Endpoint Configurations

Endpoint Configurations

Configuration guides for endpoints are in the child pages below. They define the various characteristics of the phone such as CUCM addresses to use, security parameters, video capabilities etc. If you are looking for a command reference on making the phone do something, please see the page **Commands**. If you want to see how the various features are all tied together with configuration and usage guidance, including sample scripts, please see the **Features** section.

Question and Answers

Q. Can I configure a 78xx phone in Camelot?

A. Yes. You can configure any phone type in Camelot by using the following commands:

sip.phone.modelnumber

sip.protocol.reguseragenthdr

sip.protocol.regsupportedhdr

See endpoint configurations for **sip.phone**|sip.phone]and **sip.protocol**|sip.protocol] for information on the above parameters. You can then define the various attributes and capabilities you want for the phone. A sample script is below:

US99980	Test: Zydeco device registration for sipx endpoints using US99810		Zydecco.tcl

With that said, ***I need to get on my soapbox***. By defining the phone type, you are not actually testing the phone. So if your test cases call for testing feature X on a 7861 and feature X on a 6941 to prove feature X works on both endpoint types, by using Camelot ***you are not*** accomplishing your test objective. If a test case with Camelot passes with one phone type defined this way, it will pass with all endpoints defined this way, all other configurations being equal. You can even do things not available in the real world like have your 7960 endpoint do presentation sharing!

Camelot is designed to test the systems, not the phones. Camelot is its own implementation of phone features that does not use actual phone endpoint code. It is used primarily for load, performance, and scale testing. It is also used for feature testing and unit testing of the systems where using an actual phone is inconvenient or not necessary. A **red flag** ***is if you know the registration sequence is different for a particular phone, or it handles features in a different way, or if the server behaves in a different way based on phone model***, you need to ***request a new feature*** to handle the variations. A good example of this are RoundTable endpoints (handle conferences and transfers differently, FCP etc) and Jabber endpoints (different registration method using DNS, UDS etc).