Configuring AAA for Cisco Voice Gateways Configuration Guide, Cisco IOS Release 15.2M&T
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 noninfringement 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.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at 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. (1005R)

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

Configuring AAA for Cisco Voice Gateways Configuration Guide, Cisco IOS Release 15.2M&T
© 2011 Cisco Systems, Inc. All rights reserved.
Configuring AAA for Cisco Voice Gateways Configuration Guide, Cisco IOS Release 15.2M&T
Configuring AAA for Cisco Voice Gateways

© 2009–2011 Cisco Systems, Inc. All rights reserved.
Overview of AAA on Voice Gateways

This chapter consists of the following:

- Authentication, Authorization, and Accounting, page 1
- AAA Requests to Multiple RADIUS Servers, page 8
- Customizing Accounting Records, page 10
- Related Features and Technologies, page 12
- Related Documents, page 13

Authentication, Authorization, and Accounting

For a gateway to provide authentication and accounting services, enable and configure it to support authentication, authorization, and accounting (AAA) services. AAA enables the gateway to interact with a RADIUS security server to authenticate users (typically incoming calls) and to perform accounting services. For more information about RADIUS and AAA security services, refer to the Cisco IOS Security Configuration Guide.

AAA Authentication

The gateway normally uses AAA with interactive voice response (IVR) to check the legitimacy of a prospective gateway user on the basis of an account number (collected by IVR) or Automatic Number Identification (ANI). When the gateway uses AAA with IVR, the IVR application collects the user account and personal identification number (PIN) information and then passes it to the AAA interface. The AAA interface makes a RADIUS authentication request using the given information and, based on the information received from the RADIUS server, forwards either a pass message or a fail message to the IVR application.

For more information about authentication services using AAA, refer to the “Configuring Authentication” chapter in the Cisco IOS Security Configuration Guide.
AAA Accounting

A call leg is a discrete segment of a call connection that lies between two points in the connection. Each call made through the gateway consists of two call legs: incoming and outgoing. The RADIUS server collects basic start-stop connection accounting data or syslog accounting information during the accounting process for each call leg created on the gateway.

To collect basic start-stop connection accounting data, the gateway must be configured to support gateway-specific H.323 accounting functionality. The gateway sends accounting data to the RADIUS server in one of four ways, as is shown in the following sections:

- Using RADIUS AV Pairs, page 2
- Appendix 1, “Using RADIUS AV Pairs” Overloading the Acct-Session-Id Field, page 5
- Using Vendor-Specific RADIUS Attributes, page 6
- Using Syslog Records, page 8

Using RADIUS AV Pairs

Basic start-stop connection accounting data and standard RADIUS attributes are used where possible using standard Internet Engineering Task Force (IETF) RADIUS attribute/value (AV) pairs. Table 1 shows the accounting-related IETF RADIUS attributes supported in Cisco IOS Release 12.2.

<table>
<thead>
<tr>
<th>Number</th>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>Called-Station-Id</td>
<td>Allows the network access server to send the called telephone number as part of the Access-Request packet (using Dialed Number Identification Service [DNIS] or similar technology). This attribute is only supported on ISDN and on modem calls on the Cisco ASR1000, Cisco AS5200, and Cisco AS5300 routers if used with ISDN PRI.</td>
</tr>
<tr>
<td>31</td>
<td>Calling-Station-Id</td>
<td>Allows the network access server to send the calling telephone number as part of the Access-Request packet (using ANI or similar technology). This attribute has the same value as the remote-addr attribute from TACACS+. This attribute is supported only on ISDN and on modem calls on the Cisco ASR1000, Cisco AS5200, and Cisco AS5300 routers if used with ISDN PRI.</td>
</tr>
<tr>
<td>40</td>
<td>Acct-Status-Type</td>
<td>(Accounting) Indicates whether this Accounting-Request marks the beginning of the user service (start) or the end (stop).</td>
</tr>
<tr>
<td>41</td>
<td>Acct-Delay-Time</td>
<td>(Accounting) Indicates how many seconds the client has been trying to send a particular record.</td>
</tr>
<tr>
<td>42</td>
<td>Acct-Input-Octets</td>
<td>(Accounting) Indicates how many octets have been received from the port over the course of this service being provided.</td>
</tr>
<tr>
<td>43</td>
<td>Acct-Output-Octets</td>
<td>(Accounting) Indicates how many octets have been sent to the port in the course of delivering this service.</td>
</tr>
</tbody>
</table>
### Table 1  Supported IETF RADIUS Accounting Attributes (continued)

<table>
<thead>
<tr>
<th>Number</th>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>44</td>
<td>Acct-Session-Id</td>
<td>(Accounting) A unique accounting identifier that makes it easy to match start and stop records in a log file. Acct-Session ID numbers restart at 1 each time the router is power cycled or the software is reloaded. To send this attribute in access-request packets, use the <code>radius-server attribute 44 include-in-access-req</code> command in global configuration mode.</td>
</tr>
<tr>
<td>45</td>
<td>Acct-Authentic</td>
<td>(Accounting) Indicates how the user was authenticated, whether by RADIUS, the network access server itself, or another remote authentication protocol. This attribute is set to “radius” for users authenticated by RADIUS; “remote” for TACACS+ and Kerberos; or “local” for local, enable, line, and if-needed methods. For all other methods, the attribute is omitted.</td>
</tr>
<tr>
<td>46</td>
<td>Acct-Session-Time</td>
<td>(Accounting) Indicates how long (in seconds) the user has received service.</td>
</tr>
<tr>
<td>47</td>
<td>Acct-Input-Packets</td>
<td>(Accounting) Indicates how many packets have been received from the port over the course of this service being provided to a framed user.</td>
</tr>
<tr>
<td>48</td>
<td>Acct-Output-Packets</td>
<td>(Accounting) Indicates how many packets have been sent to the port in the course of delivering this service to a framed user.</td>
</tr>
</tbody>
</table>
Table 1  **Supported IETF RADIUS Accounting Attributes (continued)**

<table>
<thead>
<tr>
<th>Number</th>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>49</td>
<td>Acct-Terminate-Cause</td>
<td>(Accounting) Reports details on why the connection was terminated. Termination causes are indicated by a numeric value as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1. User request</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. Lost carrier</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Lost service</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Idle timeout</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. Session timeout</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. Admin reset</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7. Admin reboot</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8. Port error</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9. NAS error</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10. NAS request</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11. NAS reboot</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12. Port unneeded</td>
</tr>
<tr>
<td></td>
<td></td>
<td>13. Port pre-empted</td>
</tr>
<tr>
<td></td>
<td></td>
<td>14. Port suspended</td>
</tr>
<tr>
<td></td>
<td></td>
<td>15. Service unavailable</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16. Callback</td>
</tr>
<tr>
<td></td>
<td></td>
<td>17. User error</td>
</tr>
<tr>
<td></td>
<td></td>
<td>18. Host request</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note</strong> For attribute 49, Cisco IOS supports values 1 to 6, 9, 12, and 15 to 18.</td>
</tr>
<tr>
<td>50</td>
<td>Acct-Multi-Session-Id</td>
<td>(Accounting) A unique accounting identifier used to link multiple related sessions in a log file.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Each linked session in a multilink session has a unique Acct-Session-Id value, but shares the same Acct-Multi-Session-Id.</td>
</tr>
<tr>
<td>51</td>
<td>Acct-Link-Count</td>
<td>(Accounting) Indicates the number of links known in a given multilink session at the time an accounting record is generated. The network access server can include this attribute in any accounting request that might have multiple links.</td>
</tr>
<tr>
<td>52</td>
<td>Acct-Input-Gigawords</td>
<td>Indicates how many times the Acct-Input-Octets counter has wrapped around $2^{32}$ over the course of the provided service.</td>
</tr>
</tbody>
</table>
For complete information about RADIUS and the use of IETF-defined attributes, refer to the Cisco IOS Security Configuration Guide.

**Overloading the Acct-Session-Id Field**

Attributes that cannot be mapped to standard RADIUS attributes are packed into the Acct-Session-Id attribute field as ASCII strings separated by the “/” character. The Acct-Session-Id attribute contains the RADIUS account session ID, a unique identifier that links accounting records associated with the same login session for a user. To support additional fields, the following string format has been defined for this field:

```
<session id>/<call leg setup time>/<gateway id>/<connection id>/<call origin>/<call type>/<connect time>/<disconnect time>/<disconnect cause>/<remote ip address>
```

Table 2 shows the field attributes to be used with the Overloaded Acct-Session-Id method and provides a brief description of each.
Because of the limited size of the Acct-Session-Id string, it is not possible to embed many information elements in it. Therefore, this feature supports only a limited set of accounting information elements.

Use the `gw-accounting h323` command to configure the overloaded session ID method of applying H.323 gateway-specific accounting.

### Using Vendor-Specific RADIUS Attributes

The IETF draft standard specifies a method for communicating vendor-specific information between the network access server (NAS) and the RADIUS server by using the vendor-specific attribute (Attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes that are

<table>
<thead>
<tr>
<th>Field Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SESSION-ID</td>
<td>Specifies the standard RADIUS account session ID.</td>
</tr>
</tbody>
</table>
| SETUP-TIME          | Provides the Q.931 setup time for this connection in Network Time Protocol (NTP) format. NTP time formats are displayed as \%H:\%M:\%S.\%k \%Z \%tw \%tn \%td \%Y where:
  * \%H is hour (00 to 23).
  * \%M is minutes (00 to 59).
  * \%S is seconds (00 to 59).
  * \%k is milliseconds (000 to 999).
  * \%Z is time zone string.
  * \%tw is day of week (Saturday through Sunday).
  * \%tn is month name (January through December).
  * \%td is day of month (01 to 31).
  * \%Y is year including century (for example, 1998). |
| GATEWAY-ID          | Indicates the name of the underlying gateway in the form of “gateway.domain_name.” |
| CALL-ORIGIN         | Indicates the origin of the call relative to the gateway. Possible values are originate and answer. |
| CALL-TYPE           | Indicates call leg type. Possible values are telephony and VoIP.             |
| CONNECTION-ID       | Specifies the unique global identifier used to correlate call legs that belong to the same end-to-end call. The field consists of 4 long words (128 bits). Each long word is displayed as a hexadecimal value and is separated by a space character. |
| CONNECT-TIME        | Provides the Q.931 connect time for this call leg, in NTP format.            |
| DISCONNECT-TIME     | Provides the Q.931 disconnect time for this call leg, in NTP format.         |
| DISCONNECT-CAUSE    | Specifies the reason a call was taken offline as defined in the Q.931 specification. |
| REMOTE-IP-ADDRESS   | Indicates the address of the remote gateway port where the call is connected. |
not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option using the format recommended in the specification. The Cisco vendor-ID is 9, and the supported option has vendor-type 1, which is named “cisco-avpair.” The value is a string of the format:

```
protocol: attribute sep value *
```

“Protocol” is a value of the Cisco “protocol” attribute for a particular type of authorization. “Attribute” and “value” are an appropriate AV pair defined in the Cisco TACACS+ specification, and “sep” is “=” for mandatory attributes and “*” for optional attributes. The full set of features available for TACACS+ authorization can also be used for RADIUS.

For complete and current information on voice-related vendor-specific RADIUS attributes, refer to the RADIUS Vendor-Specific Attributes Voice Implementation Guide at the following URL:

Some of the VSA fields and their ASCII values are listed in Table 3. You can review the rest at the location noted above.

Table 3 VSA Fields and Their ASCII Values

<table>
<thead>
<tr>
<th>IETF RADIUS Attribute</th>
<th>Vendor-Specific Company Code</th>
<th>Subtype Number</th>
<th>Attribute Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>26 9 23</td>
<td>h323-remote-address</td>
<td>Indicates the IP address of the remote gateway.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 9 24</td>
<td>h323-conf-id</td>
<td>Identifies the conference ID.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 9 25</td>
<td>h323-setup-time</td>
<td>Indicates the setup time for this connection in Coordinated Universal Time (UTC), formerly known as Greenwich Mean Time (GMT) and Zulu time.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 9 26</td>
<td>h323-call-origin</td>
<td>Indicates the origin of the call relative to the gateway. Possible values are originating and terminating, which are equivalent to originate and answer in the Call-Origin field.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 9 27</td>
<td>h323-call-type</td>
<td>Indicates call leg type. Possible values are telephony and VoIP.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 9 28</td>
<td>h323-connect-time</td>
<td>Indicates the connection time for this call leg in UTC.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 9 29</td>
<td>h323-disconnect-time</td>
<td>Indicates the time this call leg was disconnected in UTC.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 9 30</td>
<td>h323-disconnect-cause</td>
<td>Specifies the reason a connection was taken offline per the Q.931 specification.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 9 31</td>
<td>h323-voice-quality</td>
<td>Specifies the impairment/calculated planning impairment factor (ICPIF) affecting voice quality for a call.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>26 9 33</td>
<td>h323-gw-id</td>
<td>Indicates the name of the underlying gateway.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Use the `gw-accounting h323 vsa` command to configure the VSA method of applying H.323 gateway-specific accounting.

### Using Syslog Records

The syslog accounting option exports the information elements associated with each call leg through a system log message, which can be captured by a syslog daemon on the network. The syslog output consists of the following:

```
<server timestamp> <gateway id> <message number> : <message label> : <list of AV pairs>
```

The syslog message fields are listed in Table 4.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>server timestamp</td>
<td>The time stamp created by the server when it receives the message to log.</td>
</tr>
<tr>
<td>gateway id</td>
<td>The name of the gateway that emits the message.</td>
</tr>
<tr>
<td>message number</td>
<td>The number assigned to the message by the gateway.</td>
</tr>
<tr>
<td>message label</td>
<td>A string that identifies the message category.</td>
</tr>
<tr>
<td>list of AV pairs</td>
<td>A string consisting of <code>&lt;attribute name&gt; &lt;attribute value&gt; pairs</code> separated by commas.</td>
</tr>
</tbody>
</table>

Use the `gw-accounting h323 syslog` command to configure the syslog record method of gathering H.323 accounting data.

### AAA Requests to Multiple RADIUS Servers

AAA requests can be made to different RADIUS servers based on account number, called party number, and incoming trunk groups.

Consider the topology shown in Figure 1 below.
In Figure 1, the gateway identifies the TCL IVR application to be invoked based on:

- The access number dialed by the caller, or
- The account number or card number

Calling party A has a different access number compared to calling parties B and C. Because calling parties B and C use the same access number, the service provider can use the TCL IVR script to manipulate the call and direct the AAA information to the appropriate billing/RADIUS server based on the individual card numbers entered by callers B and C. If a caller is using an account number instead of a prepaid calling card, the service provider can use the caller’s account number to direct the AAA information to the appropriate RADIUS/billing server that is used to authorize calls based on account numbers.

Figure 2 below shows RADIUS/Billing server selections based on T1/E1 trunk groups. For example, if caller A is using a pre-paid application, the service provider directs AAA information to the appropriate RADIUS/Billing server based on the T1/E1 trunk group that is assigned to receive prepaid application calls.
Customizing Accounting Records

You can create an accounting template to customize your accounting records based on your billing needs. An accounting template is a text-based interface that allows you to customize and define the content of that template and helps reduce billing traffic from the gateway to the accounting servers.

A sample accounting template applicable to POTS and VoIP dial-peers is shown below.

Vendor specific attributes (VSAs) used in session applications such as h323-ivr-out, h323-credit-amount, h323-credit-time, h323-billing-model, are only controlled in the TCL script and not in the accounting template. If you specify these VSAs in the accounting template, they are ignored and no error messages are reported. You cannot control h323-conf-id and h323-incoming-conf-id; they are mandatory VSAs required for co-relating accounting messages on the incoming and outgoing legs.

Session applications also use some VSAs for authentication and authorization which are not controlled by the accounting template. For example, h323-ivr-out, h323-credit-amount, h323-credit-time, and h323-billing-model are only controlled by the TCL script. The VSAs listed in this template are voice-specific only. Non-voice specific attributes cannot be controlled through this template. To add new attributes not defined in this template, contact your Cisco marketing representative.

To delete an attribute, add the # sign in front of the attribute name.
## Accounting Template

You can create a custom accounting template by selecting only those VSAs that are applicable to your billing needs. The list below shows some VSAs that can be used to create custom accounting templates.

Each accounting template attribute is unique. For example, the attribute `disconnect-time` is applied to a stop message because you can only get that information at the end of a call and not at the start of that call.

If you want to generate individual accounting templates for different incoming trunk calls on an accounting server, you can define multiple templates and associate them with different sets of incoming dial-peers. You can customize the template by deleting attributes that are not required for your specific template.

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Usage and Restrictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>h323-gw-id</td>
<td></td>
</tr>
<tr>
<td>h323-call-origin</td>
<td></td>
</tr>
<tr>
<td>h323-call-type</td>
<td></td>
</tr>
<tr>
<td>h323-setup-time</td>
<td></td>
</tr>
<tr>
<td>h323-connect-time</td>
<td></td>
</tr>
<tr>
<td>h323-disconnect-time</td>
<td></td>
</tr>
<tr>
<td>h323-disconnect-cause</td>
<td></td>
</tr>
<tr>
<td>h323-remote-address</td>
<td></td>
</tr>
<tr>
<td>h323-voice-quality-subscriber</td>
<td>ICPIF</td>
</tr>
<tr>
<td><strong>Detail CallHistory</strong></td>
<td></td>
</tr>
<tr>
<td>acom-level</td>
<td>#POTS leg only</td>
</tr>
<tr>
<td>noise-level</td>
<td>#POTS leg only</td>
</tr>
<tr>
<td>img-pages-count</td>
<td>#POTS leg only</td>
</tr>
<tr>
<td>voice-tx-duration</td>
<td>#POTS leg only</td>
</tr>
<tr>
<td>tx-duration</td>
<td>#POTS leg only</td>
</tr>
<tr>
<td>charged-units</td>
<td>#</td>
</tr>
<tr>
<td>disconnect-text</td>
<td>#</td>
</tr>
</tbody>
</table>

For the latest list of VSAs, refer to the *RADIUS Vendor-Specific Attributes Voice Implementation Guide*. 
Related Features and Technologies

- Service Provider feature set for VoIP uses the IVR for interaction with the caller; collects digits for accounting and billing purposes.
- Authentication, Authorization, and Accounting (AAA) feature is used in conjunction with IVR.
- Settlement for Packet Telephony on Cisco Access Platforms uses the TCL IVR scripts for the billing process.
- Debit Card for Packet Telephony on Cisco Access Platforms uses TCL IVR extensively for interoperability.
• **Enhanced Multi-Language Support for Cisco IOS Interactive Voice Response** allows you to implement and add support for new languages and text-to-speech (TTS) notations to the core IVR infrastructure on Cisco voice gateways.

**Related Documents**

For related information on the features described in this document, refer to the following documents:

- Internetworking Terms and Acronyms
- Hardware and software guides for Cisco 5000 series universal voice gateways for information on installing the hardware and performing basic configuration
- **Platform Specific Information for Cisco 2600 Series Routers**
- **Platform Specific Information for Cisco 3600 Series Routers**
- Voice over IP for the Cisco AS5300, the section “VFC Management” provides VCWare download instructions
- **Enhanced Multi-Language Support for Cisco IOS Interactive Voice Response**, Cisco IOS Release 12.2(2)T feature module, describes multi-language support for dynamic prompts
- **Configuring Debit Card Applications**, for information on debit card applications that work in conjunction with Cisco interactive voice response (IVR) software, AAA, RADIUS, and an integrated third party billing system
- **Cisco IOS Voice, Video, and Fax Configuration Library**, Cisco IOS Release 12.3
- **Cisco IOS Voice, Video, and Fax Command Reference**, Cisco IOS Release 12.3, for command reference information on Cisco IOS commands
- “Authentication, Authorization, and Accounting (AAA)” chapter in the **Cisco IOS Security Configuration Guide**, Cisco IOS Release 12.2, for information on configuring accounting records
- **RADIUS Vendor-Specific Attributes Voice Implementation Guide**
- **TCL IVR API Version 2.0 Programmer’s Guide** for information about creating and implementing Tool Command Language (TCL) IVR scripts
- **Cisco Prepaid Debitcard Multi-language Programmer’s Reference**
- **Enhanced Multi-Language Support for Cisco IOS Interactive Voice Response**
Cisco IOS Configuration

This chapter includes the following sections:

- Prerequisite Configuration, page 1
- Directing AAA Requests to a RADIUS Server, page 3
- Enabling and Disabling Accounting for any Call Leg, page 6
- Customizing Accounting Packets, page 7

Prerequisite Configuration

The following general tasks are prerequisites to configuring the Cisco IOS features described in this document:

- Establish a working IP network. For more information about configuring IP, refer to the Cisco IOS IP Configuration Guide.
- Configure Voice over IP. For more information about configuring Voice over IP, refer to the Cisco IOS Voice Configuration Library.
- Program and configure the interface between the RADIUS server and the Cisco voice gateway to operate with vendor specific attributes (VSAs). Refer to the RADIUS Vendor-Specific Attributes Voice Implementation Guide.
- Download the TCL scripts that are not embedded in Cisco IOS from the Cisco CCO software support URL:
- Define and apply IVR applications on the dial peer to direct AAA requests to a RADIUS server. For more information, see the Cisco IOS TCL and VoiceXML Application Guide

Configuring AAA Basics

You must follow these steps to set up AAA before you start directing AAA requests to a RADIUS server:

**Step 1**
Enable authentication, authorization, and accounting (AAA) security services:

```
Router(config)# aaa new-model
```
Example:

```
aaa new-model
```

**Step 2** Define a RADIUS server host by entering the following command:

```
Router(config)# radius server host ipaddress auth-port port-number acct-port port-number
```

Example:

```
radius server host 1.5.35.10 auth-port 2001 acct-port 2002
```

**Step 3** Use the RADIUS server defined in Step 2 to define a AAA group.

a. To define a group name, enter the following command in global configuration mode:

```
Router(config)# aaa group server radius group-name
```

**Note** For the argument `group-name` in the command, enter the name of the specific RADIUS server (for example server1) you want to authenticate, or enter the argument `radius` if you want to authenticate all RADIUS servers.

Example:

```
aaa group server radius server1
```

b. To configure the IP address of the RADIUS server for the group server, enter the following command in group server configuration mode:

```
Router(config-sg-radius)# server ip-address auth-port port-number acct-port port-number
```

Example:

```
server 1.5.35.10 auth-port 2001 acct-port 2002
```

**Step 4** Exit group server configuration mode.

```
Router(config-sg-radius)# exit
```

**Step 5** To specify the password for use between the gateway and the RADIUS server, enter the following command in global configuration mode:

```
Router(config)# radius-server key key
```

Example:

```
radius-server key luser23
```

**Step 6** Use the AAA group defined in Step 2 above to define an AAA method list.

a. For voice authentication, enter the `aaa authentication login` command

a.

```
Router(config)# aaa authentication login list-name method1 [method2...]
```

Examples:

```
aaa authentication login h323 group server2
aaa authentication login MIS-access group radius
```

b. For voice authorization, enter the `aaa authorization` command.
Cisco IOS Configuration

Directing AAA Requests to a RADIUS Server

You can use TCL scripts or the CLI to direct AAA requests to a specific RADIUS server based on:

- Customer account number
- Called party number
- Trunk group

Directing AAA Requests by Using Account Numbers

It is easier to use TCL scripts instead of the CLI to direct AAA requests using account numbers. To use TCL scripts for directing AAA requests using account numbers, follow the steps below:

**Step 1**
Before you start using TCL scripts to direct AAA requests using account numbers, you must define and apply the interactive voice response (IVR) application on the dial peer.

**Step 2**
Use the authentication, authorization, and accounting TCL verbs to customize your TCL scripts. Refer to the Accounting Template, page 21 in Chapter 1, “Overview of AAA on Voice Gateways” for an example of a TCL script.

The authentication, accounting, and authorization TCL verbs are:

- **a.** Authentication: Use the following TCL verb:
  
  ```
  aaa authenticate account password [-a avlistSend][-s servertag]
  ```

- **b.** Authorization: Use the following TCL verb:
  
  ```
  aaa authorize account password ani destination {legID|info-tag} [-s servertag]
  ```

- **c.** Accounting: Use the following TCL verbs to start or update accounting messages:

  ```
  aaa accounting start {legID|info-tag} [-s servertag]
  ```

**Step 3**
(Optional). If you use the accounting TCL verb, then use the **accounting suppress** command to suppress accounting on the same dial peer on which you have specified your application.

Follow the steps below to suppress accounting on the dial peer:

- **a.** Enter the **voice class aaa** command in global configuration mode.

  ```
  Router(config)# voice-class aaa tag
  ```

  Example:
voice-class aaa 1001

b. Enter the **accounting suppress** command in voice class configuration mode.
   
   ```
   Router(config-class)# accounting suppress
   ```

c. Enter the **voice class aaa** command in dial peer configuration mode.
   
   ```
   Router(config)# dial-peer voice tag {pots|voip} 
   Router(config-dial-peer)# voice class aaa tag 
   ```

   Example:
   ```
   dial-peer voice 101 voip 
   voice class aaa 1001 
   ```

### Directing AAA Requests using Called Party Number

You can use the called party number to direct AAA requests in dial peer configuration mode as follows:

---

**Step 1** Define a dial peer.

a. Enter dial peer configuration mode using the **dial peer voice** command. The argument **number** defines a particular dial peer.

   ```
   Router(config)# dial-peer voice tag {pots|voip} 
   ```

   Example:
   ```
   dial-peer voice 202 pots 
   ```

d. Specify the incoming called number using the **incoming called number** command in dial peer configuration mode. The argument **string** is a series of digits that specifies the incoming called number.

   ```
   Router(config-dial-peer)# incoming called number string 
   ```

   Example:
   ```
   incoming called number 5550900 
   ```

**Step 2** Define the voice class.

a. Enter the **voice class aaa** command in global configuration mode. The argument **tag** identifies the dial peer.

   ```
   Router(config)# voice class aaa tag 
   ```

   Example:
   ```
   voice-class aaa 202 
   ```

d. Define authentication, authorization, and accounting methods. Enter the authentication, authorization and accounting commands in voice class mode. The argument **methodListName** is used to name the list of authentication, authorization or accounting methods applicable to each command.

   ```
   Router(config-class)# authentication method methodListName 
   Router(config-class)# accounting method methodListName 
   Router(config-class)# authorization method methodListName 
   ```

   Example:
authentication method pw
accounting method rd
authorization method pc

c. Define voice class in dial peer configuration mode. Enter dial peer configuration mode and then define the voice class in that mode. The argument tag identifies the same dial peer as in step a) above.

Router(config)# dial-peer voice tag (pots|voip)
Router(config-dial-peer)# voice-class aaa tag

Example:
dial-peer voice 202 pots
voice-class aaa 202

### Directing AAA Requests Using Trunk Groups

To direct AAA requests using trunk groups, a trunk group must first associate with a dial peer. To use this method, group all the interfaces using one trunk group and define only one dial peer instead of individual ports for the interfaces using that trunk group.

You can direct AAA requests using trunk groups in dial-peer configuration mode as follows:

---

**Step 1** Define the trunk group by entering the `trunk group` command in global configuration mode. The argument `tag` is a number.

Router(config)# trunk group tag

Example:

```
trunk group 303
```

**Step 2** Use the trunk group tag in Step 1 to group the interfaces.

a. Enter the `interface serial` command in global configuration mode to specify a serial interface on the channelized T1 or E1 controller. The argument `slot/port` denotes the slot and port number where the channelized T1 or E1 controller is located. The argument `timeslot` denotes the ISDN D channel timeslot which is 15 for channelized E1 and 23 for channelized T1.

Router(config)# interface serial slot/port: timeslot

Example:

```
interface serial 1/1:23
```

b. Enter the `trunk group` command.

Router(config-inter-serial)# trunk group tag

Example:

```
trunk group 303
```

**Step 3** Use the `tag` defined in Step 2b) above.

a. Enter the `voice class aaa` command in global configuration mode.

Router(config)# voice-class aaa tag

Example:
voice-class aaa 303

b. Define authentication, accounting, and authorization methods. Enter the **authentication method**, **accounting method**, and **authorization method** commands in voice class mode. The argument *methodListName* is used to name the list of authentication, accounting, or authorization methods applicable to each command.

Router(config-class)# authentication method methodListName
Router(config-class)# accounting method methodListName
Router(config-class)# authorization method methodListName

Example:

authentication method ab
accounting method cd
authorization method ef

c. Enter dial peer configuration mode using the **dial peer voice** command.

Router(config)# dial-peer voice {pots|voip}

Example:

dial-peer voice 303 pots

d. Define the voice class in dial peer configuration mode. The argument *tag* identifies the same dial peer as in Step a above.

Router(config-dial-peer)# voice-class aaa tag

Example:

voice-class aaa 303

e. Define the trunk group in dial peer configuration mode. The argument *tag* is the same number as in Step b) above.

Router(config-dial-peer)# trunk group tag

Example:

trunk group 303

---

**Enabling and Disabling Accounting for any Call Leg**

Enabling voice accounting by using the **gw-accounting aaa** command will send only the default list of VSAs to the accounting server.

**Global Configuration Mode**

To enable and disable accounting for any call leg in global configuration mode, follow these steps:

---

**Step 1**

To enable accounting for any call leg, enter the **gw-accounting aaa** command in global configuration mode. Use the no form of the command to disable accounting.

Router (config)# gw-accounting aaa
Router (config)# no gw-accounting aaa
To disable accounting based on the type of dial peer, use the following command:

**Step 2**

To disable accounting based on the type of dial peer, use the following commands:

a. Enter the `gw-accounting aaa` command.
   
   ```
   Router(config) # gw-accounting aaa
   ```

b. Enter the `suppress` command.
   
   ```
   Router(config-gw-accounting-aaa) # suppress
   ```

   You have a choice of entering `pots` or `voip`, based on the type of dial peer.

c. Enter the `suppress pots` or `suppress voip` command.
   
   ```
   Router(config-gw-accounting-aaa) # suppress pots
   or
   Router(config-gw-accounting-aaa) # suppress voip
   ```

dial-peer configuration mode

To disable accounting in dial-peer configuration mode, follow these steps:

**Step 1**

Enter the `voice class aaa` command in global configuration mode.

```
Router(config) # voice class aaa tag
```

Example:

```
voice-class aaa 303
```

**Step 2**

Enter the `accounting suppress` command in voice class aaa mode.

```
Router(config-class)# accounting suppress [in-bound|out-bound]
```

Example:

```
accounting suppress
```

**Step 3**

Enter the `voice class aaa` command in dial peer configuration mode.

```
Router(config)# dial-peer voice tag {pots|voip}
Router(config-dial-peer)# voice-class aaa tag
```

Example:

```
dial-peer voice 303 pots
voice-class aaa 303
```

Customizing Accounting Packets

This section contains the following sub-sections:

- Configuration Overview, page 8
Configuration Overview

Accounting packets for voice calls consist of voice-specific attributes as well as those that are not specific to voice. This document focuses only on voice-specific attributes. You can add some application-level attributes through the TCL script and fine tune the attribute list created by the system; the result is an accounting template that is customized to your accounting needs.

To customize your accounting packets, first create accounting templates.

Note

If you do not want to customize your accounting packets, enable voice accounting by using the `gw-accounting aaa` command to generate accounting packets. A specific set of attributes, which include both non voice-specific and voice-specific attributes, is automatically sent by the gateway to the RADIUS server.

To view the current list of VSAs, refer to the *RADIUS Vendor Specific Attributes Voice Implementation Guide*. For example, in the “Accounting Template” section on page 21 of Chapter 1, “Overview of AAA on Voice Gateways”, the default attributes are:

```
 h323-gw-id
 h323-call-origin
 h323-call-type
 h323-setup-time
 h323-connect-time
 h323-disconnect-time
 h323-disconnect-cause
 h323-remote-address
 h323-voice-quality ICPIF
 subscriber
```

To send all the VSAs to the accounting server use the `template callhistory-detail` command in global configuration mode. The Accounting Template, page 21 in Chapter 1, “Overview of AAA on Voice Gateways” includes the default and new VSAs. Refer to the “Using Callhistory-detail to Send All VSAs” section on page 9 for configuration details.

For the latest list of VSAs, refer to *RADIUS Vendor-Specific Attributes Voice Implementation Guide*. 
To fine tune your accounting packets based on your billing needs, create accounting templates using specific VSAs that are applicable to your accounting needs. For example, to target different accounting servers for incoming calls from different trunks, you must define multiple accounting templates and associate them with different sets of incoming dial peers. To create a template, remove the attributes that are not applicable by adding the # sign in front of each of those attributes.

To tune your accounting packets, remove attributes that do not apply to your billing needs. Deleting these attributes creates a custom accounting template that acts as a filter, allowing only the defined attributes to be sent to the accounting server. To apply a customized template, first define the template using the `call accounting template voice` command in global configuration mode, and then apply it using either TCL scripts or the CLI. If you are using the CLI, you can apply the template either in global configuration or dial-peer configuration mode. Refer to the “Defining and Applying Customized Accounting Templates” section on page 10 for configuration details.

Specific VSAs that cannot be controlled by the accounting template are sent as attribute-value (AV) pairs through the `avlistSend` argument of the TCL verbs used in the script, and they are:

- h323-ivr-out
- h323-ivr-in
- h323-credit-amount
- h323-return-code
- h323-prompt-id
- h323-time-and-delay
- h323-redirect-number
- h323-preferred-lang
- h323-redirect-ip-addr
- h323-billing-model
- h323-currency

## Configuration Tasks for Customizing Accounting Packets

Use the “Configuration Overview” section on page 8 to plan your customizing needs before you begin the applicable configuration tasks below.

### Generate Accounting Packets by Enabling Voice Accounting

To automatically generate accounting packets by enabling voice accounting, enter the `gw-accounting aaa` command in global configuration mode.

```
Router(config)# gw-accounting aaa
```

```
Router(gw-accounting aaa)# exit
```

### Using Callhistory-detail to Send All VSAs

To send all VSAs (default and new) to the accounting server:

#### Step 1

Enter the `gw-accounting aaa` command to enter C mode.

```
Router(config)# gw-accounting aaa
```
Step 2  Enter the **acct-template callhistory-detail** command in V mode.

```
Router(config-gw-accounting-aaa)# acct-template callhistory-detail
Router(config-gw-accounting-aaa)#
```

## Defining and Applying Customized Accounting Templates

To define an accounting template:

**Step 1** Enter the **call accounting-template voice** command in global configuration mode. Enter the template name for *acctTempName*. The *url* is the address where you store the template. Always assign a `.cdr` extension to the filename in the URL.

```
Router(config)# call accounting-template voice acctTempName url
```

Example:

```
call accounting-template voice cdr1 tftp://highway/mjs/templates/cdr1.cdr
```

**Note**

After bootup, if the template file fails to load from the TFTP server, the system tries to automatically reload the file at five minute intervals.

You can use an accounting template through the CLI (in global configuration or dial-peer configuration mode), or by using TCL verbs.

To use an accounting template through the CLI in global configuration mode, use the following commands:

**Step 1** Enter the **gw-accounting aaa** command to enter gateway accounting AAA mode.

```
Router(config)# gw-accounting aaa
```

**Step 2** Enter the **acct-template** command. Assign your template name to *acctTempName*.

```
Router(config-gw-accounting-aaa)# acct-template acctTempName
```

Example:

```
acct-template april1
```

## Applying a Customized Accounting Template through the CLI in Dial-Peer Configuration Mode

To apply a customized accounting template through the CLI in dial peer configuration mode, follow these steps:

**Step 1** Enter the **call accounting-template voice** command in global configuration mode. Assign your template name to *acctTempName* and your template address (usually your tftp address) to *url*.

```
Router(config)# call accounting-template voice acctTempName url
```

Example:

```
call accounting-template voice cdr1 tftp://highway/mjs/templates/cdr1.cdr
```

**Step 2** Enter the **voice class aaa** command in global configuration mode. Assign a numerical value to *tag*. 
Router(config)# voice class aaa tag

Example:
voice-class aaa 404

**Step 3** Enter the **accounting-template** command in voice class AAA mode. Assign your template name to `acctTempName`.

Router(config-class)# accounting-template acctTempName

Example:
accounting-template april1

**Step 4** Change configuration mode from global to dial peer and using the **dial peer voice** command; enter the **voice class aaa** command in dial-peer configuration mode. The numerical value of `tag` is the same value of `tag` in Step 2 above.

Router(config)# dial peer voice number [pots|voip] 
Router(config-dial-peer)# voice class aaa tag

Example:
dial-peer voice 404 pots
voice-class aaa 404

---

**Applying a Customized Accounting Template through a TCL Script**

Use the **aaa accounting start** TCL verb. Assign an incoming or outgoing call leg, or assign an information tag. Assign your template name to `acctTempName`.

`aaa accounting start {legID|info-tag} -t acctTempName`

**Adding Attributes to Accounting Packets through TCL scripts**

To add attributes to accounting packets through TCL scripts, follow these steps:

---

**Step 1** Use the `avlistSend` argument in the TCL verbs to send the following attributes:

- h323-ivr-out
- h323-ivr-in
- h323-credit-amount
- h323-return-code
- h323-prompt-id
- h323-time-and-delay
- h323-redirect-number
- h323-preferred-lang
- h323-redirect-ip-addr
- h323-billing-model
- h323-currency
Step 2  Use TCL verbs for authentication, authorization, and accounting.

a. For authentication, use the **aaa authenticate** TCL verb.  
   ```tcl
   aaa authenticate account password [-a avlistSend]
   ```

b. For authorization, use the **aaa authorize** TCL verb.  
   ```tcl
   aaa authorize account password ani destination {legID | info-tag} [-a avlistSend]
   ```

c. For accounting, use the **aaa accounting start** TCL verb.  
   ```tcl
   aaa accounting start {legID | info-tag} [-a avlistSend]
   ```

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco’s trademarks can be found at [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. (1005R)

© 2007—2011 Cisco Systems, Inc. All rights reserved.
Examples

This chapter includes Cisco IOS configuration examples and application scenarios of sending customized accounting templates to billing servers and directing AAA requests based on account number, called number, and trunk grouping.

This chapter includes the following examples:

- Directing a AAA Request Based on Trunk Grouping, page 1
- Directing a AAA Request Using an Account Number, page 2
- Directing AAA Traffic Using Called Number, page 9
- Directing Accounting Templates Using Called Number, page 12

Directing a AAA Request Based on Trunk Grouping

A Cisco IOS configuration example for directing a AAA request based on trunk grouping is shown below:

```
aaa new-model
aaa group server radius sg1
server 10.1.0.1 auth-port 1645 acct-port 1646
!
aaa group server radius sg2
server 10.20.0.1 auth-port 1645 acct-port 1646
!
aaa group server radius sg-def
server 10.0.50.50 auth-port 1645 acct-port 1646
!
aaa authentication login abc group sg1
aaa authentication login def group sg2
aaa authentication login h323 group sg-def
aaa authorization exec abc group sg1
aaa authorization exec def group sg2
aaa authorization exec h323 group sg-def
aaa accounting connection abc start-stop group sg1
aaa accounting connection def start-stop group sg2
aaa accounting connection h323 start-stop group sg-def
!
gw-accounting aaa
!```
Examples

Directing a AAA Request Using an Account Number

radius-server host 10.1.0.1 auth-port 1645 acct-port 1646 key abc
radius-server host 10.20.0.1 auth-port 1645 acct-port 1646 key ghi
radius-server host 10.0.50.50 auth-port 1645 acct-port 1646 key ghi
!
trunk group 101
!
interface Serial1/0/0:23
no ip address
no ip directed-broadcast
isdn switch-type primary-ni
isdn protocol-emulate network
isdn incoming-voice modem
no cdp enable
trunk-group 101
!
interface Serial1/0/1:23
no ip address
no ip directed-broadcast
isdn switch-type primary-ni
isdn protocol-emulate network
isdn incoming-voice modem
no cdp enable
trunk-group 101
!
voice-class aaa 1
authentication abc
authorization abc
accounting abc
!
voice-class aaa 1
authentication def
authorization def
accounting def
!
dial-peer voice 700 pots
destination-pattern 1700.......  
voice-class aaa 1
trunkgroup 101
!
dial-peer voice 900 pots
destination-pattern 1900.......  
voice-class aaa 2
trunkgroup 202
!

Directing a AAA Request Using an Account Number

In Figure 1, the caller places a call to MGCP 1 with a modified debit card application applied to an incoming POTS dial peer. After the caller enters the card number (for example, 5555550001), AAA requests are directed to RADIUS 4.
A Cisco IOS configuration example for the topology in Figure 1 is shown below:

Radius protocol debugging is on
Radius packet hex dump debugging is off
Radius packet protocol debugging is on

gcp1#
02:27:24: %ISDN-6-CONNECT: Interface Serial3:22 is now connected to 4111234567
02:27:30: %ISDN-6-CONNECT: Interface Serial3:22 is now connected to 4111234567
02:27:44: RADIUS/ENCODER: Unsupported AAA attribute timezone
02:27:44: RADIUS(00000024): Encoding nas-port...Only port-type avlbl
02:27:44: RADIUS(00000024): sending
02:27:44: RADIUS(00000024): Encoding nas-port...Only port-type avlbl
02:27:44: RADIUS/ENCODER(00000024): acct_session_id: 36
02:27:44: RADIUS(00000024): sending
02:27:44: RADIUS(00000024): Sending to unknown id 10 10.5.20.60:1234, Accounting-Request, len 480 <---
Accounting request sent
02:27:44: RADIUS/ENCODER(00000024): acct_session_id: 36
02:27:44: RADIUS(00000024): sending
02:27:44: RADIUS(00000024): Encoding nas-port...Only port-type avlbl
02:27:44: RADIUS(00000024): acct_session_id: 36
02:27:44: RADIUS(00000024): sending
02:27:44: RADIUS: Acct-Status-Type [40] 6 Start [1]
02:27:44: RADIUS: Acct-Session-Id [44] 10 "00000024"
02:27:44: RADIUS: Vendor, Cisco [26] 25
02:27:44: RADIUS: Vendor, Cisco [26] 56
02:27:44: RADIUS: Conf-Id [24] 50 "h323-conf-id=2E176F2 8E6611D4 804F9FAD CD27A5B5"
02:27:44: RADIUS: Vendor, Cisco [26] 65
02:27:44: RADIUS: Cisco AVPair [1] 59 "h323-incoming-conf-id=2E176F2 8E6611D4 804F9FAD CD27A5B5"
02:27:44: RADIUS: Vendor, Cisco [26] 38
02:27:44: RADIUS: Vendor, Cisco [26] 33
02:27:44: RADIUS: Vendor, Cisco [26] 41

Figure 1 AAS Redirect Using An Account Number: Scenario 1
Examples

Directing a AAA Request Using an Account Number

02:27:44: RADIUS: Vendor, Cisco [26] 38
02:27:44: RADIUS: Vendor, Cisco [26] 33
02:27:44: RADIUS: Vendor, Cisco [26] 41
02:27:44: RADIUS: NAS-Port-Type [61] 6 Async [0]
02:27:44: RADIUS: Vendor, Cisco [26] 19
02:27:44: RADIUS: Calling-Station-Id [31] 12 "4111234567"
02:27:44: RADIUS: Called-Station-Id [30] 7 "13120"
02:27:44: RADIUS: Delay-Time [41] 6 0
02:27:44: RADIUS: Send to unknown id 20 10.6.20.60:1704, Access-Request, len 170 <--
Authentication request sent
02:27:44: RADIUS: authenticator 12 F5 47 5D 2D 07 74 A4 - 97 02 33 36 14 34 DE 07 to
server specified in
02:27:44: RADIUS: User-Name [1] 8 "555555" application and defined in
02:27:44: RADIUS: Vendor, Cisco [26] 56
CD27A5B5"
02:27:44: RADIUS: Vendor, Cisco [26] 37 but was included here to exercise
authentication tcl verb.
02:27:44: RADIUS: NAS-Port-Type [61] 6 Async [0] Also
authentication
02:27:44: RADIUS: Vendor, Cisco [26] 19 done using account number
02:27:44: RADIUS: Send to unknown id 21 10.6.20.60:1704, Access-Request, len 200 <--
Authorization request
02:27:44: RADIUS: authenticator CA 67 12 31 EE 78 19 F4 - 4E 4E 21 1B FB B2 06
02:27:44: RADIUS: User-Name [1] 8 "0555550"
02:27:44: RADIUS: Vendor, Cisco [26] 56
CD27A5B5"
02:27:44: RADIUS: Vendor, Cisco [26] 37
02:27:44: RADIUS: Calling-Station-Id [31] 12 "4111234567"
02:27:44: RADIUS: NAS-Port-Type [61] 6 Async [0]
02:27:44: RADIUS: Vendor, Cisco [26] 19
02:27:44: RADIUS: Calling-Station-Id [31] 12 "4111234567"
02:27:44: RADIUS: Received from id 10 10.5.20.60:1234, Accounting-response, len 20 <--
AccountingResponse from 02:27:44: RADIUS: authenticator E8 35 04 32 38 3E 78 78 - F1 D4
38 10 41 F3 EE F3 Radius observed
02:27:44: RADIUS: Received from id 20 10.6.20.60:1704, Access-Accept, len 200 <-- Response from Radius
02:27:44: RADIUS: authenticator 46 30 66 55 18 DC 82 A0 - 05 14 ED D8 3B 17 0C E7
02:27:44: RADIUS: Vendor, Cisco [26] 27
02:27:44: RADIUS: Vendor, Cisco [26] 34
02:27:44: RADIUS: Vendor, Cisco [26] 26
02:27:44: RADIUS: Cisco AVpair [1] 20 "h323-return-code=0"
02:27:44: RADIUS: Vendor, Cisco [26] 30
02:27:44: RADIUS: h323-credit-time [102] 24 "h323-credit-time=54123"
02:27:44: RADIUS: Vendor, Cisco [26] 33
Examples

Directing a AAA Request Using an Account Number

02:27:44: RADIUS: Vendor, Cisco       [26]  24
02:27:44: RADIUS:   h323-currency      [110]  18  "h323-currency=US"
02:27:44: RADIUS: Received from id 24
02:27:44: RADIUS: Received from id 21 10.6.20.60:1704, Access-Accept, len 200
02:27:44: RADIUS: authenticator 0F 13 36 EA B1 FC B3 95 - 85 FC CC FE 1F 2F 95 D0
02:27:44: RADIUS: Cisco AVpair       [1]   20  "h323-return-code=0"
02:27:44: RADIUS: Cisco AVpair       [1]   20  "h323-return-code=0"

In Figure 2, the caller places a call to MGCP 1 with a modified debit card application applied to an incoming POTS dial peer. After the caller enters the card number (for example, 7777770001), the AAA requests are directed to RADIUS server 1.

**Figure 2** AAA Redirect Using An Account Number: Scenario 2

Shown below are some Cisco IOS configuration examples for the topology in Figure 2:

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname mgcp1
!
```

![Cisco IOS configuration examples](image.png)
aaa new-model
!

aaa group server radius sg1
    server 10.6.20.60 auth-port 1698 acct-port 1699
!

aaa group server radius sg4
    server 10.6.20.60 auth-port 1704 acct-port 1705
!

aaa group server radius sg6
    server 10.6.20.60 auth-port 1708 acct-port 1709
!

aaa authentication login sanjo_aaa1 group sg1
aaa authentication login sanjo_aaa4 group sg4
aaa authentication login sanjo_aaa6 group sg6

aaa authorization exec sanjo_aaa1 group sg1
aaa authorization exec sanjo_aaa4 group sg4
aaa authorization exec sanjo_aaa6 group sg6

aaa accounting connection sanjo_aaa1 start-stop group sg1
aaa accounting connection sanjo_aaa4 start-stop group sg4
aaa accounting connection sanjo_aaa6 start-stop group sg6

aaa session-id common
!
resource-pool disable
!
ip subnet-zero
ip host milp 10.223.254.255
!
isdn switch-type primary-5ess
!
fax interface-type modem
mta receive maximum-recipients 0
!
controller T1 3
framing esf
clock source line secondary
linecode b8zs
pri-group timeslots 1-24
!
gw-accounting aaa
!
interface Ethernet0
    ip address 10.0.254.255
    no ip redirects
    no ip mroute-cache
    no cdp enable
!
interface Serial0:23
    no ip address
    isdn switch-type primary-5ess
    isdn T321 0
    isdn T310 4000
    no cdp enable
!
interface Serial3:23
    no ip address
    trunk-group 1
    isdn switch-type primary-5ess
    isdn incoming-voice modem
    isdn T321 0
    no cdp enable
!
interface FastEthernet0
    ip address 10.1.2.3 254.254.255.0
duplex auto
speed auto
!
ip classless
ip route 254.255.1.0  223.254.10.0.1.5.0.1 <- Route to tftp server
no ip http server              10.223.254.255
radius-server host 10.6.20.60 auth-port 1704 acct-port 1705
radius-server host 10.6.20.60 auth-port 1698 acct-port 1699
radius-server retransmit 3
radius-server key cisco
radius-server vsa send accounting
radius-server vsa send authentication
call rsvp-sync
call accounting-template voice cdr1 tftp://10.223.254.255/johndoe/sanjose/cdr/cdr1.cdr
!
call application voice integ_debit
tftp://10.223.254.255/johndoe/sanjose/apps/debitcard_int_redirect.tcl
call application voice integ_debit uid-len 6
call application voice integ_debit language 1 en
call application voice integ_debit language 2 sp
call application voice integ_debit set-location en 0
call application voice integ_debit set-location sp 0 tftp://10.6.20.50/prompts/
!
call application voice plain_debit
tftp://10.223.254.255/johndoe/sanjose/apps/debitcard.tcl
call application voice plain_debit uid-len 6
call application voice plain_debit language 1 en
call application voice plain_debit language 2 sp
call application voice plain_debit set-location en 0
call application voice plain_debit set-location sp 0 tftp://10.6.20.50/prompts/

voice-port 3:D
!
voice-port 0:D
!
dial-peer cor custom
dial-peer voice 100 pots
  application integ_debit <--------!Modified debit app to play
  incoming called-number 13120 new dynamic prompts, do
dial-peer voice 101 voip authentication,
destination-pabcern 510...... accounting redirect
  session target ipv4:10.0.0.1
  !
  !
dial-peer voice 102 voip
  destination-pabcern 408....... and use accounting
  session target ipv4:10.0.0.1
  !
  !
line con 0 exec-timeout 0 0
line aux 0
line vty 0 4
!
end
Debit Card Application for Directing AAA Requests Using an Account Number

A debit card application for directing AAA requests using an account number is shown below:

```tcl
proc act_GotCardNumber { } {
    global count
    global retryCnt
    global promptFlag
    global account
    global pin
    global accountLen
    global cardLen
    global pinLen
    global ani
    global method
    global acct-template
    set status [infotag get evt_status]
    # Do we need to calculate len ???
    if {$status == "cd_005"} {
        set number [infotag get evt_dcdigits]
        set numberLen [string length $number]
        if { $numberLen == $cardLen } {
            set account [string range $number 0 [expr $accountLen -1]]
            set pin [string range $number $accountLen [expr $cardLen -1]]
            ______________________________________________________
            # Select Servertag based on first 2 digits of account number.  #
            ______________________________________________________
            set spcode [string range $account 0 1]
            if { $spcode == 77 } {
                set method "sanjo_aaa1"
                set avList(h323-ivr-out,1) "account range: 77"
                set avList(h323-ivr-out,2) "color:violet"
                set avList(h323-ivr-out,3) "MethodName:sanjo_aaa1"
                puts "note that accounting template cdr1 needs to be defined in IOS CLI"
                set acct-template "cdr1"
            } else {
                set method "sanjo_aaa4"
                set avList(h323-ivr-out,1) "account range: 55"
                set avList(h323-ivr-out,2) "color:red"
                set avList(h323-ivr-out,3) "MethodName:sanjo_aaa4"
                set acct-template "cdr2"
            }
            puts "$spcode"
            set method "h323"
            set acct-template "none"
        } else {
            set method "h323"
            set acct-template "none"
            puts "THIS IS THE METHOD LIST USED: $method \r"
            puts "THIS IS THE TEMPLATE USED   : $template \r"
        }
        #Do accounting
        if { $method != "h323" } {
            aaa accounting start leg_incoming -a avList -s $method -t $template
            set avList(h323-ivr-out,2) "color:purple"
            aaa accounting update leg_incoming -a avList -s $method -t $template
            # suppress service provide module automatically generating
        }
    }
}
```
Directing AAA Traffic Using Called Number

In Figure 3, the caller places a phone call to MGCP 1. A modified debit card application is applied to an incoming POTS dial peer. The caller enters the card number and the destination number sequence which is completed at MGCP 1. The destination number (for example 408-428-xxxx) is now dialed and directed to MGCP 2. Accounting information based on the called number is directed from MGCP 2 to RADIUS server 5.
An example of a AAA redirect using the called number is shown below:

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname mgcp2
!
aaa new-model
!
!!!
!!!
aaa group server radius sg2
   server 10.6.20.60 auth-port 1700 acct-port 1701
!
!!!
aaa group server radius sg5
   server 10.6.20.60 auth-port 1706 acct-port 1707
!
!!!
aaa group server radius sg6
   server 10.6.20.60 auth-port 1708 acct-port 1709
!
!!!
!!!
aaa accounting connection sanjo_aaa5 start-stop group sg5
aaa accounting connection sanjo_aaa2 start-stop group sg2
!!!
!!!
username lab password 0 lab
!
!!!
!!!
resource-pool disable
!
!!
ip subnet-zero
!
!!
isdn switch-type primary-5ess
!!!
voice class aaa 2
   accounting method sanjo_aaa2
!
voice class aaa 5
```
accounting method sanjo_aaa5
!

fax interface-type modem
mta receive maximum-recipients 0
!
controller T1 0
   framing sf
   clock source line primary
   linecode ami
!
controller T1 1
   framing sf
   clock source line secondary 1
   linecode ami
!
controller T1 2
   framing sf
   linecode ami
!
controller T1 3
   framing esf
   linecode b8zs
   pri-group timeslots 1-24
!
gew-accounting aaa
!
!
interface Ethernet0
   ip address 10.6.20.501 254.255.1.0
   no cdp enable
interface Serial3:23
   no ip address
   isdn switch-type primary-5ess
   isdn incoming-voice modem
   no cdp enable
!
interface FastEthernet0
   ip address 10.0.0.1 255.255.255.0
   duplex auto
   speed auto
!
   ip classless
   ip route 254.255.1.0 223.254.10.0.1.5.0.1
   no ip http server
!
!
radius-server host 10.6.20.60 auth-port 1700 acct-port 1701 key cisco
radius-server host 10.6.20.60 auth-port 1706 acct-port 1707 key cisco
radius-server retransmit 3
radius-server key cisco
radius-server vsa send accounting
radius-server vsa send authentication

voice-port 3:D
!
!
mgcp profile default
!
On the terminating side, the incoming dial peers are VoIP. In this configuration, if the incoming called number begins with 510, VoIP dial peer 101 is applied. If the incoming called number begins with 408, VoIP dial peer 102 is applied.

```
dial-peer voice 100 pots
  destination-pabcern 510 .......
  port 3:D
  prefix 91510 ........
!
dial-peer voice 101 voip
  incoming called-number 510 .......
  voice-class aaa 5
!
dial-peer voice 200 pots
  destination-pabcern 408......
  port 3:D
  prefix 9
!
dial-peer voice 102 voip
  incoming called-number 408....... 
  voice-class aaa 2
!
dial-peer voice 300 pots
  application integ_debit
  incoming called-number .
  port 3:D
!
  line con 0
  exec-timeout 0 0
  line aux 0
  line vty 0 4
!
end
```

**Directing Accounting Templates Using Called Number**

In Figure 4 below, accounting template cdr1 allows h323-gw-id av pair but does not allow h323-call-origin. Accounting template cdr2 allows h323-call-origin but does not allow h323-gw-id av pair. The caller places a phone call to MGCP 1. A modified debit card application is applied to an incoming POTS dial peer.

Accounting template cdr1 is selected using an account number (for example 7777770001). To verify that cdr1 was applied correctly, examine the accounting record on RADIUS server 1.

Accounting template cdr2 is selected using an account number (for example 5555550001). To verify that cdr2 was applied correctly, examine the accounting record on RADIUS server 2.
Shown below is an example of directing an accounting template using an incoming called number:

```
version 12.2
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname mgcp1
!
aaa new-model
!

aaa group server radius sg1
   server 10.6.20.60 auth-port 1698 acct-port 1699
!
aaa group server radius sg4
   server 10.6.20.60 auth-port 1704 acct-port 1705
!
aaa group server radius sg6
   server 10.6.20.60 auth-port 1708 acct-port 1709
!

aaa authentication login sanjo_aaa1 group sg1
aaa authentication login sanjo_aaa4 group sg4
aaa authentication login sanjo_aaa6 group sg6
aaa authorization exec sanjo_aaa1 group sg1
aaa authorization exec sanjo_aaa4 group sg4
aaa authorization exec sanjo_aaa6 group sg6
aaa accounting connection sanjo_aaa1 start-stop group sg1
aaa accounting connection sanjo_aaa4 start-stop group sg4
aaa accounting connection sanjo_aaa6 start-stop group sg6
aaa session-id common
!
!
resource-pool disable
!
ip subnet-zero
ip host milp 10.223.254.255
trunk group 1
carrier-id test
hunt-scheme least-used both
```
isdn switch-type primary-5ess

voice class aaa 1

voice class aaa 6
  authentication method sanjo_aaa6
  authorization method sanjo_aaa6
  accounting method sanjo_aaa6

fax interface-type modem
mta receive maximum-recipients 0

controller T1 0
  framing sf
  linecode ami

controller T1 1
  framing sf
  clock source line secondary 1
  linecode ami

controller T1 2
  framing sf
  linecode ami

controller T1 3
  framing esf
  clock source line primary
  linecode b8zs
  pri-group timeslots 1-24

gw-accounting aaa

interface Ethernet0
  ip address 10.0.254.255
  no ip redirects
  no ip mroute-cache
  no cdp enable

interface Serial3:23
  no ip address
  trunk-group 1
  isdn switch-type primary-5ess
  isdn incoming-voice modem
  isdn T321 0
  no cdp enable

interface FastEthernet0
  ip address 10.1.2.3 254.255.0
  duplex auto
  speed auto

ip classless
ip route 254.255.1.0 223.254.10.1.5.0.1
no ip http server

radius-server host 10.6.20.60 auth-port 1708 acct-port 1709
radius-server host 10.6.20.60 auth-port 1704 acct-port 1705
radius-server host 10.6.20.60 auth-port 1698 acct-port 1699
radius-server host 10.6.43.255 auth-port 1645 acct-port 1646
radius-server host 10.6.37.10 auth-port 1645 acct-port 1646
radius-server retransmit 3
radius-server key cisco
radius-server vsa send accounting
radius-server vsa send authentication
call rsvp-sync
call accounting-template voice cdr1 tftp://10.223.254.255/johndoe/sanjose/cdr/cdr1.cdr
! call application voice acct_redirect_debit
tftp://10.6.20.50/sanjose/apps/debitcard_acct_redirect.tcl
call application voice acct_redirect_debit uid-len 6
call application voice acct_redirect_debit language 1 en
call application voice acct_redirect_debit language 2 sp
call application voice acct_redirect_debit set-location en 0 tftp://10.6.20.50/prompts/
call application voice acct_redirect_debit set-location sp 0 tftp://10.6.20.50/prompts/
!
call application voice integ_debit
tftp://10.223.254.255/johndoe/sanjose/apps/debitcard_int_redirect.tcl
call application voice integ_debit uid-len 6
call application voice integ_debit language 1 en
call application voice integ_debit language 2 sp
call application voice integ_debit set-location en 0
call application voice integ_debit set-location sp 0 tftp://10.6.20.50/prompts/
!
call application voice plain_debit
tftp://10.223.254.255/johndoe/sanjose/apps/debitcard.tcl
call application voice plain_debit uid-len 6
call application voice plain_debit language 1 en
call application voice plain_debit language 2 sp
call application voice plain_debit set-location en 0
call application voice plain_debit set-location sp 0 tftp://10.6.20.50/prompts/
!
voice-port 3:D
!
!
mgcp profile default
!
dial-peer cor custom
!
dial-peer voice 100 pots
debitcard_acct_redirect.tcl
incoming called-number .
!
dial-peer voice 101 voip
destination-pattern 1111234567
session target ipv4:10.0.0.1
!
dial-peer voice 102 voip
destination-pattern 408........
session target ipv4:10.0.0.1
!
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
!
end
Cisco IOS Troubleshooting

Troubleshooting

This chapter contains the following sections:

- Using Debug Commands, page 1
- Using Show Commands, page 8

Using Debug Commands

debug radius

The output below is from troubleshooting AAA redirect using called number for an incoming POTS dial peer.

! In this example, an incoming call is set up using dial-peer voice 1000 pots. Applying voice-class aaa 1 to dial-peer voice 1000 redirects AAA requests to the server specified for method list sanj_aaa1:10.6.20.70 auth-port 1698 acct-port 1699.

aaa group server radius sg1
  server 10.6.20.70 auth-port 1698 acct-port 1699
!
aaa group server radius sg6
  server 10.6.20.70 auth-port 1704 acct-port 1705
!
aaa group server radius sg7
  server 10.6.20.70 auth-port 1720 acct-port 1721
!
aaa authentication login sanj_aaa1 group sg1
aaa authorization exec sanj_aaa1 group sg1
aaa accounting connection sanj_aaa1 start-stop group sg1
!
aaa authentication login sanj_aaa6 group sg6
aaa authorization exec sanj_aaa6 group sg6
aaa accounting connection sanj_aaa6 start-stop group sg6
!
aaa authentication login sanj_aaa7 group sg7
aaa authorization exec sanj_aaa7 group sg7
aaa accounting connection sanj_aaa7 start-stop group sg7
voice class aaa 1
  authentication method sanj_aaa1
  authorization method sanj_aaa1
  accounting method sanj_aaa1
  accounting template cdr1
!
voice class aaa 2
  authentication method sanj_aaa6
  authorization method sanj_aaa6
  accounting method sanj_aaa6
  accounting template cdr2
!
voice class aaa 3
  authentication method sanj_aaa7
  authorization method sanj_aaa7
  accounting method sanj_aaa7
!
dial-peer voice 1000 pots
  application plain_debit
  incoming called-number 12345
  voice-class aaa 1
  port 0:D
!
dial-peer voice 1001 pots
  application plain_debit
  incoming called-number 12346
  voice-class aaa 2
  port 1:D

debbug radius
Radius protocol debugging is on
Radius packet hex dump debugging is off
Radius packet protocol debugging is on
debbug isdn q931
ISDN Q931 packets debugging is on

00:17:55: ISDN Se0:23: RX <- SETUP pd = 8 callref = 0x009D
00:17:55: Bearer Capability i = 0x8090A2
00:17:55: Channel ID i = 0xE1808397
00:17:55: Calling Party Number i = 0x0080, '4081234567', Plan:Unknown, Type:Unknown
00:17:55: Called Party Number i = 0xE9, '12345', Plan:Private, Type:Abbreviated
00:17:55: RADIUS/ENCODE(0000000C): Unsupported AAA attribute timezone
00:17:55: RADIUS(0000000C): Encoding nas-port...Only port-type avlbl
00:17:55: RADIUS(0000000C): sending
00:17:55: RADIUS: Send to unknown id 4 10.6.20.70:1699, Accounting-Request, len 262
00:17:55: RADIUS: authenticator 10 41 58 99 4C F2 B1 CD - 44 3E E3 60 5D 10 C3 A9
00:17:55: RADIUS: Acct-Session-Id [44] 10 "0000000C"
00:17:55: RADIUS: Vendor, Cisco [26] 56
00:17:55: RADIUS: Conf-Id [24] 50 "h323-conf-id=B8FE8B7F BF1711D3 800CE483 89ADC43B"
00:17:55: RADIUS: Vendor, Cisco [26] 31
00:17:55: RADIUS: h323-call-origin [26] 25 "h323-call-origin=answer"
00:17:55: RADIUS: Vendor, Cisco [26] 65
00:17:55: RADIUS: Cisco AVpair [1] 59 "h323-incoming-conf-id=B8FE8B7F BF1711D3 800CE483 89ADC43B"
00:17:55: RADIUS: User-Name [1] 12 "4081234567"
00:17:55: RADIUS: Acct-Status-Type [40] 6 Start [1]
00:17:55: RADIUS: NAS-Port-Type [61] 6 Async [0]
00:17:55: RADIUS: Vendor, Cisco [26] 19
00:17:55: RADIUS: Calling-Station-Id [31] 12 "4081234567"
00:17:55: RADIUS: Called-Station-Id [30] 7 '12345''
00:17:55: RADIUS: Service-Type [6] 6 Login
00:17:55: RADIUS: Delay-Time [41] 6 0
00:17:55: ISDN Se0:23: TX -> CALL_PROC pd = 8 callref = 0x809D
00:17:55: Channel ID i = 0xA98397
00:17:55: ISDN Se0:23: TX -> CONNECT pd = 8 callref = 0x809D
00:17:55: Received from id 4 10.6.20.70:1699, Accounting-response, len 20
00:17:55: authenticator DC CD BA E8 7E 02 EA D1 - 12 67 DC 57 3C 73 56 75
00:17:55: ISDN Se0:23: RX <- CONNECT_ACK pd = 8 callref = 0x009D
00:17:55: RADIUS: Received from id 3 10.6.20.70:1698, Access-Accept, len 200
00:18:06: RADIUS/ENCODE(0000000C): Unsupported AAA attribute timezone
00:18:06: RADIUS(0000000C): Encoding nas-port...Only port-type avlbl
00:18:06: RADIUS(0000000C): sending
00:18:06: RADIUS: Send to unknown id 5 10.6.20.70:1699, Accounting-Request, len 327
00:18:06: RADIUS: authenticator 2D 65 1C 3B 6D 5B 33 3D - C8 57 D6 02 B4 4F E4 4E
00:18:06: RADIUS: Acct-Session-Id [44] 10 '0000000C'
00:18:06: RADIUS: Conf-Id [24] 50 'h323-conf-id=B8FE8B7F BF1711D3 800CE483 89ADC43B'
00:18:06: RADIUS: Vendor, Cisco [26] 31
An incoming call is set up using dial-peer voice 1001 pots. dial-peer voice 1001 has voice-class aaa 2 applied which should redirect AAA requests to the server specified for method list sanj_aaa6: 10.6.20.70 auth-port 1708 acct-port 1709.
00:30:06: RADIUS: Received from id 6 10.6.20.70:1709, Accounting-response, len 20
00:30:06: RADIUS: authenticator E1 AD 70 9F DC 09 29 32 - 74 47 96 9F 3F 77 27 82
00:30:11: %ISDN-6-CONNECT: Interface Serial1:22 is now connected to 4081234567
00:30:19: RADIUS(0000000E): Encoding nas-port...Only port-type available
00:30:19: RADIUS(0000000E): sending
00:30:19: RADIUS: Send to unknown id 4 10.6.20.70:1708, Access-Request, len 199
00:30:19: RADIUS: authenticator CE 16 21 8D A5 59 56 9F - B7 E9 CA 5C EC CS 89 A0
00:30:19: RADIUS(0000000E): acct_session_id: 14
00:30:19: RADIUS: sending
00:30:19: RADIUS: Send to unknown id 7 10.6.20.70:1709, Accounting-Request, len 327
00:30:20: RADIUS: Received from id 4 10.6.20.70:1708, Access-Accept, len 173
00:30:20: RADIUS: authenticator FF 0D 72 0D 80 12 26 - 44 13 D5 0E C4 BB 71 BE
00:30:20: RADIUS: Vendor, Cisco [26] 34
00:30:20: RADIUS: Cisco AVpair [1] 28 'h323-credit-amount=7777.77'
00:30:20: RADIUS: Vendor, Cisco [26] 65
00:30:20: RADIUS: h323-billing-model [109] 27 'h323-billing-model=prepay'
00:30:20: RADIUS: Vendor, Cisco [26] 30
00:30:20: RADIUS: h323-mid [102] 24 'h323-mid=54329'
00:30:20: RADIUS: Vendor, Cisco [26] 33
00:30:20: RADIUS: h323-credit-time [110] 18 'h323-credit-time=US'
00:30:20: RADIUS: idle-timeout [28] 6 30
00:30:20: RADIUS: Received from id E
00:30:43: ISDN Sel:1:23: RX <- DISCONNECT pd = 8 callref = 0x0004
00:30:43: Cause i = 0x8290 - Normal call clearing
00:30:43: %ISDN-6-DISCONNECT: Interface Serial1:22 disconnected from 4081234567, call lasted 37 seconds
00:30:43: ISDN Sel:1:23: TX -> RELEASE pd = 8 callref = 0x8004
00:30:43: ISDN Sel:1:23: TX -> RELEASE_COMP pd = 8 callref = 0x0004
00:30:43: RADIUS/ENCODE(0000000B): Unsupported AAA attribute timezone
00:30:43: RADIUS(0000000B): Encoding nas-port...Only port-type available
00:30:43: RADIUS: Send to unknown id 7 10.6.20.70:1709, Accounting-Request, len 327
00:30:43: RADIUS: authenticator 99 5A B4 45 67 C0 F4 91 - 9B 4B C3 1D 7E DE 7D D1
00:30:43: RADIUS: Vendor, Cisco [26] 31
00:30:43: RADIUS: Cisco AVpair [1] 29 'h323-ivr-out=transactionID:4'
00:30:43: RADIUS: Vendor, Cisco [26] 66
00:30:43: RADIUS: Cisco AVpair [1] 29 'h323-ivr-out=Target:Unknown'
00:30:43: RADIUS: User-Name [1] 12 '4081234567'
Cisco IOS Troubleshooting

Troubleshooting

00:30:43: RADIUS: Acct-Status-Type [40]  6  Stop  [2]
00:30:43: RADIUS: NAS-Port-Type [61]  6  Async  [0]
00:30:43: RADIUS: Vendor, Cisco  [26]  19
00:30:43: RADIUS: Calling-Station-Id [31]  12  "4081234567"
00:30:43: RADIUS: Called-Station-Id [30]  7  "12346"
00:30:43: RADIUS: NAS-IP-Address [4]  6  10.5.20.100
00:30:43: RADIUS: Delay-Time [41]  6  0
00:30:43: RADIUS: Received from id 7 10.6.20.70:1709, Accounting-response, len 20
00:30:43: RADIUS: authenticator 78 80 AB D1 82 75 ED ED - E4 1F 12 25 D8 83 F9 6

voice class aaa 3 is applied to dial-peer voice 1000 pots and a call is made. voice class aaa 3 uses server 10.6.20.70 with auth port 1720 and acct port 1721. The radius daemon has not started. AAA accounting and AAA authorization requests are sent to the appropriate server but no acknowledgement is received. Retries are attempted.

00:37:03: %SYS-5-CONFIG_I: Configured from console by console
00:37:11: ISDN Se0:23: RX <- SETUP pd = 8  callref = 0x009E
00:37:11:         Bearer Capability i = 0x8090A2
00:37:11:         Channel ID i = 0xE1808397
00:37:11:         Calling Party Number i = 0x0080, '4081234567', Plan:Unknown, Type:Unknown
00:37:11:         Called Party Number i = 0xE9, '12345', Plan:Private, Type:Abbreviated
00:37:11: RADIUS/ENCODE(00000010): Unsupported AAA attribute timezone
00:37:11: RADIUS(00000010): Encoding nas-port...Only port-type avlbl
00:37:11: RADIUS(00000010): sending
00:37:11: RADIUS: Send to unknown id 8 10.6.20.70:1721, Accounting-Request, len 414
00:37:11: RADIUS: authenticator EC F7 FD AB ED 0D 26 BF - F0 A4 D2 88 91 1E D9 22
00:37:11: RADIUS:  Acct-Session-Id [44]  10  "00000010"
00:37:11: RADIUS:  Vendor, Cisco [26]  56
00:37:11: RADIUS:   h323-setup-time [25]  50  "h323-setup-time=*00:37:09.095 UTC Sat Jan 1 2000"
00:37:11: RADIUS:  Vendor, Cisco [26]  34
00:37:11: RADIUS:   h323-gw-id [33]  28  "h323-gw-id=router."
00:37:11: RADIUS:  Vendor, Cisco [26]  56
00:37:11: RADIUS:   Conf-Id [24]  50  "h323-conf-id=69EAABEB BF1A11D3 8014E483 89ADC43B"
00:37:11: RADIUS:  Vendor, Cisco [26]  31
00:37:11: RADIUS:   h323-call-origin [26]  25  "h323-call-origin=answer"
00:37:11: RADIUS:  Vendor, Cisco [26]  32
00:37:11: RADIUS:   h323-call-type [27]  26  "h323-call-type=Telephony"
00:37:11: RADIUS:  Vendor, Cisco [26]  65
00:37:11: RADIUS:     Cisco AVpair [1]  59  "h323-incoming-conf-id=69EAABEB BF1A11D3 8014E483 89ADC43B"
00:37:11: RADIUS:  Vendor, Cisco [26]  30
00:37:11: RADIUS:     Cisco AVpair [1]  24  "subscriber=RegularLine"
00:37:11: RADIUS:  User-Name [1]  12  "4081234567"
00:37:11: RADIUS:  Acct-Status-Type [40]  6  Start  [1]
00:37:11: RADIUS:  NAS-Port-Type [61]  6  Async  [0]
00:37:11: RADIUS:  Vendor, Cisco [26]  19
00:37:11: RADIUS:  Calling-Station-Id [31]  12  "4081234567"
00:37:11: RADIUS:  Called-Station-Id [30]  7  "12346"
00:37:11: RADIUS:  NAS-IP-Address [4]  6  10.5.20.100
00:37:11: RADIUS:  Delay-Time [41]  6  0
00:37:11: ISDN Se0:23: TX -> CALL_PROC pd = 8  callref = 0x809E
00:37:11: Channel ID i = 0xA98397
00:37:11: ISDN Se0:23: TX -> CONNECT pd = 8  callref = 0x809E
00:37:11: ISDN Se0:23: RX <- CONNECT_ACK pd = 8  callref = 0x009E
00:37:11: ISDN Se0:23: CALL_PROGRESS: CALL_CONNECTED call id 0x65, bchan 22, dsl 0
00:37:11: %ISDN-6-CONNECT: Interface Serial0:22 is now connected to 4081234567
00:37:16: RADIUS: Retransmit id 8
00:37:16: RADIUS: acct-delay-time for 4021D9EC (at 4021DB84) now 5
00:37:17: %ISDN-6-CONNECT: Interface Serial0:22 is now connected to 4081234567
00:37:21: RADIUS: Retransmit id 1
00:37:21: RADIUS: acct-delay-time for 4021D9EC (at 4021DB84) now 10
00:37:26: RADIUS: Retransmit id 2
00:37:26: RADIUS: acct-delay-time for 4021D9EC (at 4021DB84) now 15
00:37:31: RADIUS: Tried all servers.
00:37:31: RADIUS: No valid server found. Trying any viable server
00:37:31: RADIUS: Tried all servers.
00:37:31: RADIUS: No response for id 3
00:37:31: RADIUS/DECODE: parse response no app start; FAIL
00:37:31: RADIUS/DECODE: parse response; FAIL
00:37:35: RADIUS(00000010): Encoding nas-port...Only port-type avlbl
00:37:35: RADIUS/ENCODE(00000010): acct_session_id: 16
00:37:35: RADIUS(00000010): sending
00:37:35: RADIUS: Send to unknown id 5 10.6.20.70:1720, Access-Request, len 199
00:37:35: RADIUS: authenicator 4B 6E 67 9F D4 1E 73 37 - 45 D3 CD 7C 70 FD C7 12
00:37:35: RADIUS: User-Name [1] 8 '7777777'
00:37:35: RADIUS: User-Password [2] 18 *
00:37:35: RADIUS: Vendor, Cisco [26] 56
00:37:35: RADIUS: Conf-Id [24] 50 'h323-conf-id=61A46F2C 00000003 62E664B0 634AA464'
00:37:35: RADIUS: Vendor, Cisco [26] 36
00:37:35: RADIUS: Cisco AVPair [1] 30 'h323-ivr-out=transactionID:5'
00:37:35: RADIUS: Calling-Station-Id [31] 12 '4081234567'
00:37:35: RADIUS: Vendor, Cisco [26] 19
00:37:35: RADIUS: Calling-Station-Id [31] 12 '4081234567'
00:37:35: RADIUS: Service-Type [6] 6 Login
00:37:35: RADIUS: NAS-IP-Address [4] 6 10.5.20.100
00:37:40: RADIUS: Retransmit id 5
00:37:45: RADIUS: Retransmit id 5
00:37:50: RADIUS: Retransmit id 5
00:37:55: RADIUS: Tried all servers.
00:37:55: RADIUS: No valid server found. Trying any viable server
00:37:55: RADIUS: Tried all servers.
00:37:55: RADIUS: No response for id 5
00:37:55: RADIUS/DECODE: parse response no app start; FAIL
00:37:55: RADIUS/DECODE: parse response; FAIL
00:38:00: %ISDN-6-DISCONNECT: Interface Serial0:22 disconnected from 4081234567, call lasted 48 seconds
00:38:00: ISDN Se0:23: TX -> DISCONNECT pd = 8 callref = 0x809E
00:38:00: Cause i = 0x8090 - Normal call clearing
00:38:00: RADIUS/ENCODE(00000010): Unsupported AAA attribute timezone
00:38:00: RADIUS(00000010): Encoding nas-port...Only port-type avlbl
00:38:00: RADIUS(00000010): sending
00:38:00: RADIUS: Send to unknown id 9 10.6.20.70:1721, Accounting-Request, len 660
00:38:00: RADIUS: authenticator C5 79 B7 D3 92 75 37 D0 - E7 5C 84 99 66 97 17
00:38:00: RADIUS: Acct-Session-Id [44] 10 '00000010'
00:38:00: RADIUS: Vendor, Cisco [26] 56
00:38:00: h323-setup-time [25] 50 'h323-setup-time=00:37:09.095 UTC Sat Jan 1 2000'
00:38:00: RADIUS: Vendor, Cisco [26] 34
00:38:00: h323-gw-id [33] 28 'h323-gw-id=router.'
00:38:00: RADIUS: Vendor, Cisco [26] 56
00:38:00: Conf-Id [24] 50 'h323-conf-id=69EAABEB BF1A11D3 80145830D4C38'
00:38:00: RADIUS: Vendor, Cisco [26] 31
00:38:00: h323-call-origin [26] 25 'h323-call-origin=answer'
00:38:00: RADIUS: Vendor, Cisco [26] 32
00:38:00: h323-call-type [27] 26 'h323-call-type=Telephony'
00:38:00: RADIUS: Vendor, Cisco [26] 65
Using Show Commands

**show call accounting voice summary**

The `show call accounting voice summary` command shows the status of all accounting templates that are defined and loaded.

```
show call accounting voice summary
name          url                                      last_load  is_running
=========================================================================
 cdr1          tftp://10.255.255.255/johndoe/sanjose/  success    is running
 cdr2          tftp://10.255.255.255/johndoe/sanjose/  success    is running
```
**show call accounting-template voice <template name>**

**show call accounting-template voice <template name>** shows the VSAs that are contained in the accounting template.

**show call accounting-template voice cdr1**

CDR template cdr1 is running

```
  url: tftp://10.255.255.255/johndoe/sanjose/cdr/cdr1.cdr
```

The last load was successful.

```
  attr:  h323-call-origin (56)
```

Totally 1 attrs defined.

**show call accounting-template voice cdr2**

CDR template cdr2 is running

```
  url: tftp://10.255.255.255/johndoe/sanjose/cdr/cdr2.cdr
```

The last load was successful.

```
  attr:  h323-call-origin (56)
```

Totally 1 attrs defined.

The output below results from defining a template that does not exist or that cannot be reached.

```
router(config)#$://10.255.255.255/johndoe/sanjose/cdr/cdr4000.cdr
Reading cdr template cdr10 fail, put it on retry queue.
```

01:15:46: hifs ifs could not open file

The output below is for a template with an invalid VSA.

**sh call accounting-template voice cdr1**

CDR template cdr1 is running

```
  url: tftp://10.255.255.255/johndoe/sanjose/cdr/cdr1.cdr
```

The last load was successful.

```
  attr:  h323-call-origin (56)
```

Totally 1 attrs defined.

Template cdr1.cdr is modified on the tftp server to enable an invalid VSA (for example h323-call-origin) to be put into the template.

**call accounting-template voice reload cdr1**

Loading johndoe/sanjose/cdr/cdr1.cdr from 10.255.255.255 (via Ethernet0): !

```
[OK - 88/4096 bytes]
```

```
cam: Fail to reload cdr template cdr1, unloading ...
```

02:27:29: hifs ifs file read succeeded. size=88,
url=tftp://10.255.255.255/johndoe/sanjose/cdr/cdr1.cdr

02:27:29: Error: attr name invalid-vsa-h323-call-origin (0) is not valid in line 3.

**sh call accounting-template voice cdr1**

CDR template cdr1 is running

```
  url: tftp://10.255.255.255/johndoe/sanjose/cdr/cdr1.cdr
```

Last load returned errno=8, Exec format error
attr: h323-call-origin (56)

Totally 1 attrs defined.

The template has been rejected, and previous template still applied.

show call aaa attributes

display the VSAs that are supported. Mandatory VSAs that are sent to the
accounting server are not displayed.

sh call aaa attributes

AAA ATTRIBUTE LIST:
Name=disc-cause-ext  Format=Enum
Name=Acct-Status-Type  Format=Enum
Name=acl  Format=Ulong
Name=addr  Format=IP v4 Address

......
Name=gw-password  Format=Binary
Name=h323-billing-model  Format=String
Name=h323-call-origin  Format=String

!Use the show call accounting-template voice summary command to check if a template is
loaded and running.
!The output below shows two templates successfully loaded and running, and a template that
failed to load.

show call accounting-template voice summary

name  url  last_load   is_running
=========================================================================
cdr1  tftp://10.255.255.255/johndoe/sanjose/  success   is running
cdr2  tftp://10.255.255.255/johndoe/sanjose/  success   is running
cdr10 tftp://10.255.255.255/johndoe/sanjose/  fail       is not running

!The output below shows reloading template cdr1 after modifying it.
!Initially, the original template cdr1 is loaded as shown:
show call accounting-template voice cdr1

CDR template cdr1 is running
url: tftp://10.255.255.255/johndoe/sanjose/cdr/cdr1.cdr
The last load was successful.

attr: h323-call-origin (56)

Totally 1 attrs defined.

!Additional VSAs are added to modify cdr1 on the tftp server as shown:

call accounting
call accounting-template voice reload cdr1

Loading johndoe/sanjose/cdr/cdr1.cdr from 10.255.255.255 (via Ethernet0): !
[OK - 1848/3072 bytes]

cam: Reload cdr template cdr1 success.
01:35:58: hifs ifs file read succeeded. size=1848,
url=tftp://10.255.255.255/johndoe/sanjose/cdr/cdr1.cdr

show call accounting-template voice cdr1

CDR template cdr1 is running
url: tftp://10.255.255.255/johndoe/sanjose/cdr/cdr1.cdr
The last load was successful.
attr: h323-call-origin (56)
debug radius accounting

In the output below, cdr1 includes h323-call-origin but does not include VSA h323-gw-id. cdr2 includes h323-gw-id but does not include h323-call-origin.

show call accounting-template voice cdr1
CDR template cdr1 is running
    url: tftp://10.255.255.255/johndoe/sanjose/cdr/cdr1.cdr
    The last load was successful.

    attr:  h323-call-origin (56)

    Totally 1 attrs defined.

show call accounting-template voice cdr2
CDR template cdr2 is running
url: tftp://10.255.255.255/johndoe/sanjose/cdr/cdr2.cdr
The last load was successful.

attr: h323-gw-id (65)

Totally 1 attrs defined.

The output below is from a call that uses cdr1.cdr which allows only h323-call-origin.

debug radius accounting
Radius protocol debugging is on
Radius packet hex dump debugging is off
Radius packet protocol (authentication) debugging is off
Radius packet protocol (accounting) debugging is on

02:41:32: RADIUS/ENCODE(00000023): Unsupported AAA attribute timezone
02:41:32: RADIUS(00000023): Encoding nas-port...Only port-type avlbl
02:41:32: RADIUS(00000023): sending
02:41:32: RADIUS: Send to unknown id 26 10.6.20.70:1699, Accounting-Request, len 262
02:41:32: RADIUS: authenticator 84 6E A0 C0 0F 27 79 03 - 59 96 FC 6C F4 17 05 4D
02:41:32: RADIUS: Acct-Session-Id [44] 10 "00000023"
02:41:32: RADIUS: Vendor, Cisco [26] 56
02:41:32: RADIUS: Conf-Id [24] 50 "h323-conf-id=C925CD59 BF2B11D3 8038E843 89AD43B" 8038E843 89AD43B" 8038E843 89AD43B" 8038E843 89AD43B"
02:41:32: RADIUS: Vendor, Cisco [26] 31
02:41:32: RADIUS: Vendor, Cisco [26] 65
02:41:32: RADIUS: Acct-Status-Type [40] 6 Start [1]
02:41:32: RADIUS: NAS-Port-Type [61] 6 Async [0]
02:41:32: RADIUS: Vendor, Cisco [26] 19
02:41:32: RADIUS: Calling-Station-Id [31] 12 "4081234567"
02:41:32: RADIUS: Called-Station-Id [30] 7 "12345"
02:41:32: RADIUS: Delay-Time [41] 6 0
02:41:32: RADIUS: Received from id 26 10.6.20.70:1699, Accounting-response, len 20
02:41:32: RADIUS: authenticator 90 AD C8 09 60 D7 26 01 - DE E0 BC DC C1 F8 CA 2F
02:41:32: %ISDN-6-CONNECT: Interface Serial0:22 is now connected to 4081234567
02:41:38: %ISDN-6-CONNECT: Interface Serial0:22 is now connected to 4081234567
02:41:52: RADIUS(00000023): Encoding nas-port...Only port-type avlbl
02:41:59: %ISDN-6-DISCONNECT: Interface Serial0:22 disconnected from 4081234567, call lasted 26 seconds
02:41:59: RADIUS/ENCODE(00000023): Unsupported AAA attribute timezone
02:41:59: RADIUS(00000023): Encoding nas-port...Only port-type avlbl
02:41:59: RADIUS(00000023): sending
02:41:59: RADIUS: Send to unknown id 27 10.6.20.70:1699, Accounting-Request, len 327
02:41:59: RADIUS: authenticator 13 B7 10 EE 1C 55 7A D2 - 0F 4A A5 2F 1F 85 0E 3A
02:41:59: RADIUS: Acct-Session-Id [44] 10 "00000023"
02:41:59: RADIUS: Vendor, Cisco [26] 56
02:41:59: RADIUS: Vendor, Cisco [26] 31
02:41:59: RADIUS: Vendor, Cisco [26] 65
02:41:59: RADIUS: Acct-Input-Octets [42] 6 0
02:41:59: RADIUS: Acct-Input-Packets [47] 6 0
02:41:59: RADIUS: Acct-Output-Packets [48] 6 760
02:41:59: RADIUS: Vendor, Cisco [26] 35
02:41:59: RADIUS: NAS-Port-Type [61] 6 Async [0]
02:41:59: RADIUS: Vendor, Cisco [26] 19
02:41:59: RADIUS: Calling-Station-Id [31] 12 "4081234567"
02:41:59: RADIUS: Called-Station-Id [30] 7 "12345"
02:41:59: RADIUS: Delay-Time [41] 6 0
02:41:59: RADIUS: Received from id 27 10.6.20.70:1699, Accounting-response, len 20
02:41:59: RADIUS: authenticator 7F B2 88 3A 4A 96 05 C6 - D5 81 19 D8 25 3B 4D CB

The output below is from the show debug command.

Radius protocol debugging is on
Radius packet protocol (accounting) debugging is on

! The output below is from a call that uses cdr2 which allows h323-gw-id, but does not allow h323-call-origin.

RADIUS/ENCODE(00000025): Unsupported AAA attribute timezone
02:51:35: RADIUS(00000025): Encoding nas-port...Only port-type avlbl
02:51:35: RADIUS(00000025): sending
02:51:35: RADIUS: Send to unknown id 28 10.6.20.70:1709, Accounting-Request, len 265
02:51:35: RADIUS: authenticator 15 F0 7E AB 75 07 10 70 - 5E 3C 54 78 09 18 83 E5
02:51:35: RADIUS: Acct-Session-Id [44] 10 "00000025"
02:51:35: RADIUS: Vendor, Cisco [26] 34
02:51:35: RADIUS: h323-gw-id [33] 28 "h323-gw-id=router."
02:51:35: RADIUS: Vendor, Cisco [26] 56
02:51:35: RADIUS: Conf-Id [24] 50 "h323-conf-id=306F55DD BF2D11D3 803CE483 89AD43B"
02:51:35: RADIUS: Vendor, Cisco [26] 65
02:51:35: RADIUS: Cisco AVpair [1] 59 "h323-incoming-conf-id=306F55DD BF2D11D3 803CE483 89AD43B"
02:51:35: RADIUS: User-Name [1] 12 "4081234567"
02:51:35: RADIUS: Acct-Status-Type [40] 6 Start [1]
02:51:35: RADIUS: NAS-Port-Type [61] 6 Async [0]
02:51:35: RADIUS: Vendor, Cisco [26] 19
02:51:35: RADIUS: Calling-Station-Id [31] 12 "4081234567"
02:51:35: RADIUS: Called-Station-Id [30] 7 "12345"
02:51:35: RADIUS: Delay-Time [41] 6 0
02:51:35: %ISDN-6-CONNECT: Interface Serial1:22 is now connected to 4081234567
02:51:35: RADIUS: Received from id 28 10.6.20.70:1709, Accounting-response, len 20
02:51:35: %ISDN-6-DISCONNECT: Interface Serial1:22 disconnected from 4081234567
The call lasted 8 seconds
02:51:43: RADIUS/ENCODE(00000025): Unsupported AAA attribute timezone
02:51:43: RADIUS(00000025): Encoding nas-port...Only port-type avlbl
02:51:43: RADIUS(00000025): sending
02:51:43: RADIUS: Send to unknown id 29 10.6.20.70:1709, Accounting-Request, len 330
02:51:43: RADIUS: authenticator 55 35 AB CC 20 64 69 4B - 3F EE 79 04 11 E8 AE 4F
02:51:43: RADIUS: Acct-Session-Id [44] 10 "00000025"
02:51:43: RADIUS: Vendor, Cisco [26] 34
02:51:43: RADIUS: Vendor, Cisco [26] 56
02:51:43: RADIUS: Conf-Id [24] 50 "h323-conf-id=306F55DD BF2D11D3 803CE483 89AD43B"
02:51:43: RADIUS: Vendor, Cisco [26] 65
02:51:43: RADIUS: Acct-Input-Octets [42] 6 0
02:51:43: RADIUS: Acct-Input-Packets [47] 6 0
02:51:43: RADIUS: Vendor, Cisco [26] 35
02:51:43: RADIUS: NAS-Port-Type [61] 6 Async [0]
02:51:43: RADIUS: Vendor, Cisco [26] 19
02:51:43: RADIUS: Calling-Station-Id [31] 12 "4081234567"
02:51:43: RADIUS: Called-Station-Id [30] 7 "12346"
02:51:43: RADIUS: Delay-Time [41] 6 0
02:51:43: RADIUS: Received from id 29 10.6.20.70:1709, Accounting-response, len 20
02:51:43: RADIUS: authenticator 45 31 ED 45 F4 06 ED 54 5E 6F 83 64 4D 2D 34 90
ASCII Table

<table>
<thead>
<tr>
<th>Value (Hex)</th>
<th>Character</th>
<th>Name</th>
<th>Value (Hex)</th>
<th>Character</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td></td>
<td>Space</td>
<td>21</td>
<td>!</td>
<td>Exclamation mark</td>
</tr>
<tr>
<td>22</td>
<td>&quot;</td>
<td>Double quote</td>
<td>23</td>
<td>#</td>
<td>Pound</td>
</tr>
<tr>
<td>24</td>
<td>$</td>
<td>Dollar</td>
<td>25</td>
<td>%</td>
<td>Percent</td>
</tr>
<tr>
<td>26</td>
<td>&amp;</td>
<td>Ampersand</td>
<td>27</td>
<td>'</td>
<td>Right single quote</td>
</tr>
<tr>
<td>28</td>
<td>(</td>
<td>Left parenthesis</td>
<td>29</td>
<td>)</td>
<td>Right parenthesis</td>
</tr>
<tr>
<td>2A</td>
<td>*</td>
<td>Astrick</td>
<td>2B</td>
<td>+</td>
<td>Plus</td>
</tr>
<tr>
<td>2C</td>
<td>,</td>
<td>Comma</td>
<td>2D</td>
<td>-</td>
<td>Hyphen</td>
</tr>
<tr>
<td>2E</td>
<td>.</td>
<td>Dot</td>
<td>2F</td>
<td>/</td>
<td>Slash</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>Zero</td>
<td>31</td>
<td>1</td>
<td>One</td>
</tr>
<tr>
<td>32</td>
<td>2</td>
<td>Two</td>
<td>33</td>
<td>3</td>
<td>Three</td>
</tr>
<tr>
<td>34</td>
<td>4</td>
<td>Four</td>
<td>35</td>
<td>5</td>
<td>Five</td>
</tr>
<tr>
<td>36</td>
<td>6</td>
<td>Six</td>
<td>37</td>
<td>7</td>
<td>Seven</td>
</tr>
<tr>
<td>38</td>
<td>8</td>
<td>Eight</td>
<td>39</td>
<td>9</td>
<td>Nine</td>
</tr>
<tr>
<td>3A</td>
<td>:</td>
<td>Colon</td>
<td>3B</td>
<td>;</td>
<td>Semi-colon</td>
</tr>
<tr>
<td>3C</td>
<td>&lt;</td>
<td>Less than</td>
<td>3D</td>
<td>=</td>
<td>Equal</td>
</tr>
<tr>
<td>3E</td>
<td>&gt;</td>
<td>Greater than</td>
<td>3F</td>
<td>?</td>
<td>Question mark</td>
</tr>
<tr>
<td>40</td>
<td>@</td>
<td>at</td>
<td>41</td>
<td>A</td>
<td>a</td>
</tr>
<tr>
<td>42</td>
<td>B</td>
<td>b</td>
<td>43</td>
<td>C</td>
<td>c</td>
</tr>
<tr>
<td>44</td>
<td>D</td>
<td>d</td>
<td>45</td>
<td>E</td>
<td>e</td>
</tr>
</tbody>
</table>
### Table 1  
*Prompt Name for ASCII Characters (continued)*

<table>
<thead>
<tr>
<th>Value (Hex)</th>
<th>Character</th>
<th>Name</th>
<th>Value (Hex)</th>
<th>Character</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>46</td>
<td>F</td>
<td>f</td>
<td>47</td>
<td>G</td>
<td>g</td>
</tr>
<tr>
<td>48</td>
<td>H</td>
<td>h</td>
<td>49</td>
<td>I</td>
<td>i</td>
</tr>
<tr>
<td>4A</td>
<td>J</td>
<td>j</td>
<td>4B</td>
<td>K</td>
<td>k</td>
</tr>
<tr>
<td>4C</td>
<td>L</td>
<td>l</td>
<td>4D</td>
<td>M</td>
<td>m</td>
</tr>
<tr>
<td>4E</td>
<td>N</td>
<td>n</td>
<td>4F</td>
<td>O</td>
<td>o</td>
</tr>
<tr>
<td>50</td>
<td>P</td>
<td>p</td>
<td>51</td>
<td>Q</td>
<td>q</td>
</tr>
<tr>
<td>52</td>
<td>R</td>
<td>r</td>
<td>53</td>
<td>S</td>
<td>s</td>
</tr>
<tr>
<td>54</td>
<td>T</td>
<td>t</td>
<td>55</td>
<td>U</td>
<td>u</td>
</tr>
<tr>
<td>56</td>
<td>V</td>
<td>v</td>
<td>57</td>
<td>W</td>
<td>w</td>
</tr>
<tr>
<td>58</td>
<td>X</td>
<td>x</td>
<td>59</td>
<td>Y</td>
<td>y</td>
</tr>
<tr>
<td>5A</td>
<td>Z</td>
<td>z</td>
<td>5B</td>
<td>[</td>
<td>Left bracket</td>
</tr>
<tr>
<td>5C</td>
<td>\</td>
<td>Backslash</td>
<td>5D</td>
<td>]</td>
<td>Right bracket</td>
</tr>
<tr>
<td>5E</td>
<td>^</td>
<td>Caret</td>
<td>5F</td>
<td>_</td>
<td>Underscore</td>
</tr>
<tr>
<td>60</td>
<td>'</td>
<td>Left single quote</td>
<td>61</td>
<td>a</td>
<td>a</td>
</tr>
<tr>
<td>62</td>
<td>b</td>
<td>b</td>
<td>63</td>
<td>c</td>
<td>c</td>
</tr>
<tr>
<td>64</td>
<td>d</td>
<td>d</td>
<td>65</td>
<td>e</td>
<td>e</td>
</tr>
<tr>
<td>66</td>
<td>f</td>
<td>f</td>
<td>67</td>
<td>g</td>
<td>g</td>
</tr>
<tr>
<td>68</td>
<td>h</td>
<td>h</td>
<td>69</td>
<td>i</td>
<td>i</td>
</tr>
<tr>
<td>6A</td>
<td>j</td>
<td>j</td>
<td>6B</td>
<td>k</td>
<td>k</td>
</tr>
<tr>
<td>6C</td>
<td>l</td>
<td>l</td>
<td>6D</td>
<td>m</td>
<td>m</td>
</tr>
<tr>
<td>6E</td>
<td>n</td>
<td>n</td>
<td>6F</td>
<td>o</td>
<td>o</td>
</tr>
<tr>
<td>70</td>
<td>p</td>
<td>p</td>
<td>71</td>
<td>q</td>
<td>q</td>
</tr>
<tr>
<td>72</td>
<td>r</td>
<td>r</td>
<td>73</td>
<td>s</td>
<td>s</td>
</tr>
<tr>
<td>74</td>
<td>t</td>
<td>t</td>
<td>75</td>
<td>u</td>
<td>u</td>
</tr>
<tr>
<td>76</td>
<td>v</td>
<td>v</td>
<td>77</td>
<td>w</td>
<td>w</td>
</tr>
<tr>
<td>78</td>
<td>x</td>
<td>x</td>
<td>79</td>
<td>y</td>
<td>y</td>
</tr>
<tr>
<td>7A</td>
<td>z</td>
<td>z</td>
<td>7B</td>
<td>{</td>
<td>Left brace</td>
</tr>
<tr>
<td>7C</td>
<td>l</td>
<td>Pipe</td>
<td>7D</td>
<td>}</td>
<td>Right brace</td>
</tr>
<tr>
<td>7E</td>
<td>~</td>
<td>Tilde</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### GLOSSARY

<table>
<thead>
<tr>
<th>A</th>
<th>AAA</th>
<th>Authentication, Authorization, and Accounting. AAA is a suite of network security services that provides the primary framework through which you can set up access control on your Cisco router or gateway.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANI</td>
<td>Automatic number identification. Same as calling party.</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>DNIS</td>
<td>Dialed number identification service. Same as the called number.</td>
</tr>
<tr>
<td>G</td>
<td>gatekeeper</td>
<td>A gatekeeper maintains a registry of devices in the multimedia network. The devices register with the gatekeeper at startup and request admission to a call from the gatekeeper. An H.323 gateway is an endpoint on the LAN that provides real-time, two-way communications between H.323 terminals on the LAN and other ITU-T terminals in the WAN or to another H.323 gateway.</td>
</tr>
<tr>
<td></td>
<td>gateway</td>
<td>A gateway allows H.323 terminals to communicate with non-H.323 terminals by converting protocols. A gateway is the point where a circuit-switched call is encoded and repackaged into IP packets.</td>
</tr>
<tr>
<td>I</td>
<td>IFS</td>
<td>Cisco IOS file system.</td>
</tr>
<tr>
<td>IVR</td>
<td>Interactive voice response. When someone dials in, IVR responds with a prompt to get a personal identification number (PIN), and so on.</td>
<td></td>
</tr>
</tbody>
</table>
**P**

**POTS**
Plain old telephone service. Basic telephone service supplying standard single line telephones, telephone lines, and access to the PSTN.

**PSTN**
Public Switched Telephone Network. PSTN refers to the local telephone company.

---

**V**

**VoIP**
Voice over IP. The ability to carry normal telephone-style voice signals over an IP-based network with POTS-like functionality, reliability, and voice quality. VoIP is a blanket term that generally refers to the Cisco open standards-based (for example, H.323) approach to IP voice traffic.