Configuring SIP AAA Features

This chapter describes how to configure the following SIP AAA features:

- Configurable Screening Indicator (handled in this document as a subset of SIP - Enhanced Billing Support for Gateways)

- RADIUS Pre-authentication for Voice Calls

- SIP - Enhanced Billing Support for Gateways

- SIP: Gateway HTTP Authentication Digest

Feature History for Configurable Screening Indicator

(Introduced as part of the SIP Gateway Support of RSVP and TEL URL feature)

Release	Modification
12.2(2)XB	This feature was introduced.
12.2(8)T	This feature was integrated into this release.

Feature History for RADIUS Pre-authentication for Voice Calls

Release	Modification
12.2(11)T	This feature was introduced.

Feature History for SIP - Enhanced Billing Support for SIP Gateways

Release	Modification
12.2(2)XB	This feature was introduced.
12.2(8)T	This feature was integrated into this release.
12.2(11)T	This feature was implemented on additional platforms.

Feature History for SIP: Gateway HTTP Authentication Digest

Release	Modification
12.3(8)T	This feature was introduced.

Finding Support Information for Platforms and Cisco Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn . An account on Cisco.com is not required.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for SIP AAA

All SIP AAA Features

- Establish a working IP network. For information about configuring IP, see *Cisco IOS IP Command Reference*, Release 12.3

- Configure VoIP. For information about configuring VoIP, see the following:

    - *Cisco IOS Voice Configuration Library*, Release 12.4T
    - *Cisco IOS Voice Command Reference*
    - "Enhancements to the Session Initiation Protocol for VoIP on Cisco Access Platforms"

- Ensure that the gateway has voice functionality configured for SIP.

RADIUS Pre-authentication for Voice Calls Feature

- Ensure that you have an application that supports preauthentication.

- Set up preauthentication profiles and have them running on a RADIUS-based PPM server in your network.

• Enable gateway accounting using the gw-accounting command. All call-accounting information must be forwarded to the server that is performing preauthentication. Accounting stop packets must be sent to this server so that call billing is ended when calls are disconnected from the gateway. In addition, authentication and accounting start packets are needed to enable other features, such as virtual private dialup network (VPDN).

Note For information on setting up the preauthentication profiles, see the *Cisco IOS Security Command Reference*. For information on Cisco RPMS, see the Cisco Resource Policy Management System 2.0. For standards supporting RADIUS-based PPM servers, see RFC 2865, Remote Authentication Dial In User Service (RADIUS).

SIP: Gateway HTTP Authentication Digest Feature

• Implement a Cisco IOS SIP gateway that supports SIP.

• Implement a configuration that supports SIP.

• Implement an authentication configuration for the gateway to respond to authentication challenges for requests that it originates.

Restrictions for SIP AAA

All SIP AAA Features

• If Cisco Resource Policy Management System (RPMS) is used as the RADIUS-based PPM server, it must be Version 2.0 or a later release.

• In SIP environments, if you want the Cisco SIP proxy server to generate preauthentication queries, you must run Cisco SPS 2.0 or a later version.

SIP: Gateway HTTP Authentication Digest Via SIP UA Feature

• SIP Register is supported only on platforms with digital trunk type ports.

Information About SIP AAA

AAA features for SIP provide the following benefits:

• RADIUS preauthentication allows wholesalers to accept or reject calls to enforce SLAs before calls are connected, thereby conserving gateway resources.

• Call admission control prevents call connections when resources are unavailable.

• Extended dial plan features enable the call service type to be determined from preauthentication request data, simplifying dial plan entries.

• Universal gateways provide other specific benefits:

- Flexibility in deploying new services and adapting to changes in the business environment
- Cost savings through reduction of total number of ports required to provide different services
- Optimized utilization of access infrastructure by supporting more services during off-peak hours
- Flexibility in access network engineering by leveraging dial infrastructure to handle both dial and voice

To configure AAA features for SIP, you should understand the following concepts:

RADIUS Pre-authentication for Voice Calls

This section explains how to configure the AAA RADIUS communication link between a universal gateway and a RADIUS-based PPM server for RADIUS preauthentication.

Information about an incoming call is relayed through the gateway to the RADIUS-based PPM server in the network before the call is connected. The RADIUS-based PPM server provides port policy management and preauthentication by evaluating the call information against contracted parameter levels in SLAs. If the call falls within SLA limits, the server preauthenticates the call and the universal gateway accepts it. If the server does not authorize the call, the universal gateway sends a disconnect message to the public network switch to reject the call. The available call information includes one or more of the following:

- DNIS number, also referred to as the called number.

- CLID number (calling line identification number), also referred to as the calling number.

- Call type, also referred to as the bearer capability.

- IP address of the originating domain.

- Interzone ClearToken (IZCT) information, which contains the origination gatekeeper zone name for intradomain calls or the origination domain border gatekeeper zone name for interdomain calls. Whenever IZCT information is available, it is used to preauthenticate leg-3 H.323 VoIP calls.

Note To enable IZCT, use the **security izct password** command on the gatekeeper. For multiple gatekeeper zones, use the **lrq forward-queries**command. For information on IZCT configuration, see *Inter-Domain Gatekeeper Security Enhancement*, Release 12.2(4)T.

A timer monitors the preauthentication query in case the RADIUS-based PPM server application is unavailable or slow to respond. If the timer expires before an acceptance or rejection is provided, the universal gateway rejects the call.

The RADIUS Pre-authentication for Voice Calls feature supports the use of RADIUS attributes that are configured in RADIUS preauthentication profiles to specify preauthentication behavior. These attributes can also be used, for instance, to specify whether subsequent authentication should occur and, if so, what authentication method should be used.

The commands in this section are used for both leg 1 calls (calls from a PSTN that enter an incoming, or originating, gateway) and leg 3 calls (calls that exit the IP network to an outgoing, or terminating, gateway). The use of optional commands depends on individual network factors.

Note   Before configuring AAA preauthentication, you must make sure that the supporting preauthentication application is running on a RADIUS-based PPM server in your network, such as a Cisco RPMS. You must also set up preauthentication profiles on the RADIUS-based PPM server. For full information on AAA, see the *Cisco IOS Security Configuration Guide*.

The RADIUS Pre-authentication for Voice Calls feature provides the means to evaluate and accept or reject call setup requests for both voice and dial calls received at universal gateways. This process is known as preauthentication. The feature also optionally allows voice calls to bypass this evaluation.

With universal gateways, voice customers and dial customers contend for the same gateway resources. This competition can present problems for IP service wholesalers who lease their IP services to various customers such as Internet service providers (ISPs), Internet telephony service providers (ITSPs), and telephony application service providers (T-ASPs). Wholesalers need a way to implement and enforce with these customers service-level agreements (SLAs) that describe the levels of connectivity, performance, and availability that they guarantee to provide. The RADIUS Pre-authentication for Voice Calls feature allows a wholesaler to determine whether a call is within SLA limits before gateway resources are dedicated to terminating the call.

With RADIUS preauthentication enabled, end customers from over-subscribed service providers are prevented from consuming ports that exceed the number allotted to their service provider in its SLA. If the call is accepted in the preauthentication step, it proceeds to full dial authentication and authorization or to voice dial-peer matching and voice session application authentication and authorization.

RADIUS preauthentication uses a RADIUS-based port-policy management (PPM) server, such as the Cisco Resource Policy Management System (RPMS), to interpret and enforce universal PPM and preauthentication SLAs. RADIUS provides the communication link between the PPM server and universal gateways.

Customer profiles are defined in the PPM server with information from the SLA. Then, when a call is received at the universal gateway, the server determines which specific customer SLA policy to apply to the call on the basis of information associated with the call. For example, calls can be identified as either dial or voice on the basis of the called number (also called the dialed number identification service number or DNIS). Then the PPM server might be set up to allow only a certain number of dial calls. When a new dial call is received, it is rejected if adding it to the count makes the count exceed the number of dial calls stipulated in the SLA.

Calls that are accepted by the PPM server continue with their normal call setup sequences after preauthentication. The response from the PPM server is returned to the calling entity--such as an ISDN or SIP call signaling interface--which then proceeds with the regular call flow. Calls that are rejected by the PPM server follow the given call model and apply the error codes or rejection reasons that are specified by the signaling entity.

SIP-Based Voice Termination

In the figure below, a voice call from a SIP telephone or SIP terminal is sent from an ITSP to a wholesaler.The Cisco SIP proxy server (Cisco SPS) chooses the appropriate universal gateway to which the SIP INVITE is forwarded, on the basis of its own routing mechanism. In Step 3, Cisco SPS makes a preauthentication query to the RPMS-based PPM server. Cisco SPS locks out calls that are rejected by the RPMS-based PPM server. In Step 5, the universal gateway makes a preauthentication reservation request to the RPMS-based PPM server, which locks in the resources to handle the call.

Note   This scenario requires Cisco SPS 2.0.

Figure 1: SIP-Based Voice Termination
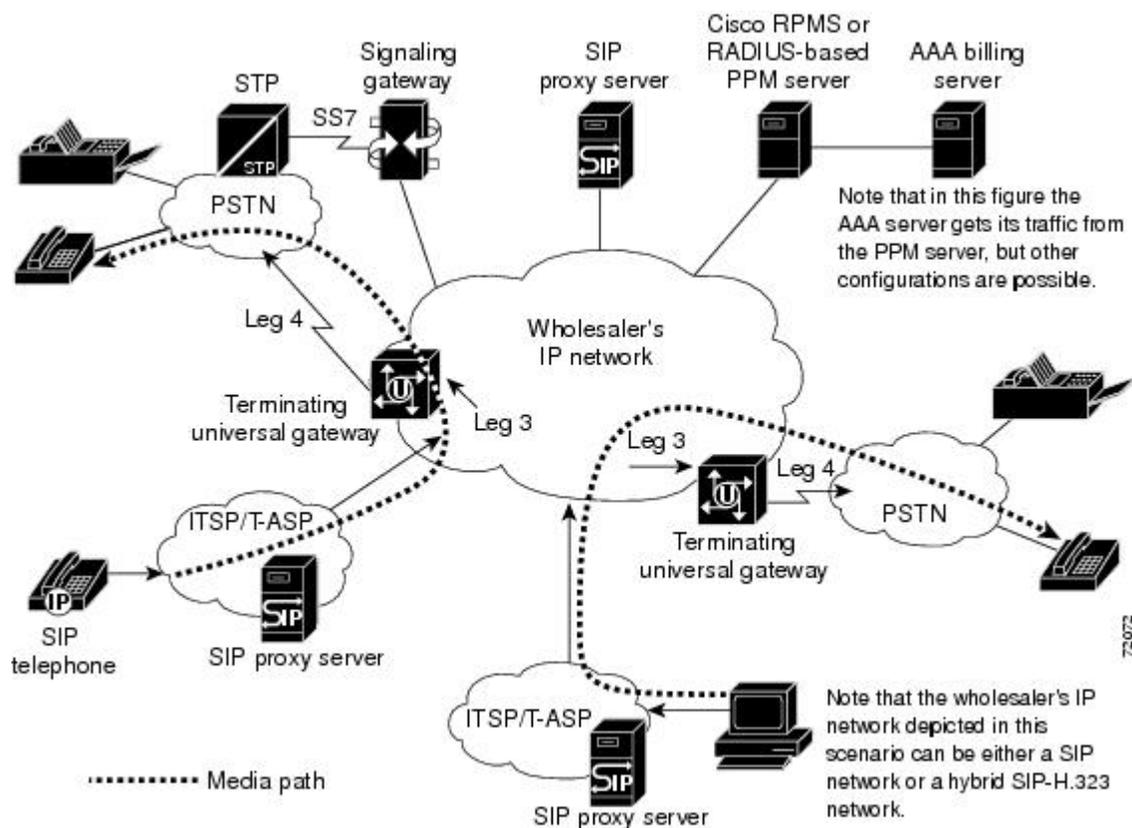
The call flow is as follows:

1. A SIP INVITE is sent from an end user's PC to an ITSP SIP proxy server.

2. The ITSP's SIP proxy server forwards the SIP INVITE to a Cisco SPS at a wholesaler or ISP.

3. Preauthentication--The Cisco SPS sends a preauthentication query to the RADIUS-based PPM server, which locates the appropriate SLA and makes sure that the call is within the SLA limits. If the call is outside the limits, the call is rejected and Cisco SPS responds to the sender with an "Error code 480 - Temporarily not available" message. Cisco SPS interaction with the RADIUS-based PPM server is optional and requires Cisco SPS version 2.0 or a later release. If you are not using Cisco SPS 2.0, the gateway makes the preauthentication query to the RADIUS-based PPM server if it has been configured to do so.

4. Gateway selection--If the preauthentication request is accepted, the Cisco SPS uses its routing logic to determine the appropriate terminating universal gateway to which it should forward the INVITE.

5. Call admission control--If the preauthentication request is accepted, the terminating universal gateway checks its configured call admission control limits. If the call is outside the limits, the call is rejected.

6. Authentication and authorization--The universal gateway reserves a port and sends an authentication, authorization, and accounting (AAA) accounting start packet to the RADIUS-based PPM server.

7. The connection between the caller and the universal gateway is completed (call leg 3).

8. The caller is connected to the PSTN (call leg 4).

9. Accounting stop--After the caller hangs up or is otherwise disconnected, the terminating universal gateway issues an accounting stop packet to the RADIUS-based PPM server. The PPM server uses the accounting stop packet to clear out the count for that call against the SLA.

SIP - Enhanced Billing Support for Gateways

This section describes the SIP - Enhanced Billing Support for Gateways feature. The feature describes the changes to authentication, authorization, and accounting (AAA) records and the Remote Authentication Dial-In User Service (RADIUS) implementations on Cisco SIP gateways. These changes were introduced to provide customers and partners the ability to effectively bill for traffic transported over SIP networks.

Username Attribute

The username attribute is included in all AAA records and is the primary means for the billing system to identify an end user. The password attribute is included in authentication and authorization messages of inbound VoIP call legs.

For most implementations, the SIP gateway populates the username attribute in the SIP INVITE request with the calling number from the FROM: header, and the password attribute with null or with data from an IVR script. If a Proxy-Authorization header exists, it is ignored. The **aaa username** command determines the information with which to populate the username attribute.

Within the Microsoft Passport authentication service that authenticates and identifies users, the passport user ID (PUID) is used. The PUID and a password are passed from a Microsoft network to the Internet telephony service provider (ITSP) network in the Proxy-Authorization header of a SIP INVITE request as a single, base-64 encoded string. For example,

```
Proxy-Authorization: basic MDAwMzAwMDA4MDM5MzJlNjou
```

The **aaa username** command enables parsing of the Proxy-Authorization header; decoding of the PUID and password; and populating of the PUID into the username attribute, and the decoded password into the password attribute. The decoded password is generally a "." because a Microsoft Network (MSN) authenticates users prior to this point. For example,

```
Username = "123456789012345"
Password = "Z\335\304\326KU\037\301\261\326GS\255\242\002\202"
```

The password in the example above is an encrypted "." and is the same for all users.

SIP Call ID

From the Call ID header of the SIP INVITE request, the SIP Call ID is extracted and populated in Cisco vendor-specific attributes *(* VSA) as an attribute value pair *call-id=string* . The value pair can be used to correlate RADIUS records from Cisco SIP gateways with RADIUS records from other SIP network elements for example, proxies.

Note   For complete information on this attribute value pair, see the *RADIUS Vendor-Specific Attributes Voice Implementation Guide* .

Session Protocol

Session Protocol is another attribute value pair that indicates whether the call is using SIP or H.323 as the signaling protocol.

Note   For complete information on this attribute value pair, see the *RADIUS Vendor-Specific Attributes Voice Implementation Guide* .

Silent Authentication Script

As part of the SIP - Enhanced Billing Support for SIP Gateways feature, a Tool Command Language (Tcl) Interactive Voice Response (IVR) 2.0 Silent Authorization script has been developed. The Silent Authorization script allows users to be authorized without having to separately enter a username or password into the system. The script automatically extracts the passport user ID (PUID) and password from the SIP INVITE request, and then authenticates that information through RADIUS authentication and authorization records. The script is referred to as *silent* since neither the caller or called party hears any prompts.

Note   You can upgrade to the latest script version through the CCO Software Center. You can download the app_passport_silent.2.0.0.0.tcl script from http://www.cisco.com/cgi-bin/tablebuild.pl/tclware . You must be a registered CCO user to log in and access these files.

  • For information regarding Tcl IVR API 2.0, see the *Tcl IVR API Version 2.0 Programmer's Guide.*

Developers using the Tcl Silent Authorization script may be interested in joining the Cisco Developer Support Program. This program provides you with a consistent level of support that you can depend on while leveraging Cisco interfaces in your development projects. It also provides an easy process to open, update, and track issues through Cisco.com. The Cisco web-site is a key communication vehicle for using the Cisco Online Case tracking tool. A signed Developer Support Agreement is required to participate in this program. For more details, and access to this agreement, please visit us at http://www.cisco.com/en/US/products/svcs/ps3034/ps5408/ps5418/serv_home.html or contact developer-support@cisco.com.

Configurable Screening Indicator

Screening Indicator (SI) is a signaling-related information element found in octet 3a of the ISDN SETUP message that can be used as an authorization mechanism for incoming calls. The Tcl IVR 2.0 command set allows SIP terminating gateways to assign a specific value to the screening indicator through the use of Tcl scripts.

The screening indicator can contain four possible values:

  • User provided, not screened

  • User provided, verified and passed

  • User provided, verified and failed

  • Network provided

Note   In all scenarios, gateway accounting must be enabled, and all call-accounting information must be forwarded to the server that is performing preauthentication. Accounting stop packets must be sent to this server so that call billing is ended when calls are disconnected from the gateway. In addition, authentication and accounting start packets are needed to enable other features, such as virtual private dial-up network (VPDN).

> • For information on using Tcl IVR scripts to set and retrieve screening indicators, see the *Tcl IVR API Version 2.0 Programmer's Guide*

SIP Gateway HTTP Authentication Digest

The SIP: Gateway HTTP Authentication Digest feature implements authentication using the digest access on the client side of a common SIP stack. The gateway responds to authentication challenges from an authenticating server, proxy server, or user-agent server (UAS). This feature also maintains parity between the Cisco gateways, proxy servers, and SIP phones that already support authentication.

Feature benefits include the following:

> • A SIP gateway is able to respond to authentication challenges from authenticating proxy servers or user-agent servers (UASs). The authentication method supported is digest authentication. Although digest authentication is not the best method, it provides a basic level of security.

Note   The UAS challenges with a 401 response and the proxy server with a 407 response. It tries to find authentication credentials appropriate to the realm issuing the challenge and response. A gateway can handle authentication challenges from both the proxy server and UAS.

> • Registration of the destination patterns on POTS dial peers extends to all PSTN interfaces.

Note   The proxy server previously performed authentication only with the SIP phones.

The SIP Survivable Remote Site Telephony (SRST) feature in an earlier release added support to register E.164 numbers for foreign exchange stations (FXSs) (analog telephone voice ports) and extended foreign exchange stations (IP phone virtual voice ports) to an external SIP registrar. This feature extends that functionality for the gateway to register numbers configured on PSTN trunks such as PRI pipes.

Digest Access Authentication

SIP provides a stateless challenge-response mechanism for authentication based on digest access. A UAS or proxy server receiving a request challenges the initiator of the request to provide its identity. The user-agent client (UAC) generates a response by performing a message digest 5 (MD5) checksum on the challenge and its password. The response is passed back to the challenger in a subsequent request.

There are two modes of authentication:

> • Proxy-server authentication
>
> • UAS authentication

This feature also supports multiple proxy authentication on the gateway. The gateway can respond to up to five different authentication challenges in the signaling path between gateway as UAC and a UAS.

UAC-to-UAS Authentication

When the UAS receives a request without credentials from a UAC, it challenges the originator to provide credentials by rejecting the request with a "401 Unauthorized" response that includes a WWW-Authenticate header. The header field value consists of arguments applicable to digest scheme, as follows:

- realm--A string to be displayed to users so they know which username and password to use.

- nonce--A server-specified data string that should be uniquely generated each time a 401 response is made.

In addition, the header field may contain the following optional arguments:

- opaque--A string of data, specified by the server, that should be returned by the client unchanged in the Authentication header of subsequent requests with URIs in the same protection space.

- stale--A flag, indicating that the previous request from the client was rejected because the nonce value was stale.

- algorithm--A string indicating a pair of algorithms used to produce the digest and a checksum.

- qop-options--A string of one or more tokens indicating the "quality of protection" values supported by the server.

- auth-param--Directive that allows for future extensions.

The UAC reoriginates the request with proper credentials in the Authorization header field. The Authorization header field value consists of authentication information and arguments:

- username--User's name in specified realm. This value is taken from the configuration, either at the dial-peer or the global level.

- digest-uri--Same as request uri of the request.

- realm, nonce-- From WWW-Authenticate header.

Message digest 5 (MD5) is computed as follows:

```
MD5(concat(MD5(A1),(unquoted)nonce-value":"nc-value":"
(unquoted)cnonce-value ":"(unquoted)qop-value":"MD5(A2)))
```

where A1 = (unquoted) username-value ":" (unquoted) realm-value ":" password

```
A2 = Method ":" request-uri if qop is "auth"
& A2 = Method ":" request-uri ":" MD5(entity-body) if qop is "auth-int".
```

- The nc-value is the hexadecimal count of the number of requests (including the current request that the client has sent with the nonce value in this request.

- The cnonce-value is an opaque string provided by client for mutual authentication between client and server.

- The qop-value is quality of protection directive, "auth" or "auth-int".

UAC-to-UAS Call Flow with Register Message

In this call flow (see the figure below), the UA sends a Register message request without the Authorization header and receives a 401 status code message response challenge from the SIP server. The UA then resends the request including the proper credentials in the Authorization header.

Figure 2: UA-to-UAS Call Flow with Register Message
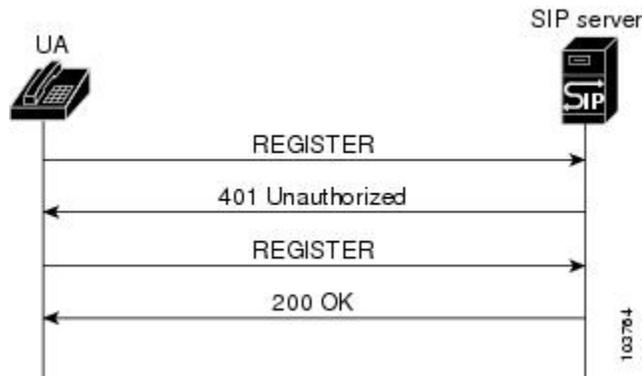
The UA sends a Register message request to the SIP server with the CSeq initialized to 1:

```
REGISTER sip:172.18.193.187:5060 SIP/2.0
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK200B
From: "36602" <sip:36602@172.18.193.120>;tag=98AS-87RT
To: <sip:36602@172.18.193.187>
Call-ID: A9EEC728-495E11D6-8003AD63-F55A9C4
User-Agent: Cisco-SIPGateway/IOS-12.x
CSeq: 1 REGISTER
Contact: <sip:36602@172.18.193.120:5060>;user=phone
Expires: 60
Content-Length: 0
```

The SIP server responds with a 401 Unauthorized challenge response to the UA:

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK200B
From: "36602" <sip:36602@172.18.193.120>;tag=98AS-87RT
To: <sip:36602@172.18.193.187>;tag=3046583040568302
Call-ID: A9EEC728-495E11D6-8003AD63-F55A9C4
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="example.com", qop="auth",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359", opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

The UA resends a Register message request to the SIP server that includes the authorization and increments the CSeq:

```
REGISTER sip:172.18.193.187:5060 SIP/2.0
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK1DEA
From: "36602" <sip:36602@172.18.193.120>;tag=98AS-89FD
To: <sip:36602@172.18.193.187>
Call-ID: A9EEC728-495E11D6-8003AD63-F55A9C4
User-Agent: Cisco-SIPGateway/IOS-12.x
Authorization: Digest username="36602", realm="example.com",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359", opaque="", uri="sip:172.18.193.187",
response="dfe56131d1958046689d83306477ecc"
CSeq: 2 REGISTER
```

```
Contact: <sip:36602@172.18.193.120:5060>;user=phone
Expires: 60
Content-Length: 0
```

The SIP server responds with a 200 OK message response to the UA:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 172.18.193.120:5060;received=172.18.193.120;branch=z9hG4bK1DEA
Call-ID: A9EEC728-495E11D6-8003AD63-F55A9C4
From: "36602" <sip:36602@172.18.193.120>;tag=98AS-89FD
To: <sip:36602@172.18.193.187>;tag=1q92461294
CSeq: 2 REGISTER
Contact: <sip:36602@172.18.193.120:5060>;expires="Wed, 02 Jul 2003 18:18:26 GMT"
Expires: 60
Content-Length: 0
```

Note   A SIP server can challenge any request except ACK and CANCEL request messages, because an ACK message request does not take any response and a CANCEL message request cannot be resubmitted. The UA uses the same credentials in an ACK message request as in an INVITE message request.

UAC-to-UAS Call Flow with INVITE Message

In this call flow (see the figure below), the UAC sends an INVITE message request to a UAS without proper credentials and is challenged with a 401 Unauthorized message response. A new INVITE message request is then sent, containing the correct credentials. Finally, the call is completed.

Figure 3: UAC-to-UAS Call Flow with INVITE Message
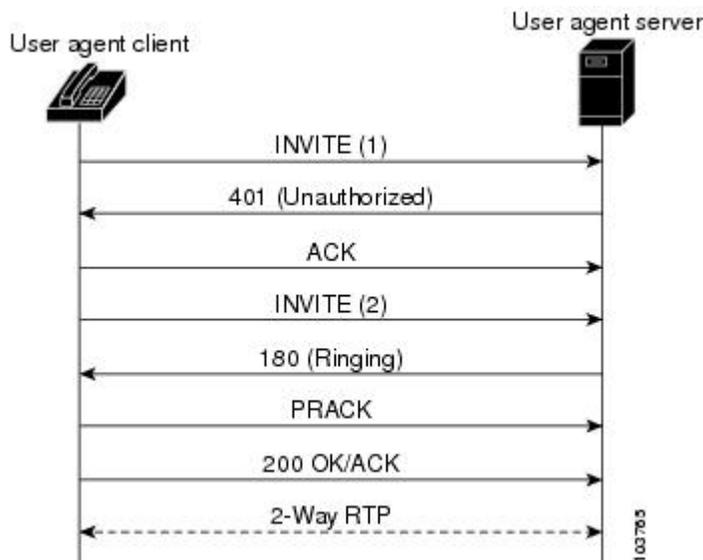
The UAS challenges the UAC to provide user credentials by issuing a 401 Unauthorized message response:

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK45TGN
From: "36602" <sip:36602@172.18.193.120>;tag=98AS-87RT
To: <sip:36602@172.18.193.187>;tag=3046583040568302
Call-ID: A9EEC728-495E11D6-8003AD63-F55A9C4
```

```
CSeq: 101 INVITE
WWW-Authenticate: Digest realm="example.com", qop="auth",
nonce="ea9c8e8809345gf1cec4341ae6cgh5a359", opaque=""
Content-Length: 0
```

The UAC resubmits the request with proper credentials in the Authorization header:

```
INVITE sip:36601@172.18.193.187:5060 SIP/2.0
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK8DF8H
From: "36602"<sip:36602@172.18.193.120>;tag=50EB48-383
To: <sip:36601@172.18.193.187>
Call-ID: A9EEC728-495E11D6-8003AD63-F55A9C4
CSeq: 102 INVITE
Authorization: Digest username="36602", realm="example.com",
nonce="ea9c8e8809345gf1cec4341ae6cgh5a359", opaque="", uri="sip:36601@172.18.193.187",
response="42ce3cef44b22f50c02350g6071bc8"
.
.
.
```

The UAC uses the same credentials in subsequent requests in that dialog:

```
PRACK sip:36601@172.18.193.187:5060 SIP/2.0
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK8YH5790
From: "36602"<sip:36602@172.18.193.120>;tag=50EB48-383
To: <sip:36601@172.18.193.187>;tag=AG09-92315
Call-ID: A9EEC728-495E11D6-8003AD63-F55A9C4
CSeq: 103 PRACK
Authorization: Digest username="36602", realm="example.com",
nonce="ea9c8e8809345gf1cec4341ae6cgh5a359", opaque="", uri="sip:36601@172.18.193.187",
response="42ce3cef44b22f50c02350g6071bc9"
Content-Length: 0
```

Proxy-Server-to-UA Authentication

When a UA submits a request to a proxy server without proper credentials, the proxy server authenticates the originator by rejecting the request with a 407 message response (Proxy Authentication Required) and includes a Proxy-Authenticate header field value applicable to the proxy server for the requested resource. The UAC follows the same procedure mentioned in "UAC-to-UAS_Authentication" to get proper credentials for the realm and resubmits the request with the credentials in the Proxy-Authorization header.

Note   realm--A string to be displayed to users so they know which username and password to use.

Proxy Server to UA Authentication Call Flow

In this call flow the UAC completes a call to user a UAS by using two proxy servers (PS 1 or PS 2, (see the figure below). The UAC has valid credentials in both domains. Because the initial INVITE message request does not contain the Authorization credentials proxy server 1 requires, a 407 Proxy Authorization message response containing the challenge information is sent. A new INVITE message request containing the correct credentials is then sent and the call proceeds after proxy server 2 challenges and receives valid credentials.

Figure 4: Proxy-Server-to-UA Call Flow
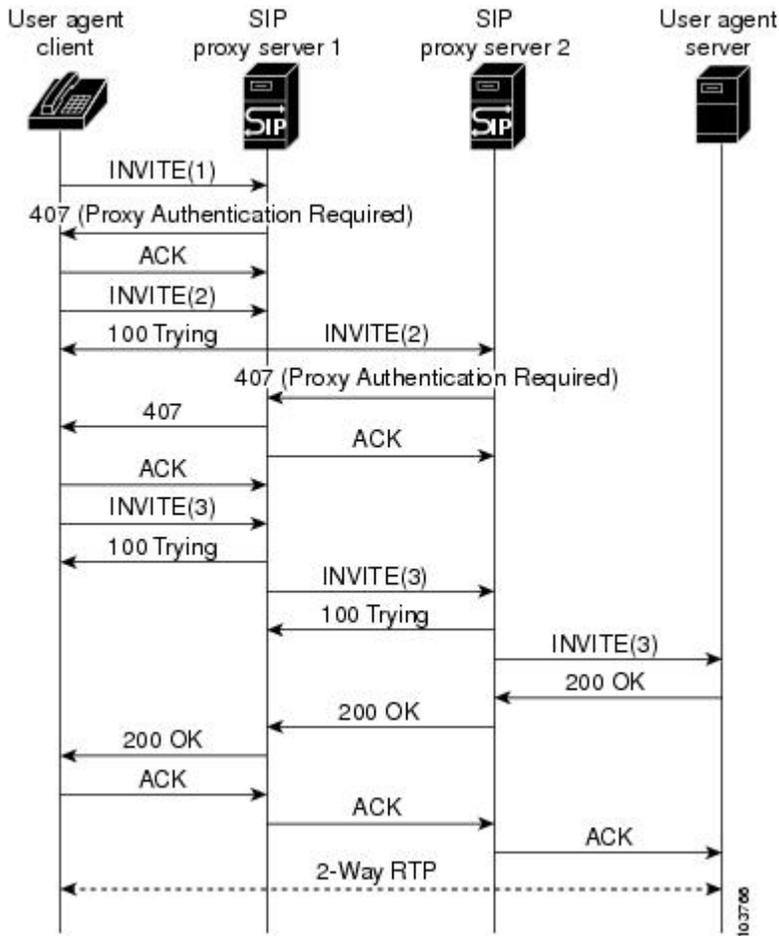
Proxy server 1 challenges the UAC for authentication:

```
SIP/2.0 407 Proxy Authorization Required
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK207H
From: <sip:36602@172.18.193.120>;tag=50EB48-383
To: <sip:36601@172.18.193.187>;tag=929523858000835
Call-ID: D61E40D3-496A11D6-80070030-9426ED30@172.18.193.120
CSeq: 101 INVITE
Proxy-Authenticate: Digest realm="proxy1.example.com", qop="auth",
nonce="wf84f1cczx41ae6cbeaea9ce88d359", opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

The UAC responds by resending the INVITE message request with authentication credentials. The same Call-ID is used, so the CSeq is increased.

```
INVITE sip:36601@172.18.193.187:5060 SIP/2.0
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bKEE1
From: <sip:36602@172.18.193.120>;tag=50EB48-383
To: <sip:36601@172.18.193.187>
Call-ID: D61E40D3-496A11D6-80070030-9426ED30@172.18.193.120
CSeq: 102 INVITE
Proxy-Authorization: Digest username="36602", realm="proxy1.example.com",
nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="", uri="sip:36601@172.18.193.187",
```

```
response="42ce3cef44b22f50c6a6071bc8"
Contact: <sip:172.18.193.120:5060>
.
.
.
```

The proxy server 2 challenges the UAC INVITE message request for authentication which is the 407 authentication message response that is forwarded to the UAC by proxy server 1.

```
SIP/2.0 407 Proxy Authorization Required
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bKEE1
From: <sip:36602@172.18.193.120>;tag=50EB48-383
To: <sip:36601@172.18.193.187>;tag=083250982545745
Call-ID: D61E40D3-496A11D6-80070030-9426ED30@172.18.193.120
Proxy-Authenticate: Digest realm="proxy2.example.com", qop="auth",
nonce="c1e22c41ae6cbe5ae983a9c8e88d359", opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

The UAC responds by resending the INVITE message request with authentication credentials for proxy server 1 and proxy server 2.

```
INVITE sip:36601@172.18.193.187:5060 SIP/2.0
Via: SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK8GY
From: <sip:36602@172.18.193.120>;tag=50EB48-383
To: <sip:36601@172.18.193.187>
Call-ID: D61E40D3-496A11D6-80070030-9426ED30@172.18.193.120
CSeq: 103 INVITE
Proxy-Authorization: Digest username="36602", realm="proxy1.example.com",
nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="", uri="sip:36601@172.18.193.187",
response="42ce3cef44b22f50c6a6071bc8"
Proxy-Authorization: Digest username="36602", realm="proxy2.example.com",
nonce="c1e22c41ae6cbe5ae983a9c8e88d359", opaque="", uri="sip:36601@172.18.193.187",
response="f44ab22f150c6a56071bce8"
.
.
.
```

Extending SIP Register Support on Gateway

The SIP: Gateway HTTP Authentication Digest feature enhances functionality for Cisco IOS SIP gateway to Register all addresses specified by destination patterns in operational POTS dial-peers for all ports. This provides customer flexibility to register and authenticate users behind a private branch exchange (PBX) connected to the gateway through a PRI interface. There is no change in the way the gateway with foreign-exchange-station (FXS) ports registers individual E.164 addresses.

This feature leverages dial peers to create granularity for registration and authentication. However, the dial peers can be created with wildcards (for example: .919T , where terminator [T] makes the gateway wait until the full dial-string is received.) and a range of numbers (for example: .919392... , where ... indicates numbers in the range 0000 to 9999). Such destination patterns are registered with a single character wildcard in the user portion of To and Contact headers. The table below shows how the various types of gateway dial plans map to its registration. You need to modify the proxy/registrar behavior to correctly route calls for wildcard patterns or destination pattern with a range. Proxy server or registrars that do not match a wildcard patterns or destination pattern with a range should be ignored for that specific request.

Table 1: SIP Cisco IOS Gateway Dial Peer Mapping to Register

Cisco IOS SIP GW Configuration	Corresponding Register
dial-peer voice 919 pots destination-pattern 919....... port 0:D	REGISTER sip:proxy.example.com SIP/2.0 To: <sip:919.......@172.18.193.120> From: <sip:172.18.192.120>;tag=ABCD Contact: <sip:919.......@172.18.193.120>;user=phone
dial-peer voice 555 pots destination-pattern 555T port 0:D	REGISTER sip:proxy.example.com SIP/2.0 To: <sip:555*@172.18.193.120> From: <sip:172.18.192.120>;tag=ABCD Contact: <sip:555*@172.18.193.120>;user=phone
dial-peer voice 5550100 pots destination-pattern 5550100 port 0:D	REGISTER sip:proxy.example.com SIP/2.0 To: <sip:5550100@172.18.193.120> From: <sip:5550100@172.18.192.120>;tag=ABCD Contact: <sip:5550100@172.18.193.120>;user=phone

How to Configure SIP AAA Features

Configuring RADIUS Pre-authentication for Voice Calls

Configure a RADIUS Group Server

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa group server radius** *groupname*
5. **server** *ip-address* [**auth-port** *port*] [**acct-port** *port*]
6. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** Router(config)# aaa new-model	Enables the authentication, authorization, and accounting access-control model.
Step 4	**aaa group server radius** *groupname* **Example:** Router(config-sg-radius)# aaa group server radius radgroup1	(Optional) Groups different RADIUS server hosts into distinct lists and distinct methods. The argument is as follows: • groupname--Character string used to name the group of servers.
Step 5	**server** *ip-address* [**auth-port** *port*] [**acct-port** *port*] **Example:** Router(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001	(Required if the aaa group server command is used) Configures the IP address of the RADIUS server for the group server. Keywords and arguments are as follows: • ip-address--IP address of the RADIUS server host. • auth-post port-number--UDP destination port for authentication requests. The host is not used for authentication if this value is set to 0. Default: 1645. • acct-port port-number--UDP destination port for accounting requests. The host is not used for accounting services if this value is set to 0. Default: 1646.
Step 6	**exit** **Example:** Router(config-sg-radius)# exit	Exits the current mode.

Configure Access and Authentication

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa authentication login h323 group** *groupname*
4. **aaa authentication ppp default group** *groupname*
5. **aaa authorization exec** *list-name* **group** *groupname*
6. **aaa authorization network default group** {**radius** | **rpms**} **if-authenticated**
7. **aaa authorization reverse-access default local**

8. **aaa accounting suppress null-user-name**
9. **aaa accounting send stop-record authentication failure**
10. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	**aaa authentication login h323 group** *groupname* **Example:** Router(config)# aaa authentication billson h323 group 123	Sets authentication, authorization, and accounting at login. Keywords and arguments are as follows: • h323--Use H.323 for authentication. • group groupname--Use a subset of RADIUS or TACACS+ servers for authentication as defined by the aaa group server radius or aaa group server tacacs+ command.
Step 4	**aaa authentication ppp default group** *groupname* **Example:** Router(config)# aaa authentication ppp default group 123	(Required for PPP dial-in methods that are to be used with preauthentication) Specifies one or more authentication, authorization, and accounting authentication methods for use on serial interfaces running PPP. Keywords and arguments are as follows: • default--Use the listed authentication methods that follow this argument as the default list of methods when a user logs in. • group groupname--Use a subset of RADIUS or TACACS+ servers for authentication as defined by the aaa group server radius or aaa group server tacacs+ command.
Step 5	**aaa authorization exec** *list-name* **group** *groupname* **Example:** Router(config)# aaa authorization exec billson group 123	(Optional) Sets parameters that restrict user access to a network. Keywords and arguments are as follows: • exec--Run authorization to determine if the user is allowed to run an EXEC shell. This facility might return user profile information such as autocommand information. • list-name--Character string used to name the list of authorization methods.

Command or Action	Purpose
	• group groupname--Use a subset of RADIUS or TACACS+ servers for authentication as defined by the aaa group server radius or aaa group server tacacs+ command.

Step 6   **aaa authorization network default group** {**radius** | **rpms**} **if-authenticated**
 	
Example:
 	
`Router(config)# aaa authorization network default`
`group radius if-authenticated`
	(Optional) Sets parameters that restrict user access to a network. Keywords are as follows:
	
	• network--Run authorization for all network-related service requests, including Serial Line Internet Protocol, Point-to-Point Protocol, PPP network Control Programs, and Apple Talk Remote Access.
	
	• default--Use the listed authorization methods that follow this argument as the default list of methods for authorization.
	
	• group radius--Use a subset of RADIUS or TACACS+ servers for authentication as defined by the aaa group server radius or aaa group server tacacs+ command.
	
	• group rpms--Use a subset of RADIUS or TACACS+ servers for authentication as defined by the aaa group server radius or aaa group server tacacs+ command.
	
	• if-authenticated--Allow the user to access the requested function if the user is authenticated.

Step 7   **aaa authorization reverse-access default local**
 	
Example:
 	
`Router(config)# aaa authorization reverse-access`
`default local`
	(Optional) Configures a network access server to request authorization information from a security server before allowing a user to establish a reverse Telnet session. Keywords are as follows:
	
	• default--Use the listed authorization methods that follow this argument as the default list of methods for authorization.
	
	• local--Use the local database for authorization.

Step 8   **aaa accounting suppress null-user-name**
 	
Example:
 	
`Router(config)# aaa accounting suppress`
`null-username`
	(Optional) Prevents the Cisco IOS software from sending accounting records for users whose username string is NULL.

Step 9   **aaa accounting send stop-record authentication failure**
 	
Example:
 	
`Router(config)# aaa accounting send stop-record`
`authentication failure`
	(Required if using Cisco RPMS) Generates account "stop" records for users who fail to authenticate at login or during session negotiation.

	Command or Action	Purpose
Step 10	exit **Example:** `Router(config)# exit`	Exits the current mode.

Configure Accounting

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting delay-start**
4. **aaa accounting update** [**periodic** *number*]
5. **aaa accounting exec default start-stop group** *groupname*
6. **aaa accounting exec** *list-name* **start-stop group** *groupname*
7. **aaa accounting network default start-stop group** *groupname*
8. **aaa accounting connection h323 start-stop group** *groupname*
9. **aaa accounting system default start-stop group** *groupname*
10. **aaa accounting resource default start-stop-failure group** *groupname*
11. **gw-accounting aaa**
12. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	aaa accounting delay-start **Example:** `Router(config)# aaa accounting delay-start`	(Optional) Delays generation of accounting "start" records until the user IP address is established. For a complete explanation of the aaa accounting command, see the *Cisco IOS Security Command Reference*
Step 4	aaa accounting update [**periodic** *number*] **Example:** `Router(config)# aaa accounting update periodic 30`	(Optional) Enables periodic interim accounting records to be sent to the accounting server. Keyword and argument are as follows:

	Command or Action	Purpose
		• periodic number--An interim accounting record is sent to the accounting server periodically, as defined by the argument number (in minutes).
Step 5	**aaa accounting exec default start-stop group** *groupname* **Example:** Router(config)# aaa accounting exec default start-stop group joe	(Optional) Enables authentication, authorization, and accounting of requested services for billing or security purposes when you use RADIUS or TACACS+ and want to run a shell session. Keywords and arguments are as follows: • **exec** --Run accounting for EXEC shell session. This keyword might return profile information such as what is generated by the autocommand command. • **default** --Use the listed accounting methods that follow this argument as the default list of methods for accounting services. • **start-stop** --Send a "start" accounting notice at the beginning of a process and a "stop" accounting notice at the end of a process. The "start" accounting record is sent in the background. The requested user process begins regardless of whether the "start" accounting notice was received by the accounting server. • **group** *groupname* --Use a subset of RADIUS or TACACS+ servers for accounting as defined by the server group groupname.
Step 6	**aaa accounting exec** *list-name* **start-stop group** *groupname* **Example:** Router(config)# aaa accounting exec joe start-stop group tacacs+	(Optional) Enables authentication, authorization, and accounting of requested services for billing or security purposes when you use RADIUS or TACACS+ and want to specify method names. Keywords and arguments are as follows: • **exec** --Run accounting for EXEC shell session. This keyword might return profile information such as what is generated by the autocommand command. • *list-name* --Character string used to name the list of at least one of the accounting methods. • **start-stop** --Send a "start" accounting notice at the beginning of a process and a "stop" accounting notice at the end of a process. The "start" accounting record is sent in the background. The requested user process begins regardless of whether the "start" accounting notice was received by the accounting server. • **group** *groupname* --Use a subset of RADIUS or TACACS+ servers for accounting as defined by the server group groupname.

	Command or Action	Purpose

Step 7	**aaa accounting network default start-stop group** *groupname*
	
	Example:
	
	`Router(config)# aaa accounting network default start-stop group tacacs+`
		(Required for PPP dial-in methods that are to be used for preauthentication) Enables authentication, authorization, and accounting of requested network services for billing and security purposes when you use RADIUS or TACACS+. Keywords and arguments are as follows:
		
		• **network** --Run accounting for all network-related service requests, including Serial Line Internet Protocol, Point-to-Point Protocol, PPP Network Control Protocols, and Apple Talk Remote Access Protocol.
		
		• **default** --Use the listed accounting methods that follow this argument as the default list of methods for accounting services.
		
		• **start-stop** --Send a "start" accounting notice at the beginning of a process and a "stop" accounting notice at the end of a process. The "start" accounting record is sent in the background. The requested user process begins regardless of whether the "start" accounting notice was received by the accounting service.
		
		• **group** *groupname* --At least one of the following:
		
		  • **group radius**--Use the list of all RADIUS servers for authentication as defined by the **aaa group server radius** command.
		  • **group-tacacs+**--Use the list of all TACACS+ servers for authentication as defined by the **aaa group server tacacs+** command.
		  • **group** *groupname*--Use a subset of RADIUS or TACACS+ servers for accounting as defined by the server group groupname.

Step 8	**aaa accounting connection h323 start-stop group** *groupname* **Example:** `Router(config)# aaa accounting connection h323 start-stop group tacacs+`	(Required for voice call accounting) Enables accounting, accounting of requested services for billing and security purposes when you use RADIUS or TACACS+ and want connection information. Keywords and arguments are as follows: • connection--Provide information about all outbound connections made from the network access server, such as Telnet, LAT, TN3270, PAD, and rlogin. • h323--Character string used to name the list of at least one of the accounting methods. Uses h323 for accounting. • start-stop--Send a "start" accounting notice at the beginning of a process and a "stop" accounting notice at the end of a process. The "start" accounting record

Command or Action	Purpose

	is sent in the background. The requested user process begins regardless of whether the "start" accounting notice was received by the accounting service.
	
	• group groupname--At least one of the following:
	
	    • group radius--Use the list of all RADIUS servers for authentication as defined by the aaa group server radius command.
	    • group-tacacs+--Use the list of all TACACS+ servers for authentication as defined by the aaa group server tacacs+ command.
	    • group groupname--Use a subset of RADIUS or TACACS+ servers for accounting as defined by the server group groupname.

Step 9   **aaa accounting system default start-stop group** *groupname*
 	
Example:
 	
`Router(config)# aaa accounting system default`
`start-stop group tacacs+`
	(Optional) Enables accounting, accounting of requested services for billing and security purposes you when use RADIUS or TACACS+ and want system-level event accounting. Keywords and arguments are as follows:
	
	• system--Perform accounting for all system-level events not associated with users, such as reloads.
	
	• default--Use the listed accounting methods that follow this argument as the default list of methods for accounting services.
	
	• start-stop--Send a "start" accounting notice at the beginning of a process and a "stop" accounting notice at the end of a process. The "start" accounting record is sent in the background. The requested user process begins regardless of whether the "start" accounting notice was received by the accounting service.
	
	• group groupname--At least one of the following:
	
	    • group radius--Use the list of all RADIUS servers for authentication as defined by the aaa group server radius command.
	    • group-tacacs+--Use the list of all TACACS+ servers for authentication as defined by the aaa group server tacacs+ command.
	    • group groupname--Use a subset of RADIUS or TACACS+ servers for accounting as defined by the server group groupname.

Step 10 **aaa accounting resource default start-stop-failure group** *groupname* **Example:**	(Optional) Enables full resource accounting, which will generate both a "start" record at call setup and a "stop" record at call termination. Keywords and arguments are as follows:

Command or Action	Purpose

`Router(config)# aaa accounting resource default start-stop-failure group tacacs+`	• default--Use the listed accounting methods that follow this argument as the default list of methods for accounting services.
	
	 • group groupname--Server group to be used for accounting services.Valid values are as follows:
	
	    • string--Character string used to name a server group.
	    • radius--Use list of all RADIUS hosts.
	    • tacacs+--Use list of all TACACS+ hosts.

Step 11 **gw-accounting aaa** **Example:** `Router(config)# gw-accounting aaa`	Enables VoIP gateway-specific accounting and define the accounting method.
Step 12 **exit** **Example:** `Router(config)# exit`	Exits the current mode.

Configure Preauthentication

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa preauth**
4. **group** {**radius** | *groupname*}
5. **clid** [**if-avail** | **required**] [**accept-stop**] [**password** *string*]
6. **ctype** [**if-avail** | **required**] [**accept-stop**] [**password** *string*]
7. **dnis** [**if-avail** | **required**] [**accept-stop**] [**password** *string*]
8. **dnis bypass** {*dnis-groupname*}
9. **filter voice**
10. **timeout leg3** *time*
11. **service-type call-check**
12. **exit**

DETAILED STEPS

Command or Action	Purpose
Step 1 **enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	**aaa preauth** **Example:** Router(config)# aaa preauth	Enters AAA preauthentication configuration mode.
Step 4	**group** {**radius** | *groupname*} **Example:** Router(config-preauth)# group radius	Specifies the authentication, authorization, and accounting RADIUS server group to use for preauthentication. Keywords and arguments are as follows: • radius--Use a RADIUS server for authentication. • groupname--Name of the server group to use for authentication.
Step 5	**clid** [**if-avail** | **required**] [**accept-stop**] [**password** *string*] **Example:** Router(config-preauth)# clid required	(Optional) Preauthenticates calls based on the Calling Line Identification (CLID) number. Keywords and arguments are as follows: • if-avail--If the switch provides the data, RADIUS must be reachable and must accept the string in order for preauthentication to pass. If the switch does not provide the data, preauthentication passes. • required--The switch must provide the associated data, that RADIUS must be reachable, and that RADIUS must accept the string in order for preauthentication to pass. If these three conditions are not met, preauthentication fails. • accept-stop--Prevents subsequent preauthentication elements such as ctype or dnis from being tried once preauthentication has succeeded for a call element. • password string--Defines the password for the preauthentication element.
Step 6	**ctype** [**if-avail** | **required**] [**accept-stop**] [**password** *string*] **Example:** Router(config-preauth)# ctype required	(Optional) Preauthenticates calls on the basis of the call type. Keywords and arguments are as described above.

	Command or Action	Purpose
Step 7	**dnis** [**if-avail** | **required**] [**accept-stop**] [**password** *string*] **Example:** `Router(config-preauth)# dnis required`	(Optional) Preauthenticates calls on the basis of the Dialed Number Identification Server (DNIS) number. Keywords and arguments are as described above.
Step 8	**dnis bypass** {*dnis-groupname*} **Example:** `Router(config-preauth)# dnis bypass abc123`	(Optional) Specifies a group of DNIS numbers that will be bypassed for preauthentication. The argument is as follows: • *dnis-groupname* --Name of the defined DNIS group.
Step 9	**filter voice** **Example:** `Router(config-preauth)# filter voice`	(Optional) Specifies that voice calls bypass authentication, authorization, and account preauthentication.
Step 10	**timeout leg3** *time* **Example:** `Router(config-preauth)# timeout leg3 100`	(Optional) Sets the timeout value for a leg 3 AAA preauthentication request. The argument is as follows: • *time* --Timeout value for leg3 preauthentication, in ms. Range: 100 to 1000. Default: 100.
Step 11	**service-type call-check** **Example:** `Router(config-preauth)# service-type call-check`	(Optional) Identifies preauthentication requests to the AAA server.
Step 12	**exit** **Example:** `Router(config-preauth)# exit`	Exits the current mode.

Configure RADIUS Communications

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server host** {*hostname* \| *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] [**alias** {*hostname* \| *ip-address*}]
4. **radius-server retransmit** *retries*
5. **radius-server attribute 6 support-multiple**
6. **radius-server attribute 44 include-in-access-req**
7. **radius-server attribute nas-port format c**
8. **radius-server key** {**0** *string* \| **7** *string* \| *string*}
9. **radius-server vsa send accounting**
10. **radius-server vsa send authentication**

    11.   **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] [**alias** {*hostname* | *ip-address*}] **Example:** `radius-server host jimname`	Specifies a RADIUS server host. Keywords and arguments are as follows: • *hostname* --DNS name of the RADIUS server host. • *ip-address* --IP address of the RADIUS server host. • **auth-port** *port-number* --UDP destination port for authentication requests; the host is not used for authentication if set to 0. Default: 1645. • **acct-port** *port-number* --UDP destination port for accounting requests; the host is not used for accounting if set to 0. Default: 1646. • **timeout** --Time interval (in seconds) that the router waits for the RADIUS server to reply before retransmitting. This setting overrides the global value of the radius-server timeout command. Range: 1 to 1000. If no timeout value is specified, the global value is used. • **retransmit** *retries* --Number of times that a RADIUS request is resent to a server, if that server is not responding or responding slowly. This setting overrides the global setting of the radius-server retransmit command. Range: 1 to 100. If no retransmit value is specified, the global value is used. • **key** *string* --Authentication and encryption key used between the router and the RADIUS daemon running on this RADIUS server. This key overrides the global setting of the radius-server key command. If no key string is specified, the global value is used. The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the radius-server host command syntax.

	Command or Action	Purpose
		This is because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks themselves are part of the key. • alias--Allow up to eight aliases per line for any given RADIUS server.
Step 4	**radius-server retransmit** *retries* **Example:** `Router(config)# radius-server retransmit 1`	(Optional) Specifies the number of times the Cisco IOS software searches the list of RADIUS server hosts before giving up. The argument is as follows: • *retries* --Maximum number of retransmission attempts. Default: 3.

Step 5	**radius-server attribute 6 support-multiple**
	
	Example:
	
	`Router(config)# radius-server attribute 6`
	`support-multiple`
		(Optional) Sets an option for RADIUS Attribute 6 (Service-Type) values in a RADIUS profile. The keyword is as follows:
		
		• **support-multiple** --Support multiple service-type values in each RADIUS profile.

Step 6	**radius-server attribute 44 include-in-access-req**
	
	Example:
	
	`Router(config)# radius-server attribute 44`
	`include-in-access-req`
		Sends RADIUS attribute 44 (Accounting Session ID) in access request packets before user authentication (including requests for preauthentication).
		
		Note For information on RADIUS attributes, see the *Cisco IOS Security Command Reference*.

Step 7	**radius-server attribute nas-port format c**
	
	Example:
	
	`Router(config)# radius-server attribute nas-port`
	`format c`
		(Required if using Cisco RPMS) Selects the NAS-Port format used for RADIUS accounting features.

Step 8	**radius-server key** {**0** *string* | **7** *string* | *string*}
	
	Example:
	
	`Router(config)# radius-server key`
	`ncmmekweisnaowkakskiiw`
		(Optional) Sets the authentication and encryption key for all RADIUS communications between the router and the RADIUS daemon. Keywords and arguments are as follows:
		
		• 0 string--An unencrypted (cleartext) shared key as specified by string.
		
		• 7 string--A hidden shared key as specified by string.
		
		• string--The unencrypted (cleartext) shared key.

Step 9	**radius-server vsa send accounting** **Example:** `Router(config)# radius-server vsa send accounting`	(Optional) Configures the network access server to recognize and use vendor-specific attributes.

	Command or Action	**Purpose**
Step 10	**radius-server vsa send authentication** **Example:** `Router(config)# radius-server vsa send authentication`	(Optional) Configures the network access server to recognize and use vendor-specific attributes.
Step 11	**exit** **Example:** `Router(config)# exit`	Exits the current mode.

Configuring SIP - Enhanced Billing Support for Gateways

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **sip-ua**
4. **aaa username {calling-name | proxy-auth}**
5. **exit**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**sip-ua** **Example:** `Router(config)# sip-ua`	Enters SIP user-agent configuration mode.
Step 4	**aaa username {calling-name | proxy-auth}** **Example:** `Router(config-sip-ua)# aaa username calling-name`	Determines the information to populate the username attribute for AAA billing records. Keywords are as follows: • **calling-number** --Use the FROM: header in the SIP INVITE (default value). This keyword is used in most implementations and is the default.

Command or Action	Purpose
	• **proxy-auth** --Parse the Proxy-Authorization header. Decode the Microsoft Passport user ID (PUID) and password, and then populate the PUID into the username attribute and a "." into the password attribute. The username attribute is used for billing and the "." is used for the password, because the user has already been authenticated.
Step 5	**exit** **Example:** Router(config-sip-ua)# exit	Exits the current mode.

Configuring SIP Gateway HTTP Authentication Digest

Configure SIP Gateway HTTP Authentication Digest Via Dial-Peer

> **Note** This configuration sets up the feature as defined under the POTS dial peer.

• This feature is configured at the POTS dial peer and SIP user agent, with configuration at the dial peer taking precedence over that at the SIP user agent.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dial-peer voice** *tag* **pots**
4. **authentication username** *username* **password** *password* [**realm** *realm*]
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**dial-peer voice** *tag* **pots** **Example:** Router(config)# dial-peer voice 100 pots	Enters dial-peer configuration mode for the specified POTS dial peer.
Step 4	**authentication username** *username* **password** *password* [**realm** *realm*] **Example:** Router(config-sip-ua)# authentication username user1 password password1 realm example.com	Enters SIP digest authentication mode. Keywords and arguments are as follows: • username username--A string representing username of the user authenticating. • password password--A string representing password for authentication. • realm realm--A string representing the applicable credential.
Step 5	**exit** **Example:** Router(config-sip-ua)# exit	Exits the current mode.

Configure SIP Gateway HTTP Authentication Digest Via SIP UA

Note You can configure this feature for a dial peer or globally, for all POTS dial peers, in SIP user-agent configuration mode. If authentication is configured in SIP user-agent configuration mode and on individual dial peers, the individual dial-peer configuration takes precedence.

Note • SIP Register is supported only on platforms with digital trunk type ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **sip-ua**
4. **registrar** {**dns:** *address* | **ipv4:** *destination-address*} **expires** *seconds* [**tcp**] [**secondary**]
5. **authentication username** *username* **password** *password* [**realm** *realm*]
6. **exit**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**sip-ua** **Example:** `Router(config)# sip-ua`	Enters SIP user-agent configuration mode.
Step 4	**registrar** {**dns:** *address* | **ipv4:** *destination-address*} **expires** *seconds* [**tcp**] [**secondary**] **Example:** `Router(config-sip-ua)# registrar ipv4:10.1.1.6 expires 60`	Registers E.164 numbers on behalf of analog telephone voice ports (FXS) and IP phone virtual voice ports (EFXS) with an external SIP proxy or SIP registrar server. Keywords and arguments are as follows: • **dns:** *address* --Domain name server that resolves the name of the dial peer to receive calls. • **ipv4:** *destination address* --IP address of the dial peer to receive calls. • **expires** *seconds* --Default registration time, in seconds. • **tcp** --Transport layer protocol is TCP. UDP is the default. • **secondary** --Registration is with a secondary SIP proxy or registrar for redundancy purposes. **Note** When registrar is provisioned, the gateway sends out register with 1.. .
Step 5	**authentication username** *username* **password** *password* [**realm** *realm*] **Example:** `Router(config-sip-ua)# authentication username user1 password password1 realm example.com`	Enters SIP digest authentication mode. Keywords and arguments are as follows: • **username** *username* --A string representing username of the user authenticating. • **password** *password* --A string representing password for authentication. • **realm** *realm* --A string representing the applicable credential.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Router(config-sip-ua)# exit`	Exits the current mode.

Verifying AAA Features for SIP

To verify AAA-feature configuration, perform the following steps as appropriate (commands are listed in alphabetical order).

SUMMARY STEPS

1. **show call active voice**
2. **show radius statistics**
3. **show rpms-proc counters**
4. **show running-config**
5. **show sip-ua register status**

DETAILED STEPS

Step 1   **show call active voice**

Use this command to display call information for active voice calls. You can thus verify the username attribute.

The following sample output shows that the proxy-auth parameter is selected.

Example:

```
Router# show call active voice
Total call-legs: 2
  GENERIC:
 SetupTime=1551144 ms
   .
   . (snip)
   .
 ReceiveBytes=63006
 VOIP:
 ConnectionId[0x220A95B7 0x6B3611D5 0x801DBD53 0x8F65BA34]
   .
   . (snip)
   .
 CallerName=
 CallerIDBlocked=False
 Username=1234567890123456            <-- PUID from Proxy-Auth header
```

The following sample output shows that the calling-number parameter is selected.

Example:

```
Router# show call active voice
Total call-legs: 2
  GENERIC:
```

```
SetupTime=1587000 ms
 .
 . (snip)
 .
ReceiveBytes=22762
VOIP:
ConnectionId[0xF7C22E07 0x6B3611D5 0x8022BD53 0x8F65BA34]
 .
 . (snip)
 .
CallerName=
CallerIDBlocked=False
Username=1234                          <-- calling-number
```

Step 2   **show radius statistics**

Use this command to display RADIUS statistics for accounting and authentication packets.

Step 3   **show rpms-proc counters**

Use this command to display the number of leg 3 preauthentication requests, successes, and rejects.

Note   Use the **clear rpms-proc counters** command to reset the counters that record the statistics that the **show rpms-proc counters** command displays.

Step 4   **show running-config**

Use this command to display the current configuration.

Step 5   **show sip-ua register status**

Use this command to verify SIP user-agent register status.

Example:

```
Router# show sip-ua register status
Line  peer    expires(sec)  registered
4001  20001   596           no
4002  20002   596           no
5100  1       596           no
9998  2       596           no
where:
line=phone number to register
peer=registration destination number
expires (sec)=amount of time, in seconds, until registration expires
registered=registration status
```

Troubleshooting Tips

Note   For general troubleshooting tips and a list of important **debug** commands, see the "General Troubleshooting Tips" section.

- Make sure that you can make a voice call.

- If the gateway does not respond to the authentication challenge, make sure that the user credentials for the appropriate domain have been configured.

- For the gateway to register destination patterns on the POTS dial peer, make sure that a registrar has been configured.

- Use the **debug aaa authentication** command to display high-level diagnostics related to AAA logins.

- Use the **debug cch323 preauth** command to enable debug tracing on the H.323 SPI for preauthentication.

- Use the **debug ccsip**family of commands to enable SIP debugging capabilities. In particular, use the following:

    - Use the **debug ccsip all** and **debug ccsip events**commands to display output specific to the SIP - Enhanced Billing Support for Gateways feature.
    - Use the **debug ccsip preauth** command to enable debug tracing on the SIP service provider interface (SPI) for preauthentication.

- Use the **debug radius** command to enable debug tracing of RADIUS attributes.

- Use the **debug rpms-proc preauth** command to enable debug tracing on the RPMS process for H.323 calls, SIP calls, or both H.323 and SIP calls.

Following is sample output for some of these commands:

Sample Output for the debug ccsip Command

```
Router# debug ccsip messages
*Oct 11 21:40:26.175://-1/xxxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Sent:
INVITE sip:5550123@172.18.193.187:5060 SIP/2.0 ! Invite request message (command sequence
101)
Via:SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK6ED
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>
Date:Fri, 11 Oct 2002 21:40:26 GMT
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
Supported:100rel,timer
Min-SE: 1800
Cisco-Guid:3787171507-3700953558-2147913662-199702180
User-Agent:Cisco-SIPGateway/IOS-12.x
Allow:INVITE, OPTIONS, BYE, CANCEL, ACK, PRACK, COMET, REFER, SUBSCRIBE, NOTIFY, INFO,
UPDATE, REGISTER
CSeq:101 INVITE
Max-Forwards:70
Remote-Party-ID:"36602" <sip:36602@172.18.193.120>;party=calling;screen=no;privacy=off
Timestamp:1034372426
Contact:<sip:36602@172.18.193.120:5060>
Expires:180
Allow-Events:telephone-event
Content-Type:application/sdp
Content-Length:244
v=0
o=CiscoSystemsSIP-GW-UserAgent 6603 1568 IN IP4 172.18.193.120
s=SIP Call
c=IN IP4 172.18.193.120
t=0 0
m=audio 17978 RTP/AVP 18 19
c=IN IP4 172.18.193.120
a=rtpmap:18 G729/8000
```

```
a=fmtp:18 annexb=no
a=rtpmap:19 CN/8000
a=ptime:20
*Oct 11 21:40:26.179://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Received:
SIP/2.0 100 Trying ! 100 Trying response message (command sequence 101)
Via:SIP/2.0/UDP 172.18.193.120:5060;received=172.18.193.120;branch=z9hG4bK6ED
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>
CSeq:101 INVITE
Content-Length:0
*Oct 11 21:40:26.179://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Received:
SIP/2.0 407 Proxy Authentication Required ! 407 proxy authentication required response
message (command sequence 101)
Via:SIP/2.0/UDP 172.18.193.120:5060;received=172.18.193.120;branch=z9hG4bK6ED
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>;tag=214b-70c4
CSeq:101 INVITE
Proxy-Authenticate:DIGEST realm="example.com", nonce="405729fe", qop="auth", algorithm=MD5
Content-Length:0
*Oct 11 21:40:26.183://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Sent:
ACK sip:5550123@172.18.193.187:5060 SIP/2.0 ! ACK request message (command sequence 101)
Via:SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK6ED
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>;tag=214b-70c4
Date:Fri, 11 Oct 2002 21:40:26 GMT
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
Max-Forwards:70
CSeq:101 ACK
Content-Length:0
*Oct 11 21:40:26.183://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Sent:
INVITE sip:5550123@172.18.193.187:5060 SIP/2.0 ! Invite message request (command sequence
102)
Via:SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK8BA
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>
Date:Fri, 11 Oct 2002 21:40:26 GMT
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
Supported:100rel,timer
Min-SE: 1800
Cisco-Guid:3787171507-3700953558-2147913662-199702180
User-Agent:Cisco-SIPGateway/IOS-12.x
Allow:INVITE, OPTIONS, BYE, CANCEL, ACK, PRACK, COMET, REFER, SUBSCRIBE, NOTIFY, INFO,
UPDATE, REGISTER
CSeq:102 INVITE
Max-Forwards:70
Remote-Party-ID:"36602" <sip:36602@172.18.193.120>;party=calling;screen=no;privacy=off
Timestamp:1034372426
Contact:<sip:36602@172.18.193.120:5060>
Expires:180
Allow-Events:telephone-event
Proxy-Authorization:Digest
username="36602",realm="example.com",uri="sip:172.18.193.187",response="404feee07cc7d3081d04b97726befef5",nonce="405729fe",cnonce="7D7E41C1",qop=auth,algorithm=MD5,nc=0000001
Content-Type:application/sdp
Content-Length:244
v=0
o=CiscoSystemsSIP-GW-UserAgent 6603 1568 IN IP4 172.18.193.120
s=SIP Call
c=IN IP4 172.18.193.120
```

```
t=0 0
m=audio 17978 RTP/AVP 18 19
c=IN IP4 172.18.193.120
a=rtpmap:18 G729/8000
a=fmtp:18 annexb=no
a=rtpmap:19 CN/8000
a=ptime:20
*Oct 11 21:40:26.187://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Received:
SIP/2.0 100 Trying ! 100 Trying response message (command sequence 102)
Via:SIP/2.0/UDP 172.18.193.120:5060;received=172.18.193.120;branch=z9hG4bK8BA
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>
CSeq:102 INVITE
Content-Length:0
*Oct 11 21:40:26.439://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Received:
SIP/2.0 180 Ringing ! 180 Ringing response message (command sequence 102)
Via:SIP/2.0/UDP 172.18.193.120:5060;received=172.18.193.120;branch=z9hG4bK8BA
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>;tag=003094c2e56a035d4326b6a1-292418c6
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
CSeq:102 INVITE
Server:CSCO/4
Contact:<sip:5550123@172.18.197.182:5060>
Record-Route:<sip:5550123@172.18.193.187:5060;maddr=172.18.193.187>
Content-Length:0
*Oct 11 21:40:28.795://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Received:
SIP/2.0 200 OK ! 200 OK response message (command sequence 102)
Via:SIP/2.0/UDP 172.18.193.120:5060;received=172.18.193.120;branch=z9hG4bK8BA
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>;tag=003094c2e56a035d4326b6a1-292418c6
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
CSeq:102 INVITE
Server:CSCO/4
Contact:<sip:5550123@172.18.197.182:5060>
Record-Route:<sip:5550123@172.18.193.187:5060;maddr=172.18.193.187>
Content-Type:application/sdp
Content-Length:146
v=0
o=Cisco-SIPUA 21297 9644 IN IP4 172.18.197.182
s=SIP Call
c=IN IP4 172.18.197.182
t=0 0
m=audio 28290 RTP/AVP 18
a=rtpmap:18 G729/8000
*Oct 11 21:40:28.799://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Sent:
ACK sip:5550123@172.18.193.187:5060;maddr=172.18.193.187 SIP/2.0 ! ACK request message
(command sequence 102)
Via:SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK20A5
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>;tag=003094c2e56a035d4326b6a1-292418c6
Date:Fri, 11 Oct 2002 21:40:26 GMT
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
Route:<sip:5550123@172.18.197.182:5060>
Max-Forwards:70
CSeq:102 ACK
Proxy-Authorization:Digest
username="36602",realm="example.com",uri="sip:172.18.193.187",response="cc86e13d76426fb65f362c4f56f334",nonce="40529fe",cnonce="9495DEBD",qop=auth,algorithm=MD5,nc=0000002
Content-Length:0
*Oct 11 21:40:32.891://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
```

```
Sent:
BYE sip:5550123@172.18.193.187:5060;maddr=172.18.193.187 SIP/2.0 ! BYE request message
(command sequence 103)
Via:SIP/2.0/UDP  172.18.193.120:5060;branch=z9hG4bK6AF
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>;tag=003094c2e56a035d4326b6a1-292418c6
Date:Fri, 11 Oct 2002 21:40:26 GMT
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
User-Agent:Cisco-SIPGateway/IOS-12.x
Max-Forwards:70
Route:<sip:5550123@172.18.197.182:5060>
Timestamp:1034372432
CSeq:103 BYE
Reason:Q.850;cause=16
Proxy-Authorization:Digest
username="36602",realm="example.com",uri="sip:172.18.193.187",response="9d4617d5978aeaf33cd49d832d12db",nonce="40572fe",cnonce="22B1F32",qop=auth,algorithm=MD5,nc=00000003
Content-Length:0
*Oct 11 21:40:32.895://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Received:
SIP/2.0 100 Trying ! 100 Trying response message (command sequence 103)
Via:SIP/2.0/UDP 172.18.193.120:5060;received=172.18.193.120;branch=z9hG4bK6AF
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>;tag=003094c2e56a035d4326b6a1-292418c6
CSeq:103 BYE
Content-Length:0
*Oct 11 21:40:32.963://-1/xxxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Received:
SIP/2.0 200 OK ! 200 OK response message (command sequence 103)
Via:SIP/2.0/UDP 172.18.193.120:5060;received=172.18.193.120;branch=z9hG4bK6AF
From:"36602" <sip:36602@172.18.193.120>;tag=3E948-4C5
To:<sip:5550123@172.18.193.187>;tag=003094c2e56a035d4326b6a1-292418c6
Call-ID:E35DBEB1-DC9811D6-80098FBE-BE736A4@172.18.193.120
CSeq:103 BYE
Server:CSCO/4
Content-Length:0
```

Sample Output of the debug ccsip events Command

The example shows how the Proxy-Authorization header is broken down into a decoded username and password.

```
Router# debug ccsip events
CCSIP SPI: SIP Call Events tracing is enabled
21:03:21: sippmh_parse_proxy_auth: Challenge is 'Basic'.
21:03:21: sippmh_parse_proxy_auth: Base64 user-pass string is 'MTIzNDU2Nzg5MDEyMzQ1Njou'.
21:03:21: sip_process_proxy_auth: Decoded user-pass string is '1234567890123456:.'.
21:03:21: sip_process_proxy_auth: Username is '1234567890123456'.
21:03:21: sip_process_proxy_auth: Pass is '.'.
21:03:21: sipSPIAddBillingInfoToCcb: sipCallId for billing records =
10872472-173611CC-81E9C73D-F836C2B6@172.18.192.19421:03:21: ****Adding to UAS Request table
```

Sample Output for the debug radius Command

```
Router# debug radius
Radius protocol debugging is on
Radius protocol brief debugging is off
Radius packet hex dump debugging is off
Radius packet protocol debugging is on
Radius packet retransmission debugging is off
Radius server fail-over debugging is off
```

```
Jan 23 14:30:25.421:RADIUS/ENCODE(00071EBF):acct_session_id:742769
Jan 23 14:30:25.421:RADIUS(00071EBF):sending
Jan 23 14:30:25.421:RADIUS:Send to unknown id 25 192.168.41.57:1812, Access-Request, len
179
Jan 23 14:30:25.421:RADIUS: authenticator 88 94 AC 32 89 84 73 6D - 71 00 50 6C D0 F8 FD
11
Jan 23 14:30:25.421:RADIUS: User-Name          [1]   9   "2210001"
Jan 23 14:30:25.421:RADIUS: User-Password      [2]   18  *
Jan 23 14:30:25.421:RADIUS: Vendor, Cisco      [26]  32
Jan 23 14:30:25.421:RADIUS: Cisco AVpair       [1]   26  "resource-service=reserve"
Jan 23 14:30:25.421:RADIUS: Service-Type       [6]   6   Call Check    [10]
Jan 23 14:30:25.421:RADIUS: Vendor, Cisco      [26]  19
Jan 23 14:30:25.421:RADIUS:  cisco-nas-port    [2]   13  "Serial6/0:0"
Jan 23 14:30:25.425:RADIUS: NAS-Port           [5]   6   6144
Jan 23 14:30:25.425:RADIUS: Vendor, Cisco      [26]  29
Jan 23 14:30:25.425:RADIUS:  Cisco AVpair      [1]   23  "interface=Serial6/0:0"
Jan 23 14:30:25.425:RADIUS: Called-Station-Id  [30]  9   "2210001"
Jan 23 14:30:25.425:RADIUS: Calling-Station-Id [31]  9   "1110001"
Jan 23 14:30:25.425:RADIUS: NAS-Port-Type      [61]  6   Async [0]
Jan 23 14:30:25.425:RADIUS: NAS-IP-Address     [4]   6   192.168.81.101
Jan 23 14:30:25.425:RADIUS: Acct-Session-Id    [44]  10  "000B5571"
Jan 23 14:30:25.429:RADIUS:Received from id 25 192.168.41.57:1812, Access-Accept, len 20
Jan 23 14:30:25.429:RADIUS: authenticator 2C 16 63 18 36 56 18 B2 - 76 EB A5 EF 11 45 BE
F4
Jan 23 14:30:25.429:RADIUS:Received from id 71EBF
Jan 23 14:30:25.429:RADIUS/DECODE:parse response short packet; IGNORE
Jan 23 14:30:25.433:RADIUS/ENCODE(00071EBF):Unsupported AAA attribute start_time
Jan 23 14:30:25.433:RADIUS/ENCODE(00071EBF):Unsupported AAA attribute timezone
Jan 23 14:30:25.433:RADIUS/ENCODE:format unknown; PASS
Jan 23 14:30:25.433:RADIUS(00071EBF):sending
Jan 23 14:30:25.433:RADIUS:Send to unknown id 26 192.168.41.57:1813, Accounting-Request,
len 443
Jan 23 14:30:25.433:RADIUS: authenticator DA 1B 03 83 20 90 11 39 - F3 4F 70 F0 F5 8C CC
75
Jan 23 14:30:25.433:RADIUS: Acct-Session-Id    [44]  10  "000B5571"
Jan 23 14:30:25.433:RADIUS: Vendor, Cisco      [26]  56
Jan 23 14:30:25.433:RADIUS:  h323-setup-time   [25]  50  "h323-setup-time=14:30:25.429 GMT
 Wed Jan 23 2002"
Jan 23 14:30:25.433:RADIUS: Vendor, Cisco      [26]  26
Jan 23 14:30:25.433:RADIUS:  h323-gw-id        [33]  20  "h323-gw-id=OrigGW."
Jan 23 14:30:25.433:RADIUS: Vendor, Cisco      [26]  56
Jan 23 14:30:25.433:RADIUS:  Conf-Id           [24]  50  "h323-conf-id=931C146B 0F4411D6
AB5591F0 CBF3D765"
Jan 23 14:30:25.433:RADIUS: Vendor, Cisco      [26]  31
Jan 23 14:30:25.437:RADIUS:  h323-call-origin  [26]  25  "h323-call-origin=answer"
Jan 23 14:30:25.437:RADIUS: Vendor, Cisco      [26]  32
Jan 23 14:30:25.437:RADIUS:  h323-call-type    [27]  26  "h323-call-type=Telephony"
Jan 23 14:30:25.437:RADIUS: Vendor, Cisco      [26]  65
Jan 23 14:30:25.437:RADIUS:  Cisco AVpair      [1]   59  "h323-incoming-conf-id=931C146B
0F4411D6 AB5591F0 CBF3D765"
Jan 23 14:30:25.437:RADIUS: Vendor, Cisco      [26]  30
Jan 23 14:30:25.437:RADIUS:  Cisco AVpair      [1]   24  "subscriber=RegularLine"
Jan 23 14:30:25.437:RADIUS: User-Name          [1]   9   "1110001"
Jan 23 14:30:25.437:RADIUS: Acct-Status-Type   [40]  6   Start              [1]
Jan 23 14:30:25.437:RADIUS: Vendor, Cisco      [26]  19
Jan 23 14:30:25.437:RADIUS:  cisco-nas-port    [2]   13  "Serial6/0:0"
Jan 23 14:30:25.437:RADIUS: NAS-Port           [5]   6   0
Jan 23 14:30:25.437:RADIUS: Vendor, Cisco      [26]  29
Jan 23 14:30:25.437:RADIUS:  Cisco AVpair      [1]   23  "interface=Serial6/0:0"
Jan 23 14:30:25.437:RADIUS: Called-Station-Id  [30]  9   "2210001"
Jan 23 14:30:25.437:RADIUS: Calling-Station-Id [31]  9   "1110001"
Jan 23 14:30:25.437:RADIUS: NAS-Port-Type      [61]  6   Async [0]
Jan 23 14:30:25.437:RADIUS: Service-Type       [6]   6   Login [1]
Jan 23 14:30:25.437:RADIUS: NAS-IP-Address     [4]   6   192.168.81.101
```

```
Jan 23 14:30:25.437:RADIUS: Event-Timestamp    [55]  6   1011796225
Jan 23 14:30:25.437:RADIUS: Delay-Time         [41]  6   0
Jan 23 14:30:25.441:RADIUS/ENCODE(00071EC0):Unsupported AAA attribute start_time
Jan 23 14:30:25.441:RADIUS/ENCODE(00071EC0):Unsupported AAA attribute timezone
Jan 23 14:30:25.441:RADIUS(00071EC0):sending
Jan 23 14:30:25.441:RADIUS:Send to unknown id 27 192.168.41.57:1813, Accounting-Request,
len 411
Jan 23 14:30:25.441:RADIUS: authenticator 15 83 23 D8 0B B2 3A C2 - 1D 8C EF B4 18 0F 1C
65
Jan 23 14:30:25.441:RADIUS: Acct-Session-Id    [44]  10  "000B5572"
Jan 23 14:30:25.441:RADIUS: Vendor, Cisco      [26]  56
Jan 23 14:30:25.441:RADIUS:  h323-setup-time   [25]  50 "h323-setup-time=14:30:25.441 GMT
 Wed Jan 23 2002"
Jan 23 14:30:25.441:RADIUS: Vendor, Cisco      [26]  26
Jan 23 14:30:25.441:RADIUS:  h323-gw-id        [33]  20  "h323-gw-id=OrigGW."
Jan 23 14:30:25.441:RADIUS: Vendor, Cisco      [26]  56
Jan 23 14:30:25.441:RADIUS:  Conf-Id           [24]  50  "h323-conf-id=931C146B 0F4411D6
AB5591F0 CBF3D765"
Jan 23 14:30:25.441:RADIUS: Vendor, Cisco      [26]  34
Jan 23 14:30:25.441:RADIUS:  h323-call-origin  [26]  28  "h323-call-origin=originate"
Jan 23 14:30:25.441:RADIUS: Vendor, Cisco      [26]  27
Jan 23 14:30:25.441:RADIUS:  h323-call-type    [27]  21  "h323-call-type=VoIP"
Jan 23 14:30:25.441:RADIUS: Vendor, Cisco      [26]  65
```

Configuration Examples for SIP AAA Features

SIP - Enhanced Billing Support for Gateways Examples

The following configuration example highlights the minimal configuration options that are necessary to carry out the full feature. After you configure the **aaa username** command described in this document, the gateway uses the information received in the SIP Authorization header and makes it available to AAA and Tcl IVR services. Typically, if you expect to use the full functionality of this feature, AAA and Tcl IVR have been configured previously.

```
Router# show running-config
Building configuration...
Current configuration : 4017 bytes
!
version 12.3
no service single-slot-reload-enable
service timestamps debug datetime msec
service timestamps log uptime
no service password-encryption
!
hostname 3640-1
!
logging rate-limit console 10 except errors
! Need the following aaa line
aaa new-model
!
! Need the following four aaa lines
aaa authentication login h323 group radius
aaa authorization exec h323 group radius
aaa accounting connection h323 start-stop group radius
aaa session-id common
enable password lab
!
memory-size iomem 15
```

```
clock timezone GMT 0
voice-card 2
!
ip subnet-zero!
ip domain-name example.sip.com
ip name-server 172.18.192.154
ip name-server 10.10.1.5
!
no ip dhcp-client network-discovery
isdn switch-type primary-5ess
isdn voice-call-failure 0
!
voice service voip
sip
rel1xx disable
!
fax interface-type fax-mail
mta receive maximum-recipients 0
call-history-mib retain-timer 500
!
controller E1 1/0
!
controller E1 1/1
!
controller T1 2/0
framing esf
linecode b8zs
pri-group timeslots 1-24
!
controller T1 2/1
framing sf
linecode ami
!
! Need the following three lines
gw-accounting h323
gw-accounting h323 vsa
gw-accounting voip
!
interface Ethernet0/0
ip address 10.10.1.4 255.255.255.0
half-duplex
ip rsvp bandwidth 7500 7500
!
interface Ethernet0/1
no ip address
shutdown
half-duplex
!
interface Ethernet0/2
no ip address
shutdown
half-duplex
!
interface Ethernet0/3
no ip address
shutdown
half-duplex
!
interface FastEthernet1/0
ip address 172.18.192.197 255.255.255.0
duplex auto
speed auto
ip rsvp bandwidth 75000 75000
!
```

```
interface Serial2/0:23
no ip address
no logging event link-status
isdn switch-type primary-5ess
isdn incoming-voice modem
isdn T306 200000
isdn T310 200000
no cdp enable
!
ip classless
ip route 10.0.0.0 255.0.0.0 172.18.192.1
ip route 172.18.0.0 255.255.0.0 172.18.192.1
no ip http server
!
ip radius source-interface FastEthernet1/0
logging source-interface FastEthernet1/0
!
! Need the following radius-server lines for accounting/authentication
radius-server host 172.18.192.154 auth-port 1645 acct-port 1646
radius-server retransmit 1
radius-server key lab
radius-server vsa send accounting
radius-server vsa send authentication
call rsvp-sync
!
! Need the following call application lines in order to enable
! tcl scripting feature.
call application voice voice_billing tftp://172.18.207.15/app_passport_silent.2.0.0.0.tcl
!
voice-port 2/0:23
!
voice-port 3/0/0
!
voice-port 3/0/1
!
voice-port 3/1/0
!
voice-port 3/1/1
!
mgcp profile default
dial-peer cor custom
!
dial-peer voice 3640110 pots
destination-pattern 3640110
port 3/0/0
!
dial-peer voice 3640120 pots
destination-pattern 3640120
port 3/0/1
!
dial-peer voice 3660110 voip
destination-pattern 3660110
session protocol sipv2
session target ipv4:172.18.192.194
codec g711ulaw
!
dial-peer voice 3660120 voip
destination-pattern 3660120
session protocol sipv2
session target ipv4:172.18.192.194
codec g711ulaw
!
dial-peer voice 222 pots
huntstop
```

```
application session
destination-pattern 222
no digit-strip
direct-inward-dial
port 2/0:23
!
! Need to add the application line below to enable the tcl script
dial-peer voice 999 voip
application voice_billing
destination-pattern ...
session protocol sipv2
session target ipv4:10.10.1.2:5061
codec g711ulaw
!
! Need to add the aaa line below in order to enable proxy-authorization
! header processing
sip-ua
aaa username proxy-auth
!
line con 0
exec-timeout 0 0
length 0
line aux 0
line vty 0 4
!
!end
```

SIP Gateway HTTP Authentication Digest Examples

SIP: Gateway HTTP Authentication Digest Feature Disabled

```
Router# show running-config
Building configuration...
Current configuration :4903 bytes
!
version 12.3
no parser cache
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
service internal
!
hostname Router
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$Fyay$DfmV/uLXX.X94CoaRy569.
enable password lab
!
voice-card 3
!
aaa new-model
!
aaa authentication login h323 group radius
aaa authorization exec h323 group radius
aaa accounting connection h323 start-stop group radius
aaa session-id common
ip subnet-zero
ip tcp path-mtu-discovery
!
```

```
ip cef
ip domain name example.sip.com
ip name-server 172.18.192.48
!
ip dhcp pool 1
host 172.18.193.173 255.255.255.0
client-identifier 0030.94c2.5d00
  option 150 ip 172.18.193.120
  default-router 172.18.193.120
!
voice call carrier capacity active
!
voice service pots
!
voice service voip
sip
  rel1xx disable
!
voice class codec 1
 codec preference 1 g729r8
 codec preference 2 g711ulaw
 codec preference 5 g726r16
 codec preference 6 g726r24
 codec preference 7 g726r32
 codec preference 8 g723ar53
 codec preference 9 g723ar63
!
voice class codec 2
 codec preference 1 g711ulaw
 codec preference 2 g729r8
 codec preference 5 g726r16
 codec preference 6 g726r24
!
fax interface-type fax-mail
!
translation-rule 100
!
interface FastEthernet0/0
 ip address 172.18.193.120 255.255.255.0
 ip mtu 900
 duplex auto
 speed auto
 no cdp enable
 ip rsvp bandwidth 75000 75000
!
interface FastEthernet0/1
 no ip address
 no ip mroute-cache
 shutdown
 duplex auto
 speed auto
 no cdp enable
!
ip http server
ip classless
ip route 0.0.0.0 0.0.0.0 FastEthernet0/0
ip route 10.0.0.0 255.0.0.0 172.18.193.1
ip route 172.18.0.0 255.255.0.0 172.18.193.1
!
ip radius source-interface FastEthernet0/0
logging source-interface FastEthernet0/0
dialer-list 1 protocol ip permit
snmp-server engineID local 00000009020000309426F6D0
snmp-server community public RO
```

```
snmp-server community private RW
snmp-server packetsize 4096
snmp-server enable traps tty
!
tftp-server flash:XMLDefault.cnf.xml
!
radius-server host 172.18.192.108 auth-port 1645 acct-port 1646
radius-server retransmit 1
radius-server key lab
radius-server vsa send accounting
radius-server vsa send authentication
!
control-plane
!
voice-port 1/0/0
!
voice-port 1/0/1
!
voice-port 1/1/0
!
voice-port 1/1/1
!
voice-port 2/0/0
 station-id name 36602
 station-id number 36602
!
voice-port 2/0/1
!
mgcp
mgcp sdp simple
!
dial-peer cor custom
!
dial-peer voice 1 pots
 application session
 destination-pattern 36602
 port 2/0/0
!
dial-peer voice 5 voip
 application session
 destination-pattern 5550123
 session protocol sipv2
 session target ipv4:172.18.193.187
!
dial-peer voice 81 voip
 application session
 destination-pattern 3100801
 session protocol sipv2
 session target ipv4:172.18.193.100
 req-qos controlled-load
 acc-qos controlled-load
!
dial-peer voice 41 voip
 application session
 destination-pattern 333
 session protocol sipv2
 session target ipv4:10.102.17.80
 dtmf-relay rtp-nte
!
dial-peer voice 7 voip
 application session
 destination-pattern 999
 session protocol sipv2
 session target ipv4:172.18.193.98
```

```
 incoming called-number 888
!
dial-peer voice 38 voip
 application session
 destination-pattern 3100802
 voice-class codec 1
 session protocol sipv2
 session target ipv4:172.18.193.99
!
dial-peer voice 88 voip
 preference 1
 destination-pattern 888
 session protocol sipv2
 session target ipv4:172.18.193.187
!
dial-peer voice 123 voip
 destination-pattern 222
 session protocol sipv2
 session target ipv4:10.102.17.80
!
dial-peer voice 6 voip
 destination-pattern 36601
 session protocol sipv2
 session target ipv4:172.18.193.98
 session transport udp
 incoming called-number 36602
!
gateway
 timer receive-rtp 1200
!
sip-ua
 retry invite 1
 retry bye 2
 timers expires 60000
!
rtr responder
!
line con 0
 exec-timeout 0 0
 transport preferred all
 transport output all
line aux 0
 transport preferred all
 transport output all
line vty 0 4
 password lab
 transport preferred all
 transport input all
 transport output all
!
end
```

SIP: Gateway HTTP Authentication Digest Feature Enabled

```
Router# show running-config
Building configuration...
Current configuration :5087 bytes
!
version 12.3
no parser cache
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
```

```
service internal
!
hostname Router
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$Fyay$DfmV/uLXX.X94CoaRy569.
enable password lab
!
voice-card 3
!
aaa new-model
!
aaa authentication login h323 group radius
aaa authorization exec h323 group radius
aaa accounting connection h323 start-stop group radius
aaa session-id common
ip subnet-zero
ip tcp path-mtu-discovery
!
ip cef
ip domain name example.sip.com
ip name-server 172.18.192.48
!
ip dhcp pool 1
  host 172.18.193.173 255.255.255.0
  client-identifier 0030.94c2.5d00
  option 150 ip 172.18.193.120
  default-router 172.18.193.120
!
voice call carrier capacity active
!
voice service pots
!
voice service voip
 sip
  rel1xx disable
!
voice class codec 1
 codec preference 1 g729r8
 codec preference 2 g711ulaw
 codec preference 5 g726r16
 codec preference 6 g726r24
 codec preference 7 g726r32
 codec preference 8 g723ar53
 codec preference 9 g723ar63
!
voice class codec 2
 codec preference 1 g711ulaw
 codec preference 2 g729r8
 codec preference 5 g726r16
 codec preference 6 g726r24
!
fax interface-type fax-mail
!
translation-rule 100
!
interface FastEthernet0/0
ip address 172.18.193.120 255.255.255.0
 ip mtu 900
 duplex auto
 speed auto
 no cdp enable
```

```
 ip rsvp bandwidth 75000 75000
!
interface FastEthernet0/1
 no ip address
 no ip mroute-cache
 shutdown
 duplex auto
 speed auto
 no cdp enable
!
ip http server
ip classless
ip route 0.0.0.0 0.0.0.0 FastEthernet0/0
ip route 10.0.0.0 255.0.0.0 172.18.193.1
ip route 172.18.0.0 255.255.0.0 172.18.193.1
!
ip radius source-interface FastEthernet0/0
logging source-interface FastEthernet0/0
dialer-list 1 protocol ip permit
snmp-server engineID local 00000009020000309426F6D0
snmp-server community public RO
snmp-server community private RW
snmp-server packetsize 4096
snmp-server enable traps tty
!
tftp-server flash:XMLDefault.cnf.xml
!
radius-server host 172.18.192.108 auth-port 1645 acct-port 1646
radius-server retransmit 1
radius-server key lab
radius-server vsa send accounting
radius-server vsa send authentication
!
control-plane
!
voice-port 1/0/0
!
voice-port 1/0/1
!
voice-port 1/1/0
!
voice-port 1/1/1
!
voice-port 2/0/0
 station-id name 36602
 station-id number 36602
!
voice-port 2/0/1
!
mgcp
mgcp sdp simple
!
dial-peer cor custom
!
dial-peer voice 1 pots
 application session
 destination-pattern 36602
 port 2/0/0
 authentication username user1 password password1 realm example1.com ! authentication
xample 1
 authentication username user2 password password2 realm example2.com ! authentication
xample 2
!
dial-peer voice 5 voip
```

```
 application session
 destination-pattern 5550123
 session protocol sipv2
 session target ipv4:172.18.193.187
!
dial-peer voice 81 voip
 application session
 destination-pattern 3100801
 session protocol sipv2
 session target ipv4:172.18.193.100
 req-qos controlled-load
 acc-qos controlled-load
!
dial-peer voice 41 voip
 application session
 destination-pattern 333
 session protocol sipv2
 session target ipv4:10.102.17.80
 dtmf-relay rtp-nte
!
dial-peer voice 7 voip
 application session
 destination-pattern 999
 session protocol sipv2
 session target ipv4:172.18.193.98
 incoming called-number 888
!
dial-peer voice 38 voip
 application session
 destination-pattern 3100802
 voice-class codec 1
 session protocol sipv2
 session target ipv4:172.18.193.99
!
dial-peer voice 88 voip
 preference 1
 destination-pattern 888
 session protocol sipv2
 session target ipv4:172.18.193.187
!
dial-peer voice 123 voip
 destination-pattern 222
 session protocol sipv2
 session target ipv4:10.102.17.80
!
dial-peer voice 6 voip
 destination-pattern 36601
 session protocol sipv2
 session target ipv4:172.18.193.98
 session transport udp
 incoming called-number 36602
!
gateway
 timer receive-rtp 1200
!
sip-ua
 authentication username user3 password password3 ! authentication example 3
 retry invite 1
 retry bye 2
 timers expires 60000
 registrar ipv4:172.18.193.187 expires 100 ! registrar example
!
rtr responder
!
```

```
line con 0
 exec-timeout 0 0
 transport preferred all
 transport output all
line aux 0
 transport preferred all
 transport output all
line vty 0 4
 password lab
 transport preferred all
 transport input all
 transport output all
!
end
```

Additional References

General SIP References

References Mentioned in This Chapter (Listed Alphabetically)

- *Cisco IOS Security Command Reference*

- *Cisco IOS Security Configuration Guide,*

- *Cisco IOS SIP Configuration Guide*

- *Cisco IOS Tcl IVR and VoiceXML Application Guide*

- *Cisco Resource Policy Management System 2.0* at
  http://www.cisco.com/en/US/products/sw/netmgtsw/ps2074/tsd_products_support_eol_series_home.html

- *Cisco Tcl IVR API Programmer's Guide*

- *Enhancements to the Session Initiation Protocol for VoIP on Cisco Access Platforms*

- *Inter-Domain Gatekeeper Security Enhancement*, Cisco IOS Release 12.2(4)T

- *RADIUS Vendor-Specific Attributes Voice Implementation Guide* at
  http://noc.hsdn.org/files/univercd/cc/td/doc/product/access/acs_serv/vapp_dev/vsaig3.htm