



MGCP Scripting Support for Cisco IOS VoiceXML

Cisco IOS VoiceXML supports Cisco's Media Gateway Control Protocol (MGCP) scripting feature to run VoiceXML documents. The MGCP scripting support for VoiceXML is based on existing support for Tcl. An MGCP call agent (CA) can signal the gateway to run a VoiceXML document on a channel. An argument string can be passed from the call agent to the running document, and the document can return an argument string to the call agent on completion.

For more information on MGCP, refer to the [Cisco IOS MGCP and Related Protocols Configuration Guide](#), Release 12.3.

The following sections describe specific details related to using MGCP with VoiceXML.

Syntax

- Because there is no VoiceXML symbol currently defined in the MGCP scripting specification, the MGCP signal uses Tcl in the syntax. The filename of the VoiceXML document must contain the vxml extension. The syntax for the MGCP signal is:

```
S:script/tcl(uri.vxml,arguments)
```

- The string returned in the <exit> tag of the document is returned in the report on completion to the call agent. The namelist is reported in the form <name>=<value>. The return status is reported as:

```
"return-status=<success or failure>".
```

The syntax is:

```
O:script/oc(return-status={success or failure}, <name>=<value>, ...)
```

- When the MGCP call agent sends a command to the Cisco gateway telling it to run a VoiceXML document, the response can be either success (OC) or failure (OF). The VoiceXML code includes a return status code in the OC response.
- The argument string from the call agent to the VoiceXML application can be up to 150 characters.

Call Legs

VoiceXML documents can only handle one call leg. If the call agent tries to run a VoiceXML document on a connection, or on a call leg connected to another call leg, the attempt fails.



Note

The Cisco MGCP feature creates a virtual, or place-holder, second call leg with the CRCX signal. The CRCX signal cannot be used by the call agent to run a VoiceXML document because the VoiceXML code fails with two call legs. The RQNT signal should be used instead.

Transfer

VoiceXML documents are not expected to run the `<transfer>` tag, because a call agent application normally initiates all call signaling. The VoiceXML document, however, is not prevented from running the transfer. If there are ports on the gateway that accept local signaling, the transfer can place a call.

Caching

VXML documents loaded from MGCP are cached; caching is not checked on every call.

The default cache check time is one minute; the check time can be configured with the **cache reload time** command in the application global configuration submenu. The cache check is on top of the HTTP protocol, which is fixed for fast rather than safe operation. Thus, the **http client cache refresh** command also affects MGCP document cache reloading if the load uses HTTP. TFTP uses the **cache reload time** command.

If the document is not used for 30 minutes, it is deleted from memory. This TTL is not configurable.

After the VoiceXML document has been loaded and run, use the **show call application voice** command (using the summary keyword) to display it. Normally, the syntax of the document name is *filename.vxml*, but if an existing application from a different URL uses that name, it will be given a suffix of *_1.vxml*.

You can preload the VoiceXML document into RAM using the application configuration commands. The name must be *filename.vxml*, and the URL must be identical to the URL in the `S:` command from the call agent for the MGCP application to use the loaded application. This permits the configuration of the language for the application.

Example of MGCP Scripting for VoiceXML

This section contains an example of a call agent running a simple VoiceXML document on a channel, including:

- Debug output on the gateway when the document runs
- Notification request from the call agent to the gateway
- Notification from the gateway back to the call agent
- A **show** command to look at the document

The following steps outline the process when the VoiceXML document is run:

1. The VoiceXML document is `debug.vxml`:

```
<vxml version="1.0" base="http://sunhost1/vxml/">
  <!-- Document to just output debug -->
  <!-- version 2 -->
  <var name="exitstring" expr="'ran this document'"/>
  <form id="dir_search" scope="document">
    <block>
      <cisco-puts>debug.vxml is running this document\n</cisco-puts>
      <cisco-puts>handoff_string=<cisco-putvar
        namelist="session.handoff_string"/>\n</cisco-puts>
      <exit namelist="exitstring"/>
    </block>
  </form>
</vxml>
```



Note

`<cisco-puts>` and `<cisco-putvar>` are Cisco private elements. For more information, see the *Cisco VoiceXML Programmer's Guide*.

2. The call agent sends this message to the gateway to run the document debug.vxml on channel 3 of the second T1:

```
RQNT 32052 ds1-2/3 MGCP 0.1
X:234
R:script/oc, script/of
S:script/tcl(http://sunhost1/sample/debug.vxml,arg1,arg2,arg3)
```

3. On the gateway, if you enter the **debug vxml puts** command, it displays:

```
Oct 18 09:49:47.061:debug.vxml is running this document
Oct 18 09:49:47.061:handoff_string=arg1,arg2,arg3
```

4. The MGCP notify message returned to the call agent is:

```
NTFY 12 ds1-2/3@sblab160.cisco.com MGCP 0.1
X:234
O:SCRIPT/oc(return-status=success, exitstring=ran this document)
```

5. Use the **show call application voice** command to view the entire document in memory:

```
Router# show call application voice debug.vxml

VoiceXML Application debug.vxml
  URL=http://sunhost1/sample/debug.vxml
  No languages configured
  It has 0 calls active.
  Interpreted by Voice Browser Version 1.0 for VoiceXML 1.0.
```

