XML Services Troubleshooting

This chapter contains the following topics:

Overview

Extensible Markup Language (XML) services are applications for IP phones in Cisco CallManager environments. For example, an XML service can be used to display directory listings or to submit billing account numbers to be associated with calls. XML is a text-based markup language that is used to create documents that convey object information and the hierarchical structure of that information. XML services for IP phones can include the display of text or graphic information.

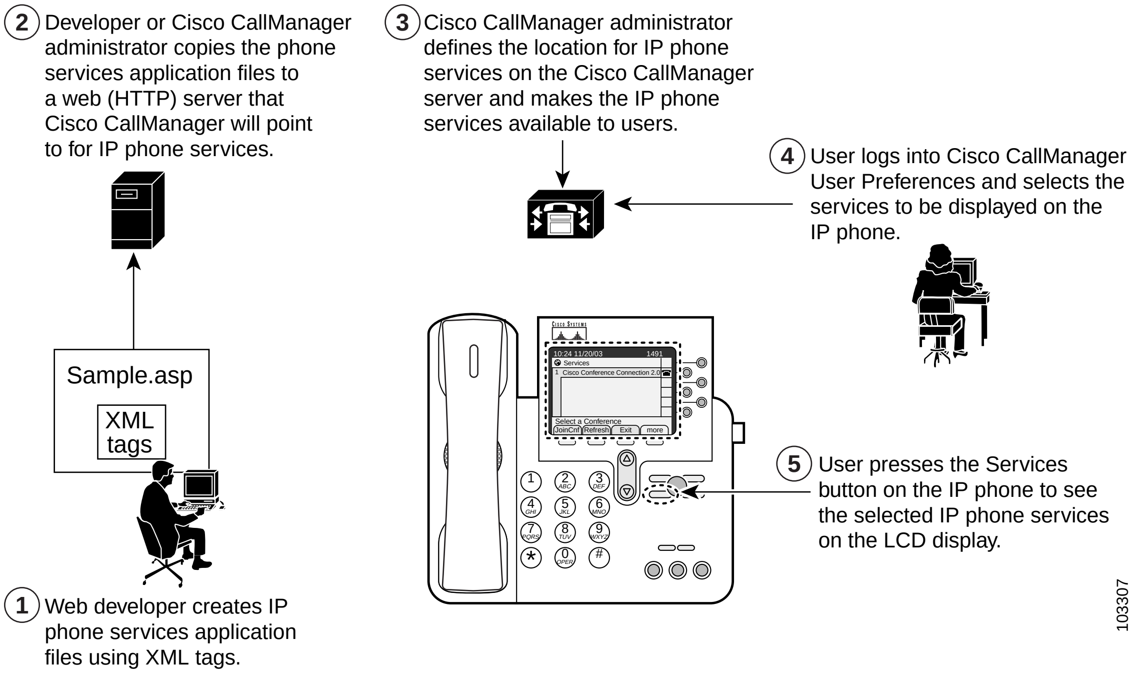
When an IP phone in a Cisco CallManager system uses a web-based XML service, it communicates with a web server using HTTP, the same protocol that PC web browsers use. Note that the applications themselves are written in XML and that Cisco IP phones do not understand HTML. For normal phone transactions, IP phones use other protocols to communicate. Messages between Cisco CallManager and an IP phone are sent using Skinny Client Control Protocol (SCCP). Voice packets are sent between phones connected in a call using Real-Time Transport Protocol (RTP), a UDP-based protocol especially designed for sending voice, video, and other time-sensitive data across packet networks.

An IP phone begins to invoke a service when the phone user presses the Services button. By default the Services button is assigned a URL that points to a web page called GetServicesMenu.asp on the Cisco CallManager server, although this default can be changed by using one of the Cisco CallManager administration screens.

The GetServicesMenu.asp web page displays a menu of phone services that have been specified for this phone. Each item on the service menu points to another URL on the web server where the XML application files for that service are stored. Figure 17 on page 28 illustrates the life cycle of an XML phone service application.

For more information, refer to the "Cisco IP Phone Services" chapter of the *Cisco CallManager System Guide* and the "Cisco IP Phone Services Configuration" chapter of the *Cisco CallManager Administration Guide.*

Figure 17 XML Services Life Cycle



2 Developer or Cisco CallManager administrator copies the phone services application files to a web (HTTP) server that Cisco CallManager will point to for IP phone services.

3 Cisco CallManager administrator defines the location for IP phone services on the Cisco CallManager server and makes the IP phone services available to users.

4 User logs into Cisco CallManager User Preferences and selects the services to be displayed on the IP phone.

5 User presses the Services button on the IP phone to see the selected IP phone services on the LCD display.

1 Web developer creates IP phone services application files using XML tags.

Architecture

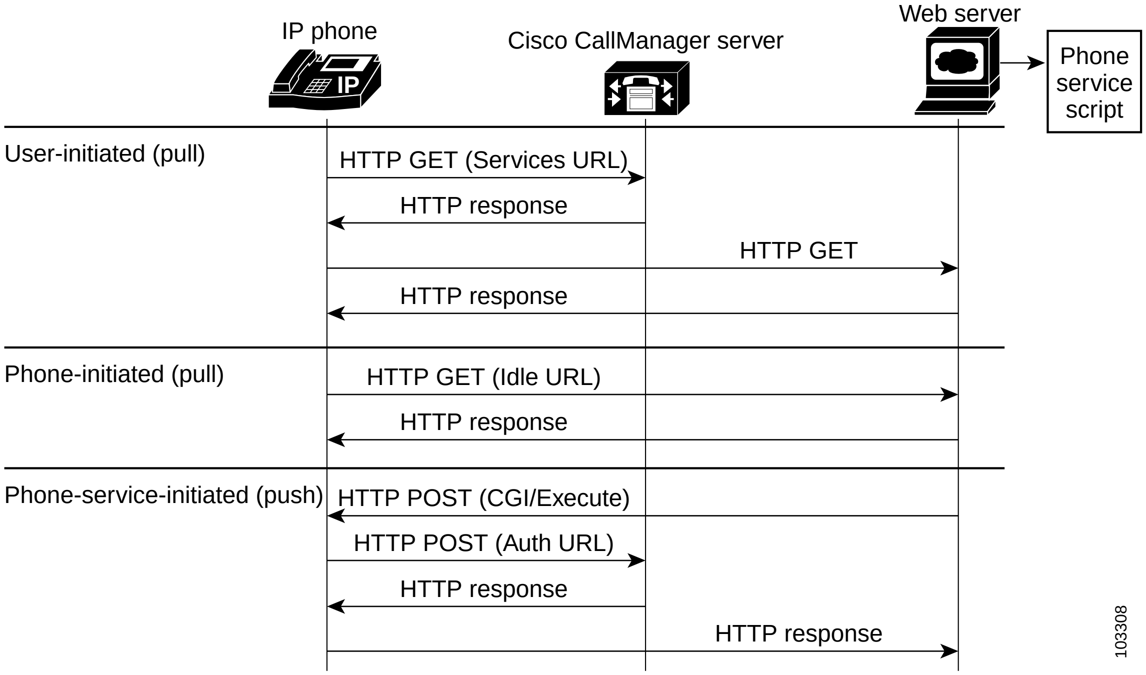
This section describes the following XML services concepts:

- XML Services Call Flow
- Architecture for Troubleshooting XML Services

XML Services Call Flow

HTTP call flows for XML services are shown in Figure 18 on page 29. Services can be initiated by the user, by the phone, or by another service that the phone is already using.

Figure 18 HTTP Web Services Request Call Flow



User-Triggered Service

When a user initiates a phone service, the following call flow is observed:

1. A user presses the Services button on an IP phone, and a menu of services is displayed.
2. The user selects a service and invokes it using a soft key on the phone.
3. The phone then sends an HTTP request to the URL on the web server that has been specified for the selected service.
4. The web server invokes the URL's ASP (Active Server Pages) or JSP (Java Server Pages) service.
5. The web server processes information and returns an XML response to the phone.
6. The phone uses an embedded web server to properly display the XML response to the user.

Event-Triggered Service

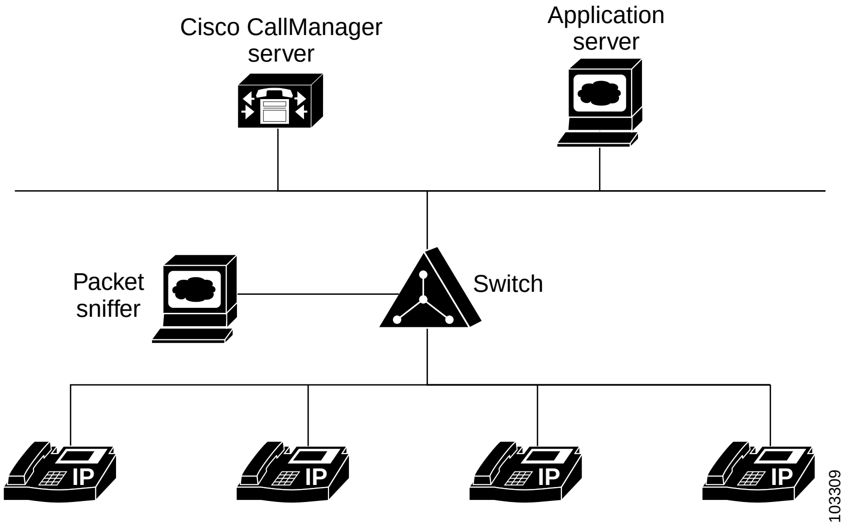
When an event initiates a phone service, the following call flow is observed:

1. An event is triggered through call processing or some other scenario that triggers a service.
2. Call information is transferred to the service from an application that is monitoring calls.
3. The service processes the information, formats the response, and locates the phone to which to push the data.
4. The XML data is pushed to the phone and presented to the user.

Architecture for Troubleshooting XML Services

Figure 19 shows the suggested architecture for troubleshooting XML services on Cisco CallManager networks, in which a packet sniffer is located at the switch. Typically, you collect sniffer traces by connecting a laptop or other sniffer-equipped device to a Catalyst port that is configured to span the VLAN or specific port(s), directly connected at the router, that contain the trouble information. If no free port is available, connect the sniffer-equipped device to a hub that is inserted between the switch and the device.

Figure 19 Architecture for Troubleshooting XML Services



Troubleshooting Tools

The following tools will help you troubleshoot problems with XML services.

- Standard web browser (Microsoft Internet Explorer 5.0 or a later version)

  Verifies authentication or connection issues to and from the XML services URL and the phone.
- Text editor such as Microsoft Notepad or an HTML editor

  Allows you to analyze the syntax of the XML code and make changes where necessary.
- Network packet analyzer such as Ethereal or SnifferPro

  Verifies exactly what data is being sent among the Cisco CallManager, the application server, and the phones.

- Validator

  Parses the XML content of a file or web page and authenticates it against a schema for Cisco IP phone XML objects. The XML Validator is a command-line tool that is available on a CD-ROM in the Cisco Press book called *Developing Cisco IP Phone Services*. Instructions for XML Validator can be found in Chapter 11 of that book. The following is an example of a validated file from Validator:

  ```
  <?xml version="1.0"?>
  <CiscoIPPhoneMenu>
      <Title>Cisco Phones Are Great, Right?</Title>
      <Prompt>What do you want to do today?</Prompt>
      <MenuItem>
          <Name>Yahoo</Name>
          <URL>http://www.yahoo.com/index.html</URL>
      </MenuItem>
  </CiscoIPPhoneMenu>
  ```

  The following is an example of a validation error reported by Validator:

  ```
  MS Parser reported this error:
  Code = 0xc00ce014
  Source = Line : b Char : 11
  Error Description = Element content is invalid according to the DTD/Schema.
  Expecting: Input Flags.

  Source text where the error occurred:

                <DefaultValue></DefaultValue>
  --------------^
  ```

Troubleshooting Checklist

The following tips apply to troubleshooting Cisco IP phone service applications:

- Microsoft Internet Explorer 5 or higher can display the XML source with its default style sheet.
- Standard IP troubleshooting techniques are important for diagnosing HTTP errors.
- Understanding the syntax and programming techniques for XML development can help you to diagnose problems.

Use the following checklist for initial troubleshooting of common problems.

____ Ensure that the user is associated with the device. The username and password that are passed to Cisco CallManager for authentication must have a corresponding user who is associated with the device that will receive the PUSH. See “PUSH-Related Issues” on page 32.

____ Check for parsing issues. See “Parsing Issues and Syntax Errors” on page 33.

____ If the AXL client application is unable to connect to the AXL service, check the following:

____ Ensure that the web service is reachable from the Cisco CallManager and the IP phone. From your web browser, type the URL of the web service page that has been configured in Cisco CallManager to confirm that the service is reachable. You can perform a quick check at the phone by pressing the Services button and selecting the service that has been configured.

____ Externally verify the name resolution for URLs and whether the phone has DNS set. If DNS is suspected as a source of trouble, use IP addresses in URLs.

____ Troubleshoot the PUSH-related issue. See “PUSH-Related Issues” on page 32.

If you still cannot locate the root cause for your problem, examine some packets using the Packet Sniffer. Typically, you collect sniffer traces by connecting a laptop or other sniffer-equipped device to a Catalyst port that is configured to span the VLAN or specific port(s), directly connected at the router, that contains the trouble information. If no free port is available, connect the sniffer-equipped device to a hub that is inserted between the switch and the device.

To use the Packet Sniffer, perform the following steps:

Step 1 Connect the Packet Sniffer to the switch.

Step 2 Set the Packet Sniffer to “span ports” and turn all filtering off.

Step 3 Turn on the Packet Sniffer.

Step 4 Try running your service again normally.

Step 5 Once the error condition has occurred, stop the Sniffer capture and save the file.

Step 6 Report the problem to Cisco using the process described in “Error Reporting” on page 33.

When reporting a problem, be sure to have available the IP and MAC addresses of all equipment that is involved, such as IP phones, gateways, and Cisco CallManager servers.

Service Connection

Copy the service URL of your service as configured in Cisco CallManager and paste it into the web browser on your PC. Then analyze the response to determine whether the service can be reached.

Use a logged Telnet session to verify that the desired HTTP headers are returned. Telnet to the server on port 80; then, enter get /*path*/*page*.

PUSH-Related Issues

A PULL results from a request made at the phone, whereas a PUSH results from the web service initiating content to be sent to the phone. When troubleshooting PUSH problems, use the following simple HTML page to verify that the username and password are correct:

```
<HTML>
  <HEAD>
    </HEAD>
      <BODY>
         <FORM action="http://10.0.0.1/CGI/Execute" Method="POST">
         <TEXTAREA NAME="XML" Rows="20" Cols="80">
              <CiscoIPPhoneExecute>
              <ExecuteItem Priority="0" URL="Play:chime.raw"/>
              </CiscoIPPhoneExecute>
         </TEXTAREA>
          <BR>
          <input type=submit value=POST>
       </FORM>
     </BODY>
</HTML>
```

This HTML page provides a text area for entering the XML object to be PUSHed to the phone. You need to replace 10.0.0.1 in the example above with the IP address of your phone. When the POST button is clicked, the object will be PUSHed to the phone. The phone will then force the browser to authenticate, and a username and password window will open. After authenticating, the phone will return the <CiscoIPPhoneResponse> object, which will be viewable in the web browser if you are using an XML-capable browser, such as Internet Explorer 5 or a later version. Using this simple HTML page allows you to quickly rule out any authentication or XML parsing and validation issues with your application.

Parsing Issues and Syntax Errors

The XML Validator can be used to validate the syntax of the XML code as a first step.

When troubleshooting XML parsing errors, do not exceed the maximum field lengths allowed by the XML schema. For example, the <Title> and <Prompt> fields are each limited to 32 characters (31 characters in some older firmware loads). A good reference for XML syntax and restrictions can be found in the Cisco Press book *Developing Cisco IP Phone Services*. For more detailed information, refer to the Cisco IP Phone XML Schema file (xsd) that is included in SDK version 3.3(4) or later. This XML-standard schema file defines all of the details of the allowed XML objects and can be used with XML editing software packages to greatly simplify the process of creating and validating Cisco IP phone XML objects.

The following tips apply to troubleshooting XML parsing errors in Cisco IP phone services applications:

- Verify the object tags (the object tags are case-sensitive).
- Verify that the ampersand (&) and the other four special characters in XML are used according to their restrictions while inside XML objects. By XML convention, the XML parser also requires that you use an escape sequence for five special characters, which are listed in Table 2.

Table 2 Escape Sequences for Special Characters

Name	Character	Escape Sequence
Ampersand	&	&amp;
Apostrophe	'	&apos;
Left angle bracket	<	&lt;
Right angle bracket	>	&gt;
Quote	"	&quot;

Error Reporting

When reporting problems with IP phone services (IPPS), be prepared to provide the following information:

- Sniffer traces connected at the switch and at the back of the phone.
- A detailed description of the network elements involved and their respective IP addresses in sniffer traces.
- A brief description of the function of the service and the problem found.
- The XML source code that shows the XML request that was sent to the phone and the XML response from the phone that was received by the application.
- The code used in the service application. This code is also helpful when you are trying to determine the root cause of problems.

Error and Status Codes

An error or status code is received by the XML application as an XML response from the web browser that is resident in the phone.

An XML error tag appears in the following format:

```
<CiscoIPPhoneError Number="x"/>
```

The following error codes can result from a PUSH.

- Error 1 = Error parsing CiscoIPPhoneExecute object
- Error 2 = Error framing CiscoIPPhoneResponse object
- Error 3 = Internal file error
- Error 4 = Authentication error

The following error messages may appear on the prompt line of the Cisco IP phone display.

- XML Error [4] = XML Parser Error (Invalid Object)
- HTTP Error [8] = Unknown HTTP Error
- HTTP Error [10] = HTTP Connection Failed