced to allow the application to control the delivery of these events. Applications that need the network events but cannot turn on this flag can use this new ini parameter to receive network events for outgoing calls.

## Solution

Complete these steps to turn on the parameter:

1. Click **Start > Run**, and type **cmd** in the Open field of the Run dialog box.
2. Check whether you have installed Cisco JTAPI in the default directory. If so, type **java CiscoJtapiVersion -parms > c:\winnt\java\lib\jtapi.ini** to create a jtapi.ini file, which contains the .ini parameters for the current CiscoJTAPI version, and copies the parameters to jtapi.ini (see arrow A in Figure 1).

**Figure 1 Creation of Jtapi.ini File**



3. Edit c:\winnt\java\lib\jtapi.ini with notepad or any other text editor. Here is a sample of this file:

```
#Cisco Jtapi version 1.4(3.19) Release ini parameters
#Fri Apr 29 11:12:48 EDT 2005
PROTOCOL_DEBUGGING=0
UseSameDirectory=1
JTAPIIMPL_DEBUGGING=0
UseSystemDotOut=0
QueueStatsEnabled=0
PeriodicWakeupInterval=50
CMAssignedAppID=0
RouteSelectTimeout=5000
UseTraceFile=1
ProviderOpenRequestTimeout=200
CtiManagers=
Directory=
DEBUG=0
DesiredServerHeartbeatInterval=30
AlarmServicePort=1444
CTI_DEBUGGING=0
SyslogCollector=
JTAPI_DEBUGGING=0
PeriodicWakeupEnabled=0
```

```
NumTraceFiles=10
AlarmServiceHostname=
MISC_DEBUGGING=0
TracePath=.
UseAlarmService=0
CTIIMPL_DEBUGGING=0
WARNING=0
Traces=WARNING;INFORMATIONAL;DEBUG
INFORMATIONAL=0
UseSyslog=0
JtapiPostConditionTimeout=15
JTAPINotificationPort=789
FileNameBase=CiscoJtapi
CtiRequestTimeout=30
TraceFileSize=1048576
Debugging=JTAPI_DEBUGGING;JTAPIIMPL_DEBUGGING;CTI_DEBUGGING;
CTIIMPL_DEBUGGING;PROTOCOL_DEBUGGING;MISC_DEBUGGING
FileNameExtension=log
QueueSizeThreshold=25
ProviderRetryInterval=30
SyslogCollectorUDPPort=514
```

4. Add this line to the end of the file:

```
AllowNetworkEventsAfterOffered=1
```

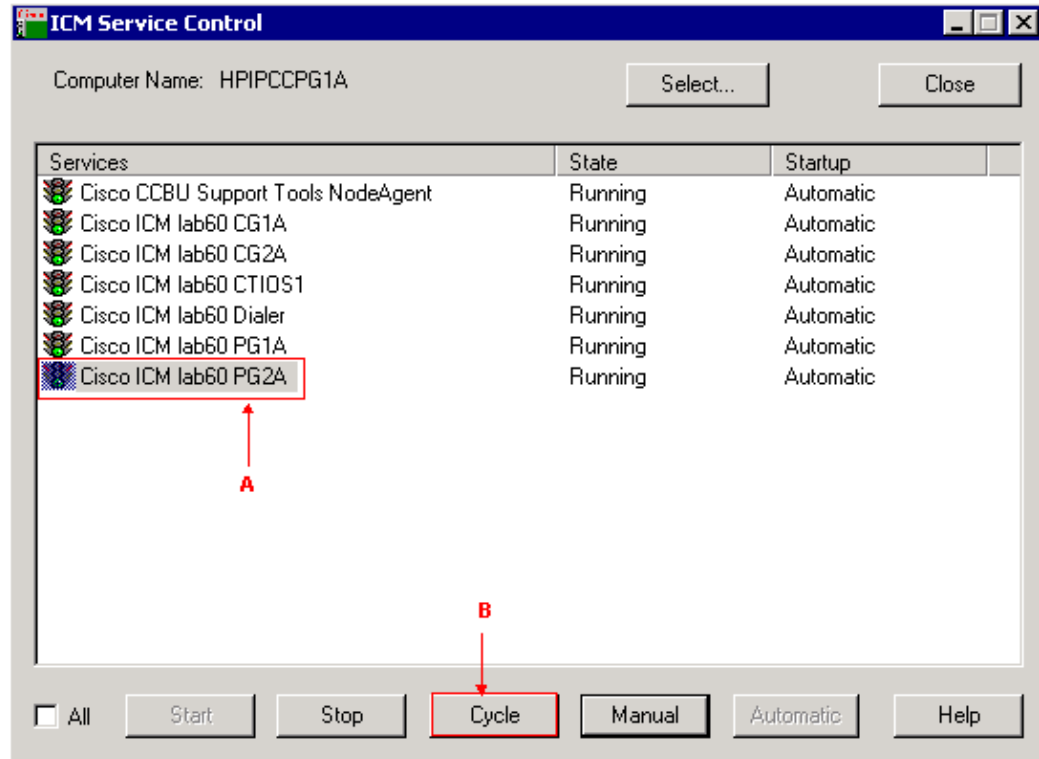
**Note:** Ensure that this line is independent. Do not append this line to any other line in the file.

5. Save the modified jtapi.ini file.

6. Cycle the CallManager PG, because the CallManager PG only reads jtapi.ini at the initialization time. Complete these steps:

- a. Run ICM Service Control, and select the CallManager PG (see arrow A in Figure 2).
- b. Select the CallManager PG (see arrow A in Figure 2).
- c. Click **Cycle** (see arrow B in Figure 2). Alternatively, click **Stop**, and then click **Start** after the process completely stops.

## Figure 2 ICM Service Control



7. Copy the jtapi.ini file to the other side of the PG into the c:\winnt\java\lib directory.
8. Cycle the other side of the PG.

**Note:** Perform these steps every time you reinstall the Cisco JTAPI client on the Peripheral Gateway.

When the AllowNetworkEventsAfterOffered flag is enabled, the application receives these events for the far end address:

- CallCtlConnOfferedEv
- CallCtlConnNetworkReachedEv or CallCtlConnNetworkAlertingEv
- CallCtlConnEstablishedEv

After this change, external calls appear correctly in reports and statistics.

## NetPro Discussion Forums – Featured Conversations

Networking Professionals Connection is a forum for networking professionals to share questions, suggestions, and information about networking solutions, products, and technologies. The featured links are some of the most recent conversations available in this technology.

NetPro Discussion Forums – Featured Conversations for Customer Contact Software

IP Communications and Video: Contact Center

## Related Information

- [Technical Support & Documentation – Cisco Systems](#)

