PROVISIONING GUIDE

Cisco SPA1112, SPA122, SPA232D Analog Telephone Adapters

Contents

1

Deployment and Provisioning

Cisco IP Telephony devicesSPA100 and SPA200 Series ATAs are intended for high-volume deployments by VoIP service providers to residential and small business customers. In business or enterprise environments, these ATAs can serve as terminal nodes. These devices are widely distributed across the Internet, connected through routers and firewalls at the customer premises.

The IP Telephony device can be used as a remote extension of the service provider back-end equipment. Remote management and configuration ensures the proper operation of the IP Telephony device at the customer premises.

This customized, ongoing configuration is supported by the following features:

- Reliable remote control of the endpoint
- Encryption of the communication controlling the endpoint
- Streamlined endpoint account binding

This chapter describes the features and functionality available when provisioning these ATAs and explains the setup required:

- **Deployment, page 7**
- **Provisioning Overview, page 9**

Deployment

These ATAs provide convenient mechanisms for provisioning, based on two deployment models:

- Bulk distribution—The service provider acquires these ATAs in bulk quantity and either preprovisions them in-house or purchases RC units from Cisco. The devices are then issued to the customers as part of a VoIP service contract.
- Retail distribution—The customer purchases the ATA from a retail outlet and requests VoIP service from the service provider. The service provider must then support the secure remote configuration of the device.

Bulk Distribution

In this model, the service provider issues these ATAs to its customers as part of a VoIP service contract. The devices are either RC units or preprovisioned in-house.

RC units are preprovisioned by Cisco to resynchronize with a Cisco server that downloads the device profile and firmware updates.

A service provider can preprovision these ATAs with the desired parameters, including the parameters that control resynchronization, through various methods: in-house by using DHCP and TFTP; remotely by using TFTP, HTTP, or HTTPS; or a combination of in-house and remote provisioning.

RC Unit Deployment

RC units eliminate in-house preprovisioning of and reduce the need for the service provider to physically handle the devices prior to shipping them to end customers. This approach also discourages the use of these ATAs with an inappropriate service provider.

A RC unit is preprovisioned by Cisco with the connection information for the provisioning servers. These servers are maintained by Cisco Systems, Inc. for the service provider that purchased the units. The MAC address of each RC unit is associated with a customizable profile on the Cisco provisioning servers. When the RC unit is connected to the broadband link, it contacts the Cisco provisioning server and downloads its customized profile.

The service provider works with a Cisco sales engineer to develop a simple provisioning profile. The profile contains minimal information that redirects the device to the service provider provisioning server. This profile is placed on the Cisco RC server by the Cisco Voice Team.

RC Unit Status

The status of an RC unit can be determined by viewing the Info > Product Information page, Customization section, on the administration web server. An RC unit that has not been provisioned displays **Pending**. An RC unit that has been provisioned displays the name of the company that owns the unit. If the unit is not an RC unit, the page displays **Open**.

Below is a sample template for an RC unit to be preprovisioned by Cisco with the connection information:

```
Restricted Access Domains "domain.com, domain1.com, domain2.com";
Primary_DNS               * "x.y.w.z";
Secondary_DNS             * "a.b.c.d";
Provision_Enable          * "Yes";
Resync_Periodic           * "30";
Resync_Error_Retry_Delay  * "30";
Profile_Rule  * "http://prov.domain.com/sipura/profile?id=$MA";
```

The `Restricted Access Domains` parameter is configured with the actual domain names of up to a maximum of five domains. The Primary_DNS and Secondary_DNS parameters are configured with the actual domain names or IP addresses of the DNS servers available to the RC unit.

Retail Distribution

In a retail distribution model, a customer purchases a Cisco IP Telephony deviceATA and subscribes to a particular service. The Internet Telephony Service Provider (ITSP) sets up and maintains a provisioning server, and preprovisions the phone to resynchronize with the service provider server. See **In-House Device Preprovisioning, page 39** for more information.

The customer signs on to the service and establishes a VoIP account, possibly through an online portal, and binds the device to the assigned service account. When the device is powered up or a specified time elapses, the IP Telephony device resynchronizes, downloading the latest parameters. These parameters can address goals such as setting up a hunt group, setting speed dial numbers, and limiting the features that a user can modify.

Resynchronization Process

The firmware for each ATA includes an administration web server that accepts new configuration parameter values. The ATA is instructed to resync with a specified provisioning server through a resync URL command in the device profile. The URL command typically includes an account PIN number or alphanumeric code to associate the device with the new account. For example:

```
http://192.168.1.102/admin/resync?https://prov.supervoip.com/cisco-init/
1234abcd
```

In this example, a device at the DHCP-assigned IP address `192.168.1.102` is instructed to provision itself to the SuperVoIP service at `prov.supervoip.com`. The PIN number for the new account is `1234abcd`. The remote provisioning server is configured to associate the ATA that is performing the resync request with the new account, based on the URL and PIN.

Through this initial resync operation, the ATA is configured in a single step, and is automatically directed to resync thereafter to a permanent URL on the server.

For both initial and permanent access, the provisioning server relies on the client certificate for authentication and supplies configuration parameter values based on the associated service account.

Provisioning Overview

An IP Telephony device can be configured to resynchronize its internal configuration state to match a remote profile periodically and on power up by contacting a normal provisioning server (NPS) or an access control server (ACS).

By default, a profile resync is only attempted when the IP Telephony device is idle, because the upgrade might trigger a software reboot interrupting a call. If intermediate upgrades are required to reach a current upgrade state from an older release, the upgrade logic is capable of automating multi-stage upgrades.

NPS

The NPS can be a TFTP, HTTP, or HTTPS server. A remote firmware upgrade is achieved by using TFTP or HTTP, but not by using HTTPS because the firmware does not contain sensitive information.

Communication with the NPS does not require the use of a secure protocol because the updated profile can be encrypted by a shared secret key. Secure first-time provisioning is provided through a mechanism that uses SSL functionality. An unprovisioned ATA can receive a 256-bit symmetric key encrypted profile specifically targeted for that device.

TR-069

The digital subscriber line (DSL) Forum TR-069, CPE WAN Management Protocol (CWMP), is used for communications between a customer premise equipment (CPE) device and an auto-configuration server (ACS). The TR-069 Agent manages a collection of CPE devices, with the primary capability for auto-configuration and dynamic service provisioning, software image management, status and performance monitoring and diagnostics.

It supports multiple scenarios, including:

- Device administration: Authenticates administrators, authorizes commands, and provides an audit trail
- Remote Access: Works with VPN and other remote network access devices to enforce access policies
- Network admission control: Communicates with posture and audit servers to enforce admission control policies

The TR-069 Agent CPE devices must be set up and enabled for TR-069. An ACS used to communicate with the CPE must be TR-069 compliant in order to enable the TR-069 Agent.

Provisioning States

The provisioning process involves these provisioning states.

State	Description
MFG-RESET Manufacturing Reset	The device returns to a fully unprovisioned state; all configurable parameters regain their default values. Manufacturing reset can be performed through the IVR sequence `****RESET#1#`. On phones that do not support IVR, power cycle the phone to reset it to the default values. Allowing the end user to perform a manufacturing reset guarantees that the device can always be returned to an accessible state.

State	Description
SP-CUST Service Provider Customization	The Profile_Rule parameter points to a device-specific configuration profile by using a provisioning server that is specific to the service provider. The methods for initiating resynchronization are: ▪ Auto-configuration by using a local DHCP server. A TFTP server name or IPv4 address is specified by DHCP. The TFTP server includes the Profile_Rule parameter in the configuration file. ▪ Entering a resync URL. The URL starts a web browser and requests a resync to a specific TFTP server by entering the URL syntax: `http://x.x.x.x/admin/resync?prvserv/device.cfg`, where: *x.x.x.x* is the IP address of the IP Telephony device, *prvserv* is the target TFTP server, and *device.cfg* is the name of the configuration file on the server. ▪ Editing the Profile_Rule parameter by opening the provisioning pane on the web interface and entering the TFTP URL in the Profile_Rule parameter. For example, *prserv*/spa962.cfg. ▪ Modifying the configuration file Profile_Rule and to contact a specific TFTP server and request a configuration file identified by the MAC-address. For example, this entry contacts a provisioning server, requesting a profile unique to the device with a MAC address identified by the `$MA` parameter: `Profile_Rule tftp.callme.com/profile/$MA/spa962.cfg;`

State	Description

SEC-PRV-1 Secure Provisioning—Initial Configuration	An initial, device-unique CFG file is targeted to a IP Telephony device by compiling the CFG file with the SPC `--target` option. This provides an encryption that does not require the exchange of keys.
	
	The initial, device-unique CFG file reconfigures the device profile to enable stronger encryption by programming a 256-bit encryption key and pointing to a randomly-generated TFTP directory. For example, the CFG file might contain:
	
	`Profile_Rule [--key $A] tftp.callme.com/profile/$B/spa962.cfg;`
	`GPP_A 8e4ca259…;  # 256 bit key`
	`GPP_B Gp3sqLn…;  # random CFG file path directory`

SEC-PRV-2 Secure Provisioning—Full Configuration	Profile resync operations subsequent to the initial SEC-PRV-1 provisioning retrieve the 256-bit encrypted CFG files that maintain the IP Telephony device in a state synchronized to the provisioning server. The profile parameters are reconfigured and maintained through this strongly encrypted profile. The encryption key and random directory location in the SEC-PRV-2 configuration can be changed periodically for extra security.

Configuration Access Control

The IP Telephony device firmware provides mechanisms for restricting end-user access to some parameters. The firmware provides specific privileges for login to an **Admin** account or a **User** account. Each can be independently password protected.:

- Admin Account—Allows the service provider full access to all interactive voice response (IVR) functions and to all administration web server parameters.
- User Account—Allows the user to access basic IVR functions and to configure a subset of the administration web server parameters.

The service provider can restrict the user account in the provisioning profile in the following ways:

- Indicate which configuration parameters are available to the User account when creating the configuration. (Described in **"Element Tags" on page 16**.)
- Disable user access to the administration web server.
- Disable the factory reset control by using the IVR.
- Restrict the Internet domains accessed by the device for resync, upgrades, or SIP registration for Line 1.

Communication Encryption

The configuration parameters communicated to the device can contain authorization codes or other information that protect the system from unauthorized access. It is in the service provider's interest to prevent unauthorized activity by the customer, and it is in the customer's interest to prevent the unauthorized use of the account. The service provider can encrypt the configuration profile communication between the provisioning server and the device, in addition to restricting access to the administration web server.

2

Creating XML Provisioning Scripts

The configuration profile defines the parameter values for the ATA.

Standard XML authoring tools are used to compile the parameters and values. To protect confidential information in the configuration profile, this type of file is typically delivered from the provisioning server to the ATA over a secure channel provided by HTTPS. See **Compression and Encryption, page 20**.

NOTE Only UTF-8 charset is supported. If you modify the profile in an editor, do not change the encoding format; otherwise, the ATA cannot recognize the file.

File Structure

The profile is a text file with XML-like syntax in a hierarchy of elements, with element attributes and values. This format lets you use standard tools to create the configuration file. A configuration file in this format can be sent from the provisioning server to the ATA during a resync operation without compiling the file as a binary object.

You can obtain the profile for your ATA by logging on to your ATA and then entering the path to the file: http://*<LAN_IP_address>*/admin/config.xml
For example, using the default IP address of the ATA, you would enter: http://192.168.15.1/admin/config.xml

To protect confidential information contained in the configuration profile, this file is generally delivered from the provisioning server to the ATA over a secure channel provided by HTTPS. Optionally, the file can be compressed by using the gzip deflate algorithm (RFC1951). In addition, the file can be encrypted by using 256-bit AES symmetric key encryption.

Example: Open Profile Format

```
<flat-profile>
```

```
<Resync_On_Reset> Yes
  </Resync_On_Reset>
<Resync_Periodic> 7200
  </Resync_Periodic>
<Profile_Rule>
    tftp://prov.telco.com:6900/cisco/config/spa504.cfg
  </Profile_Rule>
</flat-profile>
```

The `<flat-profile>` element tag encloses all parameter elements to be recognized by the ATA.

Element Tags, Attributes, Parameters, and Formatting

A file can include element tags, attributes, parameters, and formatting features.

Element Tags

The properties of element tags are:

- The ATA recognizes elements with proper parameter names, when encapsulated in the special `<flat-profile>` element.
- The `<flat-profile>` element can be encapsulated within other arbitrary elements.
- Element names are enclosed in angle brackets.
- Most of the element names are similar to the field names in the administration web pages for the device, with the following modifications:
  - Element names may not include spaces or special characters. To derive the element name from the administration web field name, substitute an underscore for every space or the special characters `[`, `]`, `(`, `)`, or `/`.

    For example, the Resync On Reset field is represented by the element <Resync_On_Reset>.
  - Each element name must be unique. In the administration web pages, the same fields might appear on multiple web pages, such as the Line,

User, and Extension pages. Append `[n]` to the element name to indicate the number that is shown in the page tab.

For example, the Dial Plan for Line 1 is represented by the element <Dial_Plan[1]>

- Each opening element tag must be matched by a corresponding closing element tag. For example:

```
<flat-profile>
<Resync_On_Reset> Yes
    </Resync_On_Reset>
<Resync_Periodic> 7200
    </Resync_Periodic>
<Profile_Rule>tftp://prov.telco.com: 6900/cisco/config/
spa962.cfg
    </Profile_Rule>
</flat-profile>
```

- Element tags are case sensitive.
- Empty element tags are allowed. Enter the opening element tag without a corresponding element tag, and insert a space and a forward slash before the less-than symbol. In this example, Profile Rule B is empty:
  `<Profile_Rule_B />`
- Unrecognized element names are ignored.
- An empty element tag can be used to prevent the overwriting of any user-supplied values during a resync operation. In the following example, the user speed dial settings are unchanged:

```
<Speed_Dial_2_2_  ua="rw"/>
  <Speed_Dial_3_2_  ua="rw"/>
  <Speed_Dial_4_2_  ua="rw"/>
  <Speed_Dial_5_2_  ua="rw"/>
  <Speed_Dial_6_2_  ua="rw"/>
  <Speed_Dial_7_2_  ua="rw"/>
  <Speed_Dial_8_2_  ua="rw"/>
  <Speed_Dial_9_2_  ua="rw"/>
</flat-profile>
```

- An empty value can be used to set the corresponding parameter to an empty string. Enter an opening and closing element without any value between them. In the following example, the GPP_A parameter is set to an empty string.

```
<flat-profile>
<GPP_A>
  </GPP_A>
</flat-profile>
```

User Access

The **ua** attribute controls access by the User account for specific parameters. If the **ua** attribute is not specified in an element tag, the factory default user access is applied for the corresponding parameter is applied. Access by the Admin account is unaffected by this attribute.

The **ua** attribute, if present, must have one of the following values:

- na—no access
- ro—read-only
- rw—read/write

The **ua** attribute is illustrated by the following example:

```
<flat-profile>
  <SIP_TOS_DiffServ_Value_1_   ua="na"/>
  <Dial_Plan_1_   ua="ro"/>
  <Dial_Plan_2_   ua="rw"/>
</flat-profile>
```

The value of the **ua** option must be enclosed by double quotes.

Parameter Properties

These properties apply to the parameters:

- Any parameters that are not specified by a profile are left unchanged in the ATA.
- Unrecognized parameters are ignored.

- The ATA recognizes arbitrary, configurable aliases for a limited number of parameter names.
- If the Open format profile contains multiple occurrences of the same parameter tag, the last such occurrence overrides any earlier ones. To avoid inadvertently overriding configuration values for a parameter, it is recommended that at most one instance of a parameter be specified in any one profile.

Formatting

These properties apply to the formatting of the strings:

- Comments are allowed by using standard XML syntax.

```
<!-- My comment is typed here -->
```

- Leading and trailing white space is allowed for readability and will be removed from the parameter value.
- New lines within a value are converted to spaces.
- An XML header of the form <? . . . ?> is allowed, but is ignored by the ATA.
- To enter special characters, use basic XML character escapes, as shown in the following table.

Special Character	XML Escape Sequence
& (ampersand)	&amp;
< (less than)	&lt;
> (greater than)	&gt;
’ (apostrophe)	&apos;
” (double quote)	&quot;

In the following example, character escapes are entered to represent the greater than and less than symbols that are required in a dial plan rule. This example defines an information hotline dial plan that sets the Dial_Plan[1] parameter equal to (S0 <:18005551212>).

```
<flat-profile>
 <Dial_Plan_1_>
    (S0 &lt;:18005551212&gt;)
 </Dial_Plan_1_>
```

```
</flat-profile>
```

- Numeric character escapes, using decimal and hexadecimal values (s.a. `&#40;` and `&#x2e;`), are translated.
- The firmware does not support the full Unicode character set, but only the ASCII subset.

Compression and Encryption

The configuration profile can be compressed to reduce the network load on the provisioning server. It also can be encrypted to protect confidential information. Compression is not required, but it must precede encryption.

The supported compression method is the gzip deflate algorithm (RFC1951). The gzip utility and the compression library that implements the same algorithm (zlib) are available from Internet sites.

To identify when compression is applied, the ATA expects the compressed file to contain a gzip compatible header, as generated by invoking the gzip utility on the original Open profile. The ATA inspects the downloaded file header to determine the format of the file.

For example, if `profile.xml` is a valid profile, the file `profile.xml.gz` is also accepted. This profile type can be generated with either of the following commands:

```
>gzip profile.xml
```

replaces original file with compressed file.

```
>cat profile.xml | gzip > profile.xml.gz
```

leaves original file in place, produces new compressed file.

A tutorial on compression is provided in **Open Profile gzip Compression, page 61**.

Encryption by using AES

A configuration profile can be encrypted by using symmetric key encryption, whether or not the file is compressed. The supported encryption algorithm is the American Encryption Standard (AES), using 256-bit keys, applied in cipher block chaining mode.

NOTE Compression must precede encryption for the ATA to recognize a compressed and encrypted profile. A tutorial on encryption is provided in **Profile Encryption by using OpenSSL, page 62**.

The OpenSSL encryption tool, available for download from various Internet sites, can be used to perform the encryption. Support for 256-bit AES encryption might require recompilation of the tool (to enable the AES code). The firmware has been tested against version openssl-0.9.7c.

If the file is encrypted, the profile expects the file to have the same format as generated by the following command:

```
# example encryption key = SecretPhrase1234

openssl enc –e –aes-256-cbc –k SecretPhrase1234 –in profile.xml –out
profile.cfg

# analogous invocation for a compressed xml file

openssl enc –e –aes-256-cbc –k SecretPhrase1234 –in profile.xml.gz –out
profile.cfg
```

A lower case -k precedes the secret key, which can be any plain text phrase and is used to generate a random 64-bit salt. Then, in combination with the secret specified with the -k argument, the encryption tool derives a random 128-bit initial vector, and the actual 256-bit encryption key.

When this form of encryption is used to encrypt a configuration profile, the ATA must be informed of the secret key value to decrypt the file. This value is specified as a qualifier in the profile URL. The syntax is as follows, using an explicit URL:

```
[--key "SecretPhrase1234"] http://prov.telco.com/path/profile.cfg
```

This value is programmed by using one of the Profile_Rule parameters. The key must be preprovisioned into the unit at an earlier time. This bootstrap of the secret key can be accomplished securely by using HTTPS.

Preencrypting configuration profiles offline with symmetric key encryption allows the use of HTTP for resyncing profiles. The provisioning server uses HTTPS to handle initial provisioning of the ATA after deployment. This feature reduces the load on the HTTPS server in large scale deployments.

The final file name does not need to follow a specific format, but it is conventional to end the name with the .cfg extension to indicate that it is a configuration profile.

Applying a Profile to the ATA

After you create an XML configuration script, it must be passed to the ATA for application. To apply the configuration, choose one of the following methods:

TFTP and the Resync URL

Complete the following steps to post the configuration file to a TFTP server application on your PC.

STEP 1 Connect your PC to the ATA LAN.

STEP 2 Run a TFTP server application on the PC and make sure that the configuration file is available in the TFTP root directory.

STEP 3 In a web browser, and enter the LAN IP address of the ATA, the IP address of the computer, the filename, and the login credentials, in this format:

http://*<WAN_IP_Address>*/admin/resync?tftp://*<PC_IP_Address>*/*<file_name>*&xuser=admin&xpassword=*<password>*

Example:
`http://192.168.15.1/admin/resync?tftp://192.168.15.100/my_config.xml&xuser=admin&xpassword=admin`

Direct HTTP Post using cURL

Complete the following steps to post the configuration to the ATA by using cURL. This command line tool is used to transfer data with a URL syntax. To download cURL, see:

http://curl.haxx.se/download.html

STEP 1 Connect your PC to the LAN port of the ATA.

STEP 2 Post the configuration file to the ATA by entering the following cURL command:

```
curl -d @my_config.xml "http://192.168.15.1/admin/config.xml&
xuser=admin&xpassword=admin"
```

Using Provisioning Parameters

This section describes the provisioning parameters broadly organized according to function:

- **General Purpose Parameters**
- **Enables**
- **Triggers**
- **Configurable Schedules**
- **Profile Rules**
- **Report Rule**
- **Upgrade Rule**

NOTE Additional parameters are described in **Chapter 6, "Voice Parameters"** and **Chapter 7, "Router Configuration Parameters."**

General Purpose Parameters

The general purpose parameters GPP_* are used as free string registers when configuring the ATA to interact with a particular provisioning server solution. The GPP_* parameters are empty by default. They can be configured to contain diverse values, including the following:

- Encryption keys
- URLs
- Multi-stage provisioning status information
- Post request templates

- Parameter name alias maps
- Partial string values, eventually combined into complete parameter values.

The GPP_* parameters are available for macro expansion within other provisioning parameters. For this purpose, single-letter upper-case macro names (A through P) are sufficient to identify the contents of GPP_A through GPP_P. Also, the two-letter upper-case macro names SA through SD identify GPP_SA through GPP_SD as a special case when used as arguments of the key URL option.

For example, if GPP_A contains the string ABC, and GPP_B contains 123, the expression AB macro expands into ABC123.

Enables

All profile resync and firmware upgrade operations are controlled by the Provision_Enable and Upgrade_Enable parameters. These parameters control resyncs and upgrades independently of each other. These parameters also control resync and upgrade URL commands issued through the administration web server. Both of these parameters are set to yes by default.

In addition, the Resync_From_SIP parameter controls requests for resync operations via a SIP NOTIFY event sent from the service provider proxy server to the ATA. If enabled, the proxy can request a resync by sending a SIP NOTIFY message containing the Event: resync header to the device.

The device challenges the request with a 401 response (authorization refused for used credentials), and expects an authenticated subsequent request before honoring the resync request from the proxy. The Event: reboot_now and Event: restart_now headers perform cold and warm restarts, respectively, are also challenged.

The two remaining enables are Resync_On_Reset and Resync_After_Upgrade_Attempt. These determine if the device performs a resync operation after power-up software reboots and after each upgrade attempt.

When enabling Resync_On_Reset, the device introduces a random delay following the boot-up sequence before actually performing the reset. The delay is a random time up to the value specified in Resync_Random_Delay (in seconds). In a pool of these ATAs, all of which are simultaneously powered up, this introduces a spread in the times at which each unit initiates a resync request to the provisioning server. This feature can be useful in a large residential deployment, in the case of a regional power failures.

Triggers

The ATA allows you to resync at specific intervals or at a specific time.

Resyncing at Specific Intervals

The ATA is designed to resync with the provisioning server periodically. The resync interval is configured in Resync_Periodic (seconds). If this value is left empty, the device does not resync periodically.

The resync typically takes place when the voice lines are idle. In case a voice line is active when a resync is due, the ATA delays the resync procedure until the line becomes idle again. However, it waits no longer than Forced_Resync_Delay (seconds). A resync might cause configuration parameter values to change. This, in turn, causes a firmware reboot and terminates any voice connection active at the time of the resync.

If a resync operation fails because the ATA was unable to retrieve a profile from the server, if the downloaded file is corrupt, or an internal error occurs, the device tries to resync again after a time specified in Resync_Error_Retry_Delay (seconds). If Resync_Error_Retry_Delay is set to 0, the device does not try to resync again following a failed resync attempt.

When upgrading, if an upgrade fails, a retry is performed after Upgrade_Error_Retry_Delay seconds.

Two configurable parameters are available to conditionally trigger a resync: Resync_Trigger_1 and Resync_Trigger_2. Each of these parameters can be programmed with a conditional expression (which undergoes macro expansion). If the condition in any of these parameters evaluates to true, a resync operation is triggered, as though the periodic resync timer had expired.

The following example condition triggers a resync if Line 1 failed to register for more than 5 minutes (300 seconds), and at least 10 minutes (600 seconds) have elapsed since the last resync attempt.

```
$REGTMR1 gt 300 and $PRVTMR ge 600
```

Resyncing at a Specific Time

The Resync_At parameter allows you to resync at a specific time. This parameter uses the 24-hour format (hhmm) to specify the time.

To avoid simultaneously flooding the server with resync requests from multiple phones set to resync at the same time, the phone triggers the resync up to 10 minutes after the specified time.

For example, if you set the resync time to 1000 (10 a.m.), the phone triggers the resync anytime between 10:00 a.m. and 10:10 a.m.

By default, this feature is disabled. If the Resync_At parameter is provisioned, the Resync_Periodic parameter is ignored.

Configurable Schedules

You can configure schedules for periodic resyncs, and you can specify the retry intervals for resync and upgrade failures by using these provisioning parameters:

- Resync_Periodic
- Resync_Error_Retry_Delay
- Upgrade_Error_Retry_Delay

Each parameter accepts a single delay value (seconds). The new extended syntax allows for a comma-separated list of consecutive delay elements. The last element in the sequence is implicitly repeated forever. Below is an example:

```
Resync_Periodic=7200
Resync_Error_Retry_Delay=1800,3600,7200,14400
```

In the above example, the ATA periodically resyncs every two hours. In case of a resync failure, the device retries at these intervals: 30 minutes, 1 hour, 2 hours, 4 hours. It continues trying at 4-hour intervals until it successfully resyncs.

Optionally, you can use a plus sign to specify an additional numeric value that appends a random extra delay, as shown in this example:

```
Resync_Periodic=3600+600
Resync_Error_Retry_Delay=1800+300,3600+600,7200+900
```

In the above example, the device periodically resyncs every hour (plus an additional random delay of up to 10 minutes). In case of a resync failure, the device retries at these intervals: 30 minutes (plus up to 5 minutes). 1 hour (plus up to 10 minutes), 2 hours (plus up to 15 minutes). It continues trying at 2-hour intervals (plus up to 15 minutes) until it successfully resyncs.

Below is another example:

```
Upgrade_Error_Retry_Delay  =  1800,3600,7200,14400+3600
```

In this example, if a remote upgrade attempt fails, the device retries the upgrade in 30 minutes, then again after one more hour, then in two hours. If it still fails, it subsequently retries every four to five hours, until it succeeds.

Profile Rules

The ATA provides multiple remote configuration profile parameters (Profile_Rule*). This means that each resync operation can retrieve multiple files, potentially managed by different servers.

In the simplest scenario, the device resyncs periodically to a single profile on a central server, which updates all pertinent internal parameters. Alternatively, the profile can be split between different files. One file is common for all of these ATAs in a deployment, while a separate file is provided that is unique for each account. Encryption keys and certificate information could be supplied by still another profile, stored on a separate server.

Whenever a resync operation is due, the ATA evaluates the four Profile_Rule* parameters in sequence:

1. Profile_Rule
2. Profile_Rule_B
3. Profile_Rule_C
4. Profile_Rule_D

Each evaluation can result in a profile being retrieved from a remote provisioning server, possibly updating some number of internal parameters. If an evaluation fails, the resync sequence is interrupted, and is retried again from the beginning specified by the Resync_Error_Retry_Delay parameter (seconds). If all evaluations succeed, the device waits for the second specified by the Resync_Periodic parameter, and then performs a resync again.

The contents of each Profile_Rule* parameter consist of a set of alternatives. The alternatives are separated by the | (pipe) character. Each alternative consists of a conditional expression, an assignment expression, a profile URL, and any associated URL options. All these components are optional within each alternative. The following are the valid combinations, and the order in which they must appear, if present:

```
[ conditional-expr ] [ assignment-expr ] [[ options ] URL ]
```

Within each Profile_Rule* parameter, all of the alternatives except the last one must provide a conditional expression. This expression is evaluated and processed as follows:

1. Conditions are evaluated from left to right, until one is found that evaluates as true (or until one alternative is found with no conditional expression)
2. Any accompanying assignment expression is evaluated, if present
3. If a URL is specified as part of that alternative, an attempt is made to download the profile located at the specified URL, and update the internal parameters accordingly.

If all alternatives have conditional expressions, and none evaluates to true (or if the whole profile rule is empty), then the entire Profile_Rule* parameter is skipped, and the next profile rule parameter in the sequence is evaluated.

The following are some examples of valid programming for a single Profile_Rule* parameter.

The following example resyncs unconditionally to the profile at the specified URL, performing an HTTP GET request to the remote provisioning server.

```
http://remote.server.com/cisco/$MA.cfg
```

In the following example, the device resyncs to two different URLs, depending on the registration state of Line 1. In case of lost registration, the device performs an HTTP POST to a CGI script, transmitting the contents of the macro expanded GPP_A (which may provide additional information on the state of the device).

```
($REGTMR1 eq 0)? http://p.tel.com/has-reg.cfg
| [--post a] http://p.tel.com/lost-reg?
```

In the following example, the device resyncs to the same server, but provides additional information if a certificate is not installed in the unit (for legacy pre-2.0 units).

```
("$CCERT" eq "Installed")? https://p.tel.com/config?
| https://p.tel.com/config?cisco$MAU
```

In the following example, Line 1 is disabled until GPP_A is set equal to Provisioned through the first URL. Afterwards, it resyncs to the second URL.

```
("$A" ne "Provisioned")? (Line_Enable_1_ = "No";)! https://p.tel.com/init-
prov
| https://p.tel.com/configs
```

In the following example, the profile returned by the server is assumed to contain XML element tags that need to be remapped to proper parameter names by the aliases map stored in GPP_B.

```
[--alias b] https://p.tel.com/account/spa$MA.xml
```

A resync is typically considered unsuccessful if a requested profile is not received from the server. This default behavior can be overridden by the parameter Resync_Fails_On_FNF. If Resync_Fails_On_FNF is set to No, then the device accepts a file-not-found response from the server as a successful resync. The default value for Resync_Fails_On_FNF is Yes.

Report Rule

The ATA provides a mechanism for reporting its current internal configuration to the provisioning server. This is useful for development and debugging. The report syntax is similar to the Open format profile. All provisionable parameters are included, except for the values of passwords, keys, and the GPP_SA to GPP_SD parameters, which are not shown.

The Report_Rule parameter is evaluated like a profile rule parameter. In other words, it accepts a URL, optionally qualified with a bracketed expression. The URL specifies the target destination for the report and an encryption key can be included as an option.

The URL scheme can be TFTP, HTTP, or HTTPS. When using TFTP, the operation performed is TFTP PUT. In the case of HTTP and HTTPS, the operation performed is HTTP POST or HTTP PUT.

If an encryption key is specified, the report is encrypted using 256-bit AES in CBC mode. The encrypted report can be decrypted with the following OpenSSL (or equivalent) command:

```
openssl enc –d –aes-256-cbc –k secretphrase –in rep.xml.enc –out rep.xml
```

The following is an example of the corresponding Report_Rule configuration:

```
[ --key secretphrase ] http://prov.serv.net/spa/$MA/rep.xml.enc
```

Once the report rule is configured, an actual report can be generated and transmitted by sending the device a SIP NOTIFY message, with the Event: report type. The SIP NOTIFY request is handled like other SIP notifies, with the device requiring authentication from the requesting server before honoring the request to issue a report. Each SIP NOTIFY report request generates one attempt to transmit the report. Retries are not supported.

Deltas Report

In addition to reporting the current internal configuration to the provisioning server, the Report Rule has an option for triggering the reporting of configuration changes (deltas) to the server since the last resync, reboot, or upgrade.

The syntax of this option is:

Report Rule: [--delta] *URL*

Where *URL* is the path to where the report is stored on the server.

For example, to store delta configuration changes in a file with a name like SPA504G_<*MAC*>_<*serial#*>.xml, do one of the following:

- On the phone Web GUI, set the **Report Rule** field on the **Configuration Profile** page (Voice tab > Provisioning tab > Configuration Profile) to:

  ```
  [--delta] http://reportTargetServer/reportPath/$PN_$MA_$SN.xml
  ```

- Add the following to your provisioning file:

  ```
  <Report_Rule ua="na">[ --delta ]
  http://reportTargetServer/reportPath/$PN_$MA_$SN.xml
  </Report_Rule>
  ```

Status.xml Report

The Report Rule has an option to report the status.xml data if [--status] is specified in the report rule. The status report file path should be defined after the [--status] keyword, separated with a comma. If the [--status] keyword or the status report file path is missing, the phone will not report the status.xml data.

For example, if the following is configured:

[--status]http://*my_http_server*/config-525.xml

The phone will report the status file to http://*my_http_server*/config-525.xml.

If the following is configured:

[--delta]http://*my_http_server*/config-525.xml; [--status]http://*my_http_server*/status-525.xml

The phone will report the delta report to http://*my_http_server*/config-525.xml and the status report to http://*my_http_server*/status-525.xml.

Upgrade Rule

The ATA provides one configurable remote upgrade parameter, Upgrade_Rule. This parameter accepts a syntax similar to the profile rule parameters. URL options not supported for upgrades, but conditional expressions and assignment expressions can be used. If conditional expressions are used, the parameter can be populated with multiple alternatives, separated by the | character. The syntax for each alternative is as follows:

```
[ conditional-expr ] [ assignment-expr ] URL
```

As in the case of Profile_Rule* parameters, the Upgrade_Rule parameter evaluates each alternative until a conditional expression is satisfied or an alternative has no conditional expression. The accompanying assignment expression is evaluated, if specified. Then, an upgrade to the specified URL is attempted.

If the Upgrade_Rule contains a URL without a conditional expression, the device upgrades to the firmware image specified by the URL. Subsequently, it does not attempt to upgrade again until either the rule itself is modified or the effective combination of scheme + server + port + filepath is changed, following macro expansion and evaluation of the rule.

In order to attempt a firmware upgrade, the device disables audio at the start of the procedure, and reboots at the end of the procedure. For this reason, an upgrade driven by the contents of Upgrade_Rule is only automatically initiated by the device if any voice line is currently inactive.

For example,

```
http://p.tel.com/firmware/spa021025.bin
```

In this example, the Upgrade_Rule upgrades the firmware to the image stored at the indicated URL. The following is another example:

```
("$F" ne "beta-customer")? http://p.tel.com/firmware/
spa021025.bin
```

```
| http://p.tel.com/firmware/spa-test-0527s.bin
```

This example directs the unit to load one of two images, based on the contents of a general purpose parameter, GPP_F.

The device can enforce a downgrade limit with respect to firmware revision number. This can be useful as a customization option. If a valid firmware revision number is configured in the parameter Downgrade_Rev_Limit, the device rejects upgrade attempts for firmware versions earlier than the specified limit.

Data Types

The data types used with configuration profile parameters are as follows:

- Uns<n>—Unsigned n-bit value, where n = 8, 16, or 32. It can be specified in decimal or hex format such as 12 or 0x18 as long as the value can fit into n bits.
- Sig<n>—Signed n-bit value. It can be specified in decimal or hex format. Negative values must be preceded by a "-" sign. A + sign before positive value is optional.
- Str<n>—A generic string with up to n non-reserved characters.
- Float<n>—A floating point value with up to n decimal places.
- Time<n>—Time duration in seconds, with up to n decimal places. Extra decimal places specified are ignored.
- PwrLevel—Power level expressed in dBm with 1 decimal place, such as –13.5 or 1.5 (dBm).
- Bool—Boolean value of either "yes" or "no."
- {a,b,c,...}—A choice among a, b, c, ...
- IP—IP Address in the form of x.x.x.x, where x between 0 and 255. For example 10.1.2.100.
- Port—TCP/UDP Port number (0-65535). It can be specified in decimal of hex format.
- UserID—User ID as appeared in a URL; up to 63 characters.

- FQDN—Fully Qualified Domain Name, such as “sip.Cisco.com:5060”, or “109.12.14.12:12345”. It can contain up to 63 characters.
- Phone—A phone number string, such as 14081234567, *69, *72, 345678, or a generic URL such as 1234@10.10.10.100:5068, or jsmith@Cisco.com. It can contain up to 39 characters.
- ActCode—Activation code for a supplementary service, such as *69. It can contain up to 7 characters.
- PhTmplt—A phone number template. Each template may contain one or more patterns separated by a “,”. White space at the beginning of each pattern is ignored. “?” and “*” represent wildcard characters. To represent literally use %xx. For example, %2a represents *. It can contain up to 39 characters. Examples: “1408*, 1510*”, “1408123????, 555?1.”.
- RscTmplt—A template of SIP Response Status Code, such as “404, 5*”, “61?”, “407, 408, 487, 481”. It can contain up to 39 characters.
- CadScript—A mini-script that specifies the cadence parameters of a signal. Up to 127 characters. Syntax: $S_1[;S_2]$, where: $S_i=D_i(on_{i,1}/off_{i,1}[,on_{i,2}/off_{i,2}[,on_{i,3}/off_{i,3}[,on_{i,4}/off_{i,4}[,on_{i,5}/off_{i,5}[,on_{i,6}/off_{i,6}]]]]])$ and is known as a *section*, $on_{i,j}$ and $off_{i,j}$ are the on/off duration in seconds of a *segment* and i = 1 or 2, and j = 1 to 6. D_i is the total duration of the section in seconds. All durations can have up to three decimal places to provide 1 ms resolution. The wildcard character “*” stands for infinite duration. The segments within a section are played in order and repeated until the total duration is played.

  Example 1:

  ```
  60(2/4)


  Number of Cadence Sections = 1
  Cadence Section 1: Section Length = 60 s
  Number of Segments = 1
  Segment 1: On=2s, Off=4s


  Total Ring Length = 60s
  ```

  Example 2—Distinctive ring (short,short,short,long):

  ```
  60(.2/.2,.2/.2,.2/.2,1/4)
  ```

```
Number of Cadence Sections = 1
Cadence Section 1: Section Length = 60s
Number of Segments = 4
Segment 1: On=0.2s, Off=0.2s
Segment 2: On=0.2s, Off=0.2s
Segment 3: On=0.2s, Off=0.2s
Segment 4: On=1.0s, Off=4.0s

Total Ring Length = 60s
```

- FreqScript—A mini-script that specifics the frequency and level parameters of a tone. Up to 127 characters. Syntax: $F_1@L_1[,F_2@_{L2}[,F_3@L_3[,F_4@L_4[,F_5@L_5[,F_6@L_6]]]]]$, where F_1–F_6 are frequency in Hz (unsigned integers only) and L_1–L_6 are corresponding levels in dBm (with up to 1 decimal places). White spaces before and after the comma are allowed (but not recommended).

  Example 1—Call Waiting Tone:

```
440@-10

Number of Frequencies = 1
   Frequency 2 = 440 Hz at -10 dBm
```

  Example 2—Dial Tone:

```
350@-19,440@-19

Number of Frequencies = 2
   Frequency 1 = 350 Hz at -19 dBm
   Frequency 2 = 440 Hz at -19 dBm
```

- ToneScript—A mini-script that specifies the frequency, level and cadence parameters of a call progress tone. May contain up to 127 characters. Syntax: FreqScript;Z_1[;Z_2]. The section Z_1 is similar to the S_1 section in a CadScript except that each on/off segment is followed by a frequency

components parameter: $Z_1 = D_1(on_{i,1}/off_{i,1}/f_{i,1}[,on_{i,2}/off_{i,2}/f_{i,2}\ [,on_{i,3}/off_{i,3}/f_{i,3}\ [,on_{i,4}/off_{i,4}/f_{i,4}\ [,on_{i,5}/off_{i,5}/f_{i,5}\ [,on_{i,6}/off_{i,6}/f_{i,6}]]]]])$, where $fi,j = n_1[+n_2]+n_3[+n_4[+n_5[+n_6]]]]$ and $1 < n_k < 6$ indicates which of the frequency components given in the FreqScript are used in that segment; if more than one frequency component is used in a segment, the components are summed together.

Example 1—Dial tone:

```
350@-19,440@-19;10(*/0/1+2)


Number of Frequencies = 2
Frequency 1 = 350 Hz at -19 dBm
Frequency 2 = 440 Hz at -19 dBm
Number of Cadence Sections = 1
Cadence Section 1: Section Length = 10 s
Number of Segments = 1
Segment 1: On=forever, with Frequencies 1 and 2


Total Tone Length = 10s
```

Example 2—Stutter tone:

```
350@-19,440@-19;2(.1/.1/1+2);10(*/0/1+2)


Number of Frequencies = 2
Frequency 1 = 350 Hz at -19 dBm
Frequency 2 = 440 Hz at -19 dBm
Number of Cadence Sections = 2
Cadence Section 1: Section Length = 2s
Number of Segments = 1
Segment 1: On=0.1s, Off=0.1s with Frequencies 1 and 2
Cadence Section 2: Section Length = 10s
Number of Segments = 1
Segment 1: On=forever, with Frequencies 1 and 2
```

```
Total Tone Length = 12s
```

Example 3—SIT tone:

```
985@-16,1428@-16,1777@-16;20(.380/0/1,.380/0/2,.380/0/
3,0/4/0)
Number of Frequencies = 3
Frequency 1 = 985 Hz at -16 dBm
Frequency 2 = 1428 Hz at -16 dBm
Frequency 3 = 1777 Hz at -16 dBm
Number of Cadence Sections = 1
Cadence Section 1: Section Length = 20s
Number of Segments = 4
Segment 1: On=0.38s, Off=0s, with Frequency 1
Segment 2: On=0.38s, Off=0s, with Frequency 2
Segment 3: On=0.38s, Off=0s, with Frequency 3
Segment 4: On=0s, Off=4s, with no frequency components
Total Tone Length = 20s
```

- ProvisioningRuleSyntax—Scripting syntax used to define configuration resync and firmware upgrade rules.
- DialPlanScript—Scripting syntax used to specify Line 1 and Line 2 dial plans.

NOTE

- `<Par Name>` represents a configuration parameter name. In a profile, the corresponding tag is formed by replacing the space with an underscore "_", such as **Par_Name**.
- An empty default value field implies an empty string < "" >.
- The ATA continues to use the last configured values for tags that are not present in a given profile.
- Templates are compared in the order given. The first, *not the closest*, match is selected. The parameter name must match exactly.

- If more than one definition for a parameter is given in a profile, the last such definition in the file is the one that takes effect in the ATA.
- A parameter specification with an empty parameter value forces the parameter back to its default value. To specify an empty string instead, use the empty string "" as the parameter value.

3

In-House Preprovisioning and Provisioning Servers

Cisco IP Telephony devicesSPA100 and SPA200 Series ATAs, other than RC units, are preprovisioned by the service provider with a profile. That pre-provision profile can range from a a limited set of parameters that resynchronizes the ATA to another profile with a complete set of parameters delivered by remote server. Or, it can be a complete set of parameters. By default, the ATA resynchronizes on power up and at intervals configured in the profile. When the user connects the ATA at the customer premises, the device downloads the updated profile and any firmware updates.

This process of preprovisioning, deployment, and remote provisioning can be accomplished many ways. This chapter describes the features and functionality available when preprovisioning these ATAs in-house and provisioning them remotely:

Server Preparation and Software Tools

The examples presented in this chapter require the availability of one or more servers. These servers can be installed and run on a local PC:

- TFTP (UDP port 69)
- syslog (UDP port 514)
- HTTP (TCP port 80)
- HTTPS (TCP port 443).

To troubleshoot server configuration, it is helpful to install clients for each type of server on a separate server machine. This establishes proper server operation, independent of the interaction with these ATAs.

Cisco also recommends the installation of the following software tools:

- To generate configuration profiles, it is useful to install the open source gzip compression utility.
- For profile encryption and HTTPS operations, install the open source OpenSSL software package.
- To test the dynamic generation of profiles and one-step remote provisioning using HTTPS, a scripting language with CGI scripting support, such as open source Perl language tools, is recommended.
- To verify secure exchanges between provisioning servers and these ATAs, install an Ethernet packet sniffer (such as the freely downloadable Ethereal/Wireshark). Capture an Ethernet packet trace of the interaction between the ATA and the provisioning server by running the packet sniffer on a PC that is connected to a switch with port mirroring enabled. For HTTPS transactions, you can use the ssldump utility.

In-House Device Preprovisioning

With the Cisco factory default configuration, an ATA automatically tries to resync to a profile on a TFTP server. The information regarding the profile and TFTP server configured for preprovisioning is delivered to the device by a managed DHCP server on a LAN. The service provider connects each new ATA that LAN and the ATA automatically resyncs to the local TFTP server, initializing its internal state in preparation for deployment. This preprovisioning profile typically includes the URL of a remote provisioning server that will keep the device updated after it is deployed and connected to the customer network.

The preprovisioned device barcode can be scanned to record its MAC address or serial number before the ATA is shipped to the customer. This information can be used to create the profile to which the ATA will resynchronize.

Upon receiving the ATA, the customer connects it to the broadband link. On power-up the ATA contacts the provisioning server through the URL configured through preprovisioning to for its resync and updates the profile and firmware as necessary.

Provisioning Server Setup

This section describes setup requirements for provisioning an ATA by using various servers and different scenarios. For testing purposes and for the purposes of this document, provisioning servers are installed and run on a local PC. Also, generally available software tools are useful for provisioning these ATAs.

TFTP Provisioning

These ATAs support TFTP for both provisioning resync and firmware upgrade operations. When devices are deployed remotely, HTTP is recommended for provisioning as it offers greater reliability, given NAT and router protection mechanisms. TFTP is useful for the in-house preprovisioning of a large number of un-provisioned devices. See **In-House Device Preprovisioning, page 39** for more information.

The ATA is able to obtain a TFTP server IP address directly from the DHCP server through DHCP option 66. If a Profile_Rule is configured with the filepath of that TFTP server, the device downloads its profile from the TFTP server when it is connected to a LAN and powered up.

The Profile_Rule provided with the factory default configuration is */device*.cfg. For example, on a SPA962 the filename is spa962.cfg. If the device has the factory default profile, when powered up it resyncs to this file on the local TFTP server specified by DHCP option 66. (The filepath is relative to the TFTP server virtual root directory.)

Remote Endpoint Control and NAT

The ATA accesses the Internet through a router by using network address translation (NAT). For enhanced security, the router might attempt to block unauthorized incoming packets by implementing symmetric NAT (a packet filtering strategy that severely restricts the packets that are allowed to enter the protected network from the Internet). For this reason, remote provisioning by using TFTP is not recommended.

Voice over IP can co-exist with NAT only when some form of NAT traversal is provided. Configure Simple Traversal of UDP through NAT (STUN). This option requires that the user have (1) a dynamic external (public) IP address from your service, (2) a computer running STUN server software, and (3) an edge device with an asymmetric NAT mechanism.

HTTP Provisioning

The ATA behaves like a browser requesting web pages from a remote Internet site. This provides a reliable means of reaching the provisioning server, even when a customer router implements symmetric NAT or other protection mechanisms. HTTP and HTTPS work more reliably than TFTP in remote deployments, especially when the deployed units are connected behind residential firewalls or NAT-enabled routers.

Basic HTTP-based provisioning relies on the HTTP GET method for retrieving configuration profiles. Typically, a configuration file is created for each deployed ATA, and these files are stored within a HTTP server directory. When the server receives the GET request, it simply returns the file specified in the GET request header.

Alternatively, the requested URL can invoke a CGI script (using the GET method). The configuration profile is generated dynamically by querying a customer database and producing the profile on-the-fly.

When CGI handles resync requests, the ATA can use the HTTP POST method to request the resync configuration data. The device can be configured to convey certain status and identification information to the server within the body of the HTTP POST request. The server uses this information to generate a desired response configuration profile, or store the status information for later analysis and tracking.

As part of both GET and POST requests, the ATA automatically includes basic identifying information in the request header, in the User-Agent field. This information conveys the manufacturer, product name, current firmware version, and product serial number of the device.

For example, the following example is the User-Agent request field from a SPA962:

```
User-Agent: cisco/SPA-962-2.0.5 (88012BA01234)
```

When the ATA is configured to resync to a configuration profile by using HTTP, it is recommended that the profile be encrypted to protect confidential information. The ATA supports 256-bit AES in CBC mode to decrypt profiles. Encrypted profiles downloaded by the ATA by using HTTP avoid the danger of exposing confidential information contained in the configuration profile. This resync mode produces a lower computational load on the provisioning server when compared to using HTTPS.

HTTPS Provisioning

For increased security managing remotely deployed units, the ATA supports HTTPS for provisioning. Each ATA carries a unique SLL Client Certificate (and associated private key), in addition to a Sipura CA server root certificate. The latter allows the ATA to recognize authorized provisioning servers, and reject non-authorized servers. On the other hand, the client certificate allows the provisioning server to identify the individual device that issues the request.

For a service provider to manage deployment by using HTTPS, a server certificate must be generated for each provisioning server to which an ATA resyncs by using HTTPS. The server certificate must be signed by the Cisco Server CA Root Key, whose certificate is carried by all deployed units. To obtain a signed server certificate, the service provider must forward a certificate signing request to Cisco, which signs and returns the server certificate for installation on the provisioning server.

The provisioning server certificate must contain the Common Name (CN) field, and the FQDN of the host running the server in the subject. It might optionally contain information following the host FQDN, separated by a slash (/) character. The following examples are of CN entries that are accepted as valid by the ATA:

```
CN=sprov.callme.com
CN=pv.telco.net/mailto:admin@telco.net
CN=prof.voice.com/info@voice.com
```

In addition to verifying the server certificate, the ATA tests the server IP address against a DNS lookup of the server name specified in the server certificate.

A certificate signing request can be generated by using the OpenSSL utility. The following example shows the **openssl** command that produces a 1024-bit RSA public/private key pair and a certificate signing request:

```
openssl req –new –out provserver.csr
```

This command generates the server private key in **privkey.pem** and a corresponding certificate signing request in **provserver.csr**. The service provider keeps the **privkey.pem** secret and submits **provserver.csr** to Cisco for signing. Upon receiving the **provserver.csr** file Cisco generates **provserver.crt**, the signed server certificate.

Cisco also provides a Sipura CA Client Root Certificate to the service provider. This root certificate certifies the authenticity of the client certificate carried by each ATA.

The unique client certificate offered by each device during an HTTPS session carries identifying information embedded in its subject field. This information can be made available by the HTTPS server to a CGI script invoked to handle secure requests. In particular, the certificate subject indicates the unit product name (OU element), MAC address (S element), and serial number (L element). The following example from a SPA962 client certificate subject field shows these elements:

```
OU=SPA-962, L=88012BA01234, S=000e08abcdef
```

Units manufactured before firmware 2.0.x do not contain individual SSL client certificates. When these units are upgraded to a firmware release in the 2.0.x tree, they become capable of connecting to a secure server using HTTPS, but are only able to supply a generic client certificate if requested to do so by the server. This generic certificate contains the following information in the identifying fields:

```
OU=cisco.com, L=ciscogeneric, S=ciscogeneric
```

To determine if an ATA carries an individualized certificate, use the $CCERT provisioning macro variable. The variable value expands to either Installed or Not Installed, according to the presence or absence of a unique client certificate. In the case of a generic certificate, it is possible to obtain the serial number of the unit from the HTTP request header in the User-Agent field.

HTTPS servers can be configured to request SSL certificates from connecting clients. If enabled, the server can verify the client certificate by using the Sipura CA Client Root Certificate supplied by Cisco. It can then provide the certificate information to a CGI for further processing.

The location for storing certificates might vary. For example, on an Apache installation the file paths for storing the provisioning server–signed certificate, its associated private key, and the Sipura CA client root certificate are as follows:

```
# Server Certificate:
SSLCertificateFile /etc/httpd/conf/provserver.crt

# Server Private Key:
SSLCertificateKeyFile /etc/httpd/conf/provserver.key
```

```
# Certificate Authority (CA):
SSLCACertificateFile /etc/httpd/conf/spacroot.crt
```

Refer to the documentation provided for a HTTPS server for specific information.

Firmware release 2.0.6 and higher supports the following cipher suites for SSL connection to a server by using HTTPS.

Table 1 Cipher Suites Supported for Connecting to an HTTPS Server

Numeric Code	Cipher Suite
0x0039	TLS_DHE_RSA_WITH_AES_256_CBC_SHA
0x0035	TLS_RSA_WITH_AES_256_CBC_SHA
0x0033	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
0x002f	TLS_RSA_WITH_AES_128_CBC_SHA
0x0005	TLS_RSA_WITH_RC4_128_SHA
0x0004	TLS_RSA_WITH_RC4_128_MD5
0x0062	TLS_RSA_EXPORT1024_WITH_RC4_56_SHA
0x0060	TLS_RSA_EXPORT1024_WITH_RC4_56_MD5
0x0003	TLS_RSA_EXPORT_WITH_RC4_40_MD5

Redundant Provisioning Servers

The provisioning server can be specified as an IP address or as a fully qualified domain name (FQDN). The use of a FQDN facilitates the deployment of redundant provisioning servers. When the provisioning server is identified through a FQDN, the ATA attempts to resolve the FQDN to an IP address through DNS. Only DNS A-records are supported for provisioning; DNS SRV address resolution is not available for provisioning. The ATA continues to process A-records until a server responds. If no server associated with the A-records responds, the ATA logs an error to the syslog server.

Syslog Server

If a syslog server is configured on the ATA (using the <Syslog_Server> or <Debug_Server> parameters), the resync and upgrade operations log messages to the syslog server. A message can be generated at the start of a remote file request (configuration profile or firmware load), and at the conclusion of the operation (indicating either success or failure).

The logged messages themselves are configured in the following parameters:

- For profile resync:
  - Log_Resync_Request_Msg
  - Log_Resync_Success_Msg
  - Log_Resync_Failure_Msg
- For firmware upgrades:
  - Log_Upgrade_Request_Msg
  - Log_Upgrade_Success_Msg
  - Log_Upgrade_Failure_Msg

These parameters are macro expanded into the actual syslog messages.

As indicated in the lower half of the diagram, a Cisco Client Certificate Root Authority signs each unique certificate. The corresponding root certificate is made available to service providers for client authentication purposes.

4

Provisioning Examples

This chapter provides example procedures for transferring configuration profiles between the ATA and the provisioning server:

- **Basic Resync, page 46**
- **Secure HTTPS Resync, page 53**
- **Profile Management, page 61**

For information about creating configuration profiles, refer to **Chapter 2, "Creating XML Provisioning Scripts."**

Basic Resync

This section demonstrates the basic resync functionality of these ATAs.

TFTP Resync

The ATA supports multiple network protocols for retrieving configuration profiles. The most basic profile transfer protocol is TFTP (RFC1350). TFTP, widely used for the provisioning of network devices within private LAN networks. Although not recommended for the deployment of remote endpoints across the Internet, it can be convenient for deployment within small organizations, for in-house preprovisioning, and for development and testing. See **"In-House Device Preprovisioning" section on page 39** for more information on in-house preprovisioning. In this exercise, a profile is modified after downloading a file from a TFTP server.

Exercise

STEP 1 Within a LAN environment connect a PC and an ATA to a hub, switch, or small router.

STEP 2 Connect an analog phone to the Phone 1 port of the ATA.

STEP 3 On the PC, install and activate a TFTP server.

STEP 4 Using a text editor, create a configuration profile that sets the value for GPP_A to 12345678 as shown in the example.

```
<flat-profile>
  <GPP_A> 12345678
  </GPP_A>
</flat-profile>
```

STEP 5 Save the profile with the name `basic.txt` in the root directory of the TFTP server.

You can verify that the TFTP server is properly configured by requesting the basic.txt file by using a TFTP client other than the ATA. Preferably, use a TFTP client that is running on a separate host from the provisioning server.

STEP 6 Using an analog phone, obtain the IP address of the ATA (IVR menu ****** 110 #**).

If the configuration has been modified since it was manufactured, perform factory reset on the phone by using the IVR RESET option (******** 73738**#**).

STEP 7 Open the PC web browser on the admin/advanced configuration page. For example, if the IP address of the phone is 192.168.1.100:

```
http://192.168.1.100/admin/advanced
```

STEP 8 Select the Provisioning tab, and inspect the values of the general purpose parameters GPP_A through GPP_P. These should be empty.

STEP 9 Resync the test ATA to the `basic.txt` configuration profile by opening the resync URL in a web browser window.

Assuming the IP address of the TFTP server is 192.168.1.200 the command should be similar to this example:

```
http://192.168.1.100/admin/resync?tftp://192.168.1.200/basic.txt
```

When ATA receives this command, the device at address 192.168.1.100 requests the file `basic.txt` from the TFTP server at IP address 192.168.1.200. It then parses the downloaded file and updates the GPP_A parameter with the value 12345678.

STEP 10 Verify that the parameter was correctly updated by refreshing the admin/advanced page on the PC web browser and selecting the Provisioning tab on that page.

The GPP_A parameter should now contain the value 12345678.

Logging with syslog

The ATA sends a syslog message to the designated syslog server when the device is about to resync to a provisioning server and after the resync has either completed or failed. This server is identified in the web server administration (admin/advanced, System tab, Syslog_Server parameter). Configure the syslog server IP address into the device and observe the messages generated during the remaining exercises.

Exercise

STEP 1 Install and activate a syslog server on the local PC.

STEP 2 Program the PC IP address into the Syslog_Server parameter of the profile and submit the change:

```
<Syslog_Server ua="na">192.168.1.210</Syslog_Server>
```

STEP 3 Click the **System** tab and enter the value of your local syslog server into the Syslog_Server parameter.

STEP 4 Repeat the resync operation as described in the **TFTP Resync** exercise.

The device generates two syslog messages during the resync. The first indicates that a request is in progress. The second marks success or failure of the resync.

STEP 5 Verify that your syslog server received messages similar to the following:

```
SPA-962 00:0e:08:ab:cd:ef -- Requesting resync tftp://192.168.1.200/basic.txt
SPA-962 00:0e:08:ab:cd:ef -- Successful resync tftp://192.168.1.200/basic.txt
```

Detailed messages are available by programming a Debug_Server parameter (instead of the Syslog_Server parameter) with the IP address of the syslog server, and setting the Debug_Level to a value between 0 and 3 (3 being the most verbose):

```
<Debug_Server ua="na">192.168.1.210</Debug_Server>
<Debug_Level ua="na">3</Debug_Level>
```

The contents of these messages can be configured by using the following parameters:

- Log_Resync_Request_Msg
- Log_Resync_Success_Msg
- Log_Resync_Failure_Msg.

If any of these parameters are cleared, the corresponding syslog message is not generated.

Automatic Device Resync

A device can resync periodically to the provisioning server to ensure that any profile changes made on the server are propagated to the endpoint device (as opposed to sending an explicit resync request to the endpoint).

To cause the ATA to periodically resync to a server, a configuration profile URL is defined by using the Profile_Rule parameter, and a resync period is defined by using the Resync_Periodic parameter.

Exercise

STEP 1 Using a web browser, open the admin/advanced page Provisioning tab.

STEP 2 Define the Profile_Rule parameter. The example assumes a TFTP server IP address of 192.168.1.200:

```
<Profile_Rule ua="na">tftp://192.168.1.200/basic.txt</Profile_Rule>
```

STEP 3 In the Resync_Periodic parameter enter a small value for testing, such as **30** seconds:

```
<Resync_Periodic ua="na">30</Resync_Periodic>
```

STEP 4 Click **Submit all Changes**.

With the new parameter settings, the ATA resyncs to the configuration file specified by the URL twice a minute.

STEP 5 Observe the resulting messages in the syslog trace (as described in the **Logging with syslog** section).

STEP 6 Ensure that the Resync_On_Reset parameter is set to **yes**:

```
<Resync_On_Reset ua="na">Yes</Resync_On_Reset>
```

STEP 7 Power cycle the ATA to force it to resync to the provisioning server.

If the resync operation fails for any reason, such as if the server is not responding, the unit waits the number of seconds configured in Resync_Error_Retry_Delay before attempting to resync again. If Resync_Error_Retry_Delay is zero, the ATA does not try to resync after a failed resync attempt.

STEP 8 (Optional) Set the value of Resync_Error_Retry_Delay is set to a small number, such as **30**:

```
<Resync_Error_Retry_Delay ua="na">30</Resync_Error_Retry_Delay>
```

STEP 9 Disable the TFTP server, and observe the results in the syslog output.

Unique Profiles, Macro Expansion, and HTTP

In a deployment where each ATA must be configured with distinct values for some parameters, such as User_ID or Display_Name, the service provider can create a unique profile for each deployed device and host those profiles on a provisioning server. Each ATA, in turn, must be configured to resync to its own profile according a predetermined profile naming convention.

The profile URL syntax can include identifying information specific to each ATA, such as MAC address or serial number, by using the macro expansion of built-in variables. Macro expansion eliminates the need to specify these values in multiple locations within each profile.

A profile rule undergoes macro expansion before being applied to the ATA. The macro expansion controls a number of values, for example:

- $MA expands to the unit 12-digit MAC address (using lower case hex digits). For example, 000e08abcdef.
- $SN expands to the unit serial number. For example, 88012BA01234.

Other values can be macro expanded in this way, including all the general purpose parameters, (GPP_A through GPP_P). An example of this process can be seen in the **TFTP Resync** section. Macro expansion is not limited to the URL file name, but can also be applied to any portion of the profile rule parameter. These parameters are referenced as $A through $P. For a complete list of variables available for macro expansion, see the **"Macro Expansion Variables" section on page 77**.

In this exercise, a profile specific to an ATA is provisioned on a TFTP server.

Exercise

STEP 1 Obtain the MAC address of the ATA from its product label. (The MAC address is the number, using numbers and lower–case hex digits, such as 000e08aabbcc.

STEP 2 Copy the `basic.txt` configuration file (described in the **TFTP Resync** exercise) to a new file named **`spa`***`_macaddress`***`.cfg`** (replacing *`macaddress`* with the MAC address of the ATA). For example:

```
spa_000e08abcdef.cfg
```

STEP 3 Move the new file in the virtual root directory of the TFTP server.

STEP 4 Open the admin/advanced page Provisioning tab.

STEP 5 Enter `tftp://192.168.1.200/spa$MA.cfg` in the Profile_Rule parameter:

```
<Profile_Rule ua="na">
  tftp://192.168.1.200/spa$MA.cfg
</Profile_Rule>
```

STEP 6 Click **Submit All Changes**. This causes an immediate reboot and resync.

When the next resync occurs, the ATA retrieves the new file by expanding the $MA macro expression into its MAC address.

HTTP GET Resync

HTTP provides a more reliable resync mechanism than TFTP because HTTP establishes a TCP connection and TFTP uses the less reliable UDP. In addition, HTTP servers offer improved filtering and logging features compared to TFTP servers.

On the client side, the ATA does not require any special configuration setting on the server to be able to resync by using HTTP. The Profile_Rule parameter syntax for using HTTP with the GET method is similar to the syntax used for TFTP. If a standard web browser can retrieve a profile from a your HTTP server, the ATA should be able to do so as well.

Exercise

STEP 1 Install an HTTP server on the local PC or other accessible host. (The open source Apache server can be downloaded from the Internet.)

STEP 2 Copy the `basic.txt` configuration profile (described in the **TFTP Resync** exercise) onto the virtual root directory of the installed server.

STEP 3 Verify proper server installation (and file access to basic.txt) by accessing the profile by using a web browser.

STEP 4 Modify the Profile_Rule of the test ATA to point to the HTTP server in place of the TFTP server, so as to download its profile periodically.

For example, assuming the HTTP server is at 192.168.1.300, enter the following value:

```
<Profile_Rule ua="na">
http://192.168.1.200/basic.txt
</Profile_Rule>
```

STEP 5 Click **Submit All Changes**. This causes an immediate reboot and resync.

STEP 6 Observe the syslog messages sent by the ATA. The periodic resyncs should now be obtaining the profile from the HTTP server.

STEP 7 In the HTTP server logs, observe how information identifying the test ATA appears in the log of user agents.

This should include the manufacturer, product name, current firmware version, and serial number.

URL Resolution by using Macro Expansion

Subdirectories with multiple profiles on the server is a convenient method for managing a large number of deployed devices. The profile URL can contain:

- A provisioning server name or an explicit IP address. If the profile identifies the provisioning server by name, the ATA performs a DNS lookup to resolve the name.
- A non-standard server port specified in the URL by using the standard syntax : *port* following the server name.
- The subdirectory of the server virtual root directory where the profile is stored, specified by using standard URL notation and managed by macro expansion.

For example, the following Profile_Rule requests the profile spa962.cfg, in the server subdirectory /cisco/config, from the TFTP server running on host prov.telco.com listening for a connection on port 6900:

```
<Profile_Rule ua="na">
/tftp://prov.telco.com:6900/cisco/config/spa962.cfg
</Profile_Rule>
```

A profile for each ATA can be identified in a general purpose parameter, with its value referred within a common profile rule by using macro expansion.

For example, assume GPP_B is defined as `Dj6Lmp23Q`.

The Profile_Rule has the value:

```
tftp://prov.telco.com/cisco/$B/$MA.cfg
```

When the device resyncs and the macros are expanded, the ATA with a MAC address of 000e08012345 requests the profile with the name that contains the device MAC address at the following URL:

```
tftp://prov.telco.com/cisco/Dj6Lmp23Q/000e08012345.cfg
```

Secure HTTPS Resync

This section demonstrates the mechanisms available on the ATA for resyncing by using a secure communication process. It includes the following topics:

- **Basic HTTPS Resync, page 53**
- **HTTPS With Client Certificate Authentication, page 56**
- **HTTPS Client Filtering and Dynamic Content, page 57**

Basic HTTPS Resync

HTTPS adds SSL to HTTP for remote provisioning so that the:

- ATA can authenticate the provisioning server
- provisioning server can authenticate the ATA
- confidentiality of information exchanged between the ATA and the provisioning server is ensured.

SSL generates and exchanges secret (symmetric) keys for each connection between the ATA and the server, using public/private key pairs preinstalled in the ATA and the provisioning server.

On the client side, the ATA does not require any special configuration setting on the server to be able to resync using HTTPS. The Profile_Rule parameter syntax for using HTTPS with the GET method is similar to the syntax used for HTTP or TFTP. If a standard web browser can retrieve a profile from a your HTTPS server, the ATA should be able to do so as well.

In addition to installing a HTTPS server, a SSL server certificate signed by Cisco must be installed on the provisioning server. The devices cannot resync to a server using HTTPS unless the server supplies a Cisco-signed server certificate. Instructions for creating signed SSL Certificates for SPA Voice products can be found at https://supportforums.cisco.com/docs/DOC-9852.

Exercise

STEP 1 Install an HTTPS server on a host whose IP address is known to the network DNS server through normal hostname translation.

The open source Apache server can be configured to operate as an HTTPS server when installed with the open source mod_ssl package.

STEP 2 Generate a server Certificate Signing Request for the server. For this step, you might need to install the open source OpenSSL package or equivalent software. If using OpenSSL, the command to generate the basic CSR file is as follows:

```
openssl req –new –out provserver.csr
```

This command generates a public/private key pair, which is saved in the privkey.pem file.

STEP 3 Submit the CSR file (provserver.csr) to Cisco for signing. (See https://supportforums.cisco.com/docs/DOC-9852 for more information.) A signed server certificate is returned (provserver.cert) along with a Sipura CA Client Root Certificate, spacroot.cert.

STEP 4 Store the signed server certificate, the private key pair file, and the client root certificate in the appropriate locations on the server.

In the case of an Apache installation on Linux, these locations are typically as follows:

```
# Server Certificate:
SSLCertificateFile /etc/httpd/conf/provserver.cert
# Server Private Key:
SSLCertificateKeyFile /etc/httpd/conf/pivkey.pem
# Certificate Authority:
SSLCACertificateFile /etc/httpd/conf/spacroot.cert
```

STEP 5 Restart the server.

STEP 6 Copy the `basic.txt` configuration file (described in the **TFTP Resync** exercise) onto the virtual root directory of the HTTPS server.

STEP 7 Verify proper server operation by downloading basic.txt from the HTTPS server by using a standard browser from the local PC.

STEP 8 Inspect the server certificate supplied by the server.

The browser probably does not recognize it as valid unless the browser has been preconfigured to accept Cisco as a root CA. However, these ATAs expect the certificate to be signed this way.

Modify the Profile_Rule of the test device to contain a reference to the HTTPS server, for example:

```
<Profile_Rule ua="na">
https://my.server.com/basic.txt
</Profile_Rule>
```

This example assumes the name of the HTTPS server is `my.server.com`.

STEP 9 Click **Submit All Changes**.

STEP 10 Observe the syslog trace sent by the ATA.

The syslog message should indicate that the resync obtained the profile from the HTTPS server.

STEP 11 (Optional) Use an Ethernet protocol analyzer on the ATA subnet to verify that the packets are encrypted.

In this exercise, client certificate verification was not enabled. The connection between ATA and server is encrypted. However, the transfer is not secure because any client can connect to the server and request the file, given knowledge of the file name and directory location. For secure resync, the server must also authenticate the client, as demonstrated in the exercise described in the **HTTPS With Client Certificate Authentication** section.

HTTPS With Client Certificate Authentication

In the factory default configuration, the server does not request a SSL client certificate from a client. Transfer of the profile is not secure because any client can connect to the server and request the profile. You can edit the configuration to enable client authentication; the server requires a client certificate to authenticate the ATA before accepting a connection request.

Because of this, the resync operation cannot be independently tested by using a browser lacking the proper credentials. The SSL key exchange within the HTTPS connection between the test ATA and the server can be observed using the ssldump utility. The utility trace shows the interaction between client and server.

NOTE Both basic and digest authentication are supported on SPA500 Series phones running firmware version 7.4.9c and higher.

Exercise

STEP 1 Enable client certificate authentication on the HTTPS server.

STEP 2 In Apache (v.2), set the following in the server configuration file:

```
SSLVerifyClient  require
```

Also ensure that the spacroot.cert has been stored as shown in the **Basic HTTPS Resync** exercise.

STEP 3 Restart the HTTPS server and observe the syslog trace from the ATA.

Each resync to the server now performs symmetric authentication, so that both the server certificate and the client certificate are verified before the profile is transferred.

STEP 4 Use ssldump to capture a resync connection between the ATA and the HTTPS server.

If client certificate verification is properly enabled on the server, the ssldump trace shows the symmetric exchange of certificates (first server-to-client, then client-to-server) before the encrypted packets containing the profile.

With client authentication enabled, only a ATA with a MAC address matching a valid client certificate can request the profile from the provisioning server. A request from an ordinary browser or other unauthorized device is rejected by the server.

HTTPS Client Filtering and Dynamic Content

If the HTTPS server is configured to require a client certificate, then the information in the certificate identifies the resyncing ATA and supplies it with the correct configuration information.

The HTTPS server makes the certificate information available to CGI scripts (or compiled CGI programs) invoked as part of the resync request. For the purpose of illustration, this exercise uses the open source Perl scripting language, and assumes that Apache (v.2) is used as the HTTPS server.

Exercise

STEP 1 Install Perl on the host running the HTTPS server.

STEP 2 Generate the following Perl reflector script:

```
#!/usr/bin/perl -wT
use strict;
print "Content-Type: text/plain\n\n";
print "<flat-profile><GPP_D>";

print "OU=$ENV{'SSL_CLIENT_I_DN_OU'},\n";
print "L=$ENV{'SSL_CLIENT_I_DN_L'},\n";
print "S=$ENV{'SSL_CLIENT_I_DN_S'}\n";
print "</GPP_D></flat-profile>";
```

STEP 3 Save this file with the file name reflect.pl, with executable permission (chmod 755 on Linux), in the CGI scripts directory of the HTTPS server.

STEP 4 Verify accessibility of CGI scripts on the server (as in /cgi-bin/...).

STEP 5 Modify the Profile_Rule on the test device to resync to the reflector script, as in the following example:

```
https://prov.server.com/cgi-bin/reflect.pl?
```

STEP 6 Click **Submit All Changes**.

STEP 7 Observe the syslog trace to ensure a successful resync.

STEP 8 Open the admin/advanced page, Provisioning tab.

STEP 9 Verify that the GPP_D parameter contains the information captured by the script.

This information contains the product name, MAC address, and serial number if the test device carries a unique certificate from the manufacturer, or else generic strings if it is a unit manufactured before firmware release 2.0.

A similar script could be used to determine information about the resyncing device and then provide it with appropriate configuration parameter values.

HTTPS Certificates

The ATA provides a reliable and secure provisioning strategy based on HTTPS requests from the device to the provisioning server. Both a server certificate and a client certificate are used to authenticate the ATA to the server and the server to the ATA.

To use HTTPS, you must generate a Certificate Signing Request (CSR) and submit it to Cisco. Cisco generates a certificate for installation on the provisioning server. The ATA accepts the certificate when it seeks to establish an HTTPS connection with the provisioning server.

How HTTPS Works

HTTPS encrypts the communication between a client and a server, protecting the message contents from other network devices. The encryption method for the body of the communication between a client and a server is based on symmetric key cryptography. With symmetric key cryptography, a single secret key is shared by a client and a server over a secure channel protected by Public/Private key encryption.

Messages encrypted by the secret key can only be decrypted using the same key. HTTPS supports a wide range of symmetric encryption algorithms. The ATA implements up to 256-bit symmetric encryption, using the American Encryption Standard (AES), in addition to 128-bit RC4.

HTTPS also provides for the authentication of a server and a client engaged in a secure transaction. This feature ensures that a provisioning server and an individual client cannot be spoofed by other devices on the network. This is an essential capability in the context of remote endpoint provisioning.

Server and client authentication is performed by using public/private key encryption with a certificate that contains the public key. Text that is encrypted with a public key can be decrypted only by its corresponding private key (and vice versa). The ATA supports the RSA algorithm for public/private key cryptography.

SSL Server Certificates

Each secure provisioning server is issued a SSL server certificate, directly signed by Cisco. The firmware running on the ATA recognizes only a Cisco certificate as valid. When a client connects to a server by using HTTPS, it rejects any server certificate that is not signed by Cisco.

This mechanism protects the service provider from unauthorized access to the ATA, or any attempt to spoof the provisioning server. Without such protection, an attacker might be able to reprovision the ATA, to gain configuration information, or to use a different VoIP service.

Client Certificates

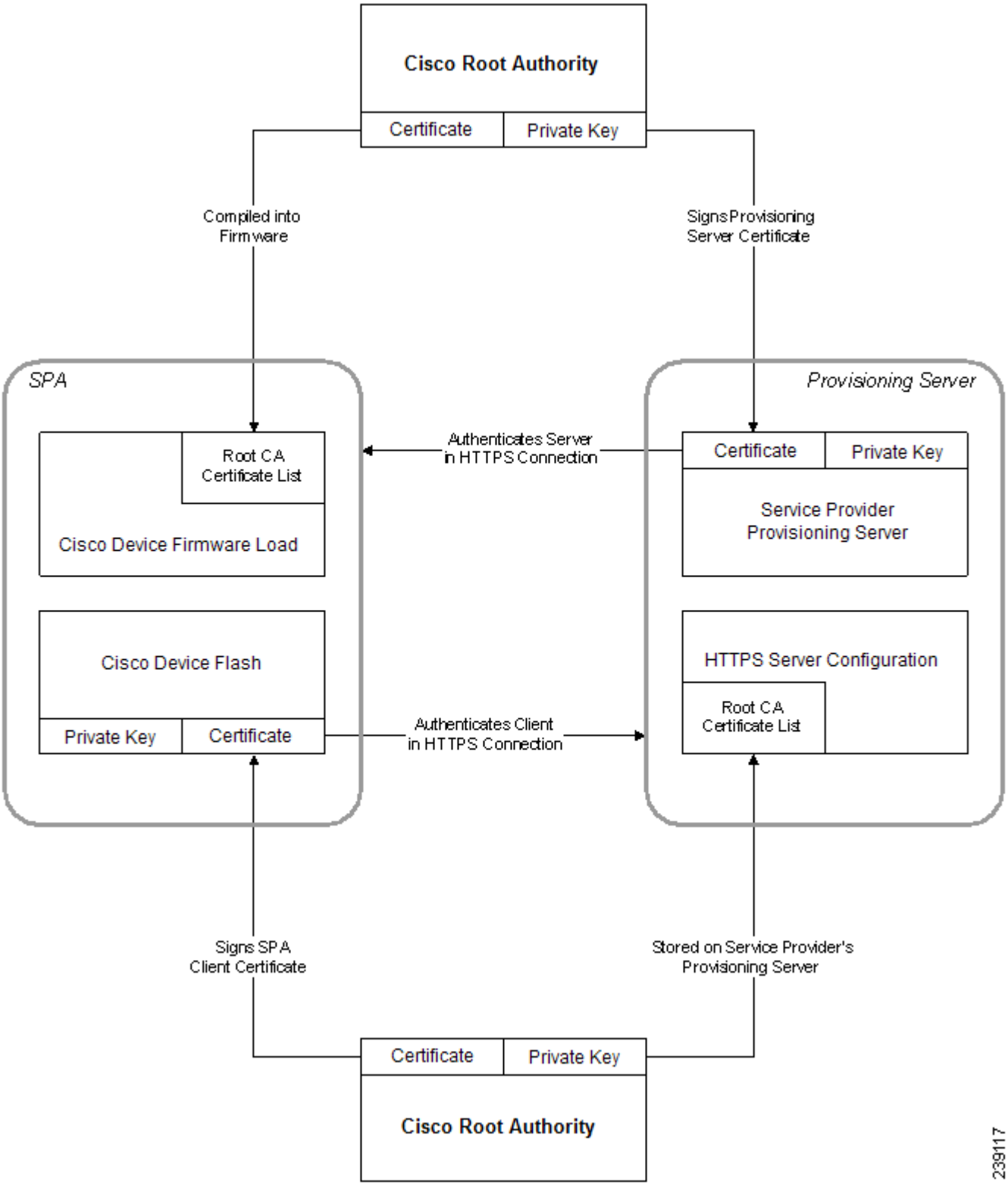
In addition to a direct attack on an ATA, an attacker might attempt to contact a provisioning server by using a standard web browser or another HTTPS client to obtain the configuration profile from the provisioning server. To prevent this kind of attack, each ATA also carries a unique client certificate, signed by Cisco, including identifying information about each individual endpoint. A certificate authority root certificate capable of authenticating the device client certificate is given to each service provider. This authentication path allows the provisioning server to reject unauthorized requests for configuration profiles.

Certificate Structure

The combination of a server certificate and a client certificate ensures secure communication between a remote ATA and its provisioning server. The **"Certificate Authority Flow"** figure illustrates the relationship and placement of certificates, public/private key pairs, and signing root authorities, among the Cisco client, the provisioning server, and the certification authority.

The upper half of the diagram shows the Provisioning Server Root Authority that is used to sign the individual provisioning server certificate. The corresponding root certificate is compiled into the firmware, allowing the ATA to authenticate authorized provisioning servers.

Certificate Authority Flow



Profile Management

This section demonstrates the formation of configuration profiles in preparation for downloading. To explain the functionality, TFTP from a local PC is used as the resync method, although HTTP or HTTPS can be used as well.

Open Profile gzip Compression

A configuration profile in XML format can become quite large if all parameters are individually specified by the profile. To reduce the load on the provisioning server, the ATA supports compression of the XML file, by using the deflate compression format supported by the gzip utility (RFC 1951).

NOTE Compression must precede encryption for the ATA to recognize a compressed and encrypted XML profile.

For integration into customized back-end provisioning server solutions, the open source zlib compression library can be used in place of the standalone gzip utility to perform the profile compression. However, the ATA expects the file to contain a valid gzip header.

Exercise

STEP 1 Install gzip on the local PC.

STEP 2 Compress the `basic.txt` configuration profile (described in the **TFTP Resync** exercise) by invoking gzip from the command line:

```
gzip basic.txt
```

This generates the deflated file `basic.txt.gz`.

STEP 3 Save the `basic.txt.gz` file in the TFTP server virtual root directory.

STEP 4 Modify the Profile_Rule on the test device to resync to the deflated file in place of the original XML file, as shown in the following example:

```
tftp://192.168.1.200/basic.txt.gz
```

STEP 5 Click **Submit All Changes**.

STEP 6 Observe the syslog trace from the ATA.

Upon resync, the new file is downloaded by the ATA and used to update its parameters.

Profile Encryption by using OpenSSL

A compressed or uncompressed profile can be encrypted (however, a file must be compressed before it is encrypted). This is useful when the confidentiality of the profile information is of particular concern, such as when using TFTP or HTTP for communication between the ATA and the provisioning server.

Exercise

STEP 1 Install OpenSSL on a local PC. This might require that the OpenSSL application be recompiled to enable AES.

STEP 2 Using the `basic.txt` configuration file (described in the **TFTP Resync** exercise), generate an encrypted file with the following command:

```
>openssl enc –aes-256-cbc –k MyOwnSecret –in basic.txt –out basic.cfg
```

The compressed basic.txt.gz file created in **Open Profile gzip Compression** also can be used, because the XML profile can be both compressed and encrypted.

STEP 3 Store the encrypted basic.cfg file in the TFTP server virtual root directory.

STEP 4 Modify the Profile_Rule on the test device to resync to the encrypted file in place of the original XML file. The encryption key is made known to the ATA with the following URL option:

```
[--key MyOwnSecret ] tftp://192.168.1.200/basic.cfg
```

STEP 5 Click **Submit All Changes**.

STEP 6 Observe the syslog trace from the ATA.

On resync, the new file is downloaded by the ATA and is used to update its parameters.

Partitioned Profiles

An ATA downloads multiple separate profiles during each resync. This allows managing different kinds of profile information on separate servers and maintaining common configuration parameter values separate from account specific values.

Exercise

STEP 1 Create a new XML profile, basic2.txt, that specifies a value for a parameter that makes it distinct from the earlier exercises. For instance, to the basic.txt profile you can add the following:

```
<GPP_B>ABCD</GPP_B>
```

STEP 2 Store the basic2.txt profile in the virtual root directory of the TFTP server.

STEP 3 Leave the first profile rule from the earlier exercises in the folder, but configure the second profile rule (Profile_Rule_B) to point to the new file:

```
<Profile_Rule_B ua="na">tftp://192.168.1.200/basic2.txt
</Profile_Rule_B>
```

STEP 4 Click **Submit All Changes**.

The ATA now resyncs to both the first and second profiles, in that order, whenever a resync operation is due.

STEP 5 Observe the syslog trace to confirm the expected behavior.

Parameter Name Aliases

When generating an XML profile for the ATA, it might be convenient to assign names to certain configuration parameters that are different from the canonical names recognized by the ATA. For example, a customer account database might generate XML element tags for a customer telephone number and SIP registration password with names, such as SIP-number and SIP-password. These names can be mapped to the canonical names (User_ID_1_ and Password_1_) before being applied to Line 1.

In many instances, the back-end provisioning solution used by the service provider can perform this mapping. However, the ATA itself can remap the parameter names internally. To do this, an alias map is defined and stored in one of the general purpose provisioning parameters. Then, the profile rule which invokes the resync is directed to remap the non-canonical XML elements as specified by the alias map.

Exercise

STEP 1 Generate a profile named customer.XML containing the proprietary customer-account XML form indicated in the following example:

```
<customer-account>
  <SIP-number> 17775551234</SIP-number>
  <SIP-password> 512835907884</SIP-password>
</customer-account>
```

STEP 2 Store the profile in the TFTP server virtual root directory.

STEP 3 Open the web interface on the device to the admin/advanced page, Provisioning tab, and edit GPP_A to contain the alias map (do not enter new lines through the web interface, instead simply enter each alias consecutively):

```
/customer-account/SIP-number = /flat-profile/User_ID_1_ ;
/customer-account/SIP-password = /flat-profile/Password_1_ ;
```

STEP 4 Edit the Profile_Rule to point to the new XML profile, and specify the alias map as a URL option, as follows:

```
[--alias a ] tftp://192.168.1.200/customer.xml
```

STEP 5 Click **Submit All Changes**.

When the ATA resyncs, it receives the XML profile, remaps the elements, as indicated by the alias map, and populates the User_ID_1_ and Password_1_ parameters.

STEP 6 View the Line 1 tab to verify the new configuration.

NOTE The ATA supports alias remapping of a limited number of parameters. It is not meant to rename all parameters in its configuration.

5

Provisioning Parameters

This chapter describes the Provisioning parameters that can be used in configuration profile scripts. It includes the following sections:

The Provisioning parameters described in this chapter are recognized by the these ATAs beginning with firmware release 2.0.6 and higher unless otherwise indicated.

Delta Configuration Report

When the report rule is set, a SPA phone reports the phone profile to the server upon boot-up or receiving a report SIP NOTIFY message. By default the entire profile is reported.

Password or encryption key–related parameter values are not reported to the server:

```
<Admin_Password> IP Phone admin password
  <User_Password>   IP Phone user password
<PPPOE_Login_Password>
< VPN_Password>
<Access_Password_N>  Camera access password for each camera
profile
```

```
<Password_N>   Sip account user password for each SIP
extension
<SRTP_Private_Key_N> SRTP private key password for each SIP
extension
<Auth_Page_Password_N> Auth page password for each SIP
extension
<PIN_Code>                  BluePhone Pin code
<Directory_Password>  Broadsoft directory
<Password>   LDAP password
```

Deltas Report

SPA phones running firmware version 7.4.9c or higher can report deltas to the server if the **-delta** option is specified in the report rule. For example:

```
Report Rule: [--delta] http://report.com/delta$$MAC.xml
```

NOTE The double hyphen (--) required.

The *main profile* supports all provisionable parameters. Parameters in the main profile include the WiFi profile parameters. The **-delta** option only applies to the main profile. The deltas can be triggered by changes to the phone parameters entered in the LCD screen, the Web GUI, a SIP event, or remote provisioning. The personal address book, call history, Bluetooth profiles, and so forth are not in the main profile and are not reported.

The delta report is generated if the phone detects changes since the last resync, reboot, or upgrade. The report is done in asynchronous manner (with a random delay) and is sent only when the phone is idle. (The phone is considered idle when there is no active call or key press.)

Status.xml Report

SPA phones running firmware version 7.5.3 or higher have an option to report the status.xml data if [--status] is specified in the report rule. The status report file path should be defined after the [--status] keyword, separated with a comma. If the [--status] keyword or the status report file path is missing, the phone will not report the status.xml data.

For example, if the following is configured:

[--status]http://*my_http_server*/config-525.xml

The phone will report the status file to http://*my_http_server*/config-525.xml.

If the following is configured:

[--delta]http://*my_http_server*/config-525.xml; [--status]http://*my_http_server*/status-525.xml

The phone will report the delta report to http://*my_http_server*/config-525.xml and the status report to http://*my_http_server*/status-525.xml.

Report Content

An administrator can define the content [**--content**] that is included in the report in the report rule. When **--content** *path* is defined, the main profile, address book and call history are reported to server. Where **p** reports the main profile parameters, **a** reports address book information, **h** reports the call history. This option is only available for UC320W.

Configuration Profile Parameters

The following table defines the function and usage of each parameter in the Configuration Profile Parameters section under the Provisioning tab.

Parameter Name	Description and Default Value
Provision_Enable	Controls all resync actions independently of firmware upgrade actions. Set to yes to enable remote provisioning. The default value is Yes.
Resync_On_Reset	Triggers a resync after every reboot except for reboots caused by parameter updates and firmware upgrades. The default value is Yes.
Resync_Random_Delay	Prevents an overload of the provisioning server when a large number of devices power-on simultaneously and attempt initial configuration. This delay is effective only on the initial configuration attempt, following a device power-on or reset. The parameter is the maximum time interval that the device waits before making contact with the provisioning server. The actual delay is a pseudo-random number between zero and this value. This parameter is in units of 20 seconds; the default value of 3 represents 60 seconds. This feature is disabled when this parameter is set to zero. The default value is 2 (40 seconds).
Resync At (SPA500 series phones)	The hour and minutes (HHmm) that the device resyncs with the provisioning server. The default value is empty. If the value is invalid, the parameter is ignored. If this parameter is set with a valid value, the Resync_Periodic parameter is ignored.

Parameter Name	Description and Default Value
Resync_At_Random_Delay (firmware v7.4.9c and higher)	Prevents an overload of the provisioning server when a large number of devices power-on simultaneously. To avoid flooding resync requests to the server from multiple phones, the phone resyncs in the range between the hours and minutes, and the hours and minutes plus the random delay (hhmm, hhmm+random_delay). For example, if the random delay = (Resync_At_Random_Delay + 30)/60 minutes. The input value in seconds is converted to minutes, rounding up to the next minute to calculate the final random_delay interval. This feature is disabled when this parameter is set to zero. The default value is 600 seconds (10 minutes). If the parameter value is set to less than 600, the default value is used.
Resync_Periodic	The time interval between periodic resyncs with the provisioning server. The associated resync timer is active only after the first successful sync with the server. Set this parameter to zero to disable periodic resyncing. The default value is 3600 seconds.

Parameter Name	Description and Default Value
Resync_Error_Retry_Delay	Resync retry interval (in seconds) applied in case of resync failure. The device has an error retry timer that activates if the previous attempt to sync with the provisioning server fails. The device waits to contact the server again until the timer counts down to zero. This parameter is the value that is initially loaded into the error retry timer. If this parameter is set to zero, the device immediately retries to sync with the provisioning server following a failed attempt. The default value is 3600 seconds.
Forced_Resync_Delay	Maximum delay (in seconds) the ATA waits before performing a resync. The device does not resync while one of its phone lines is active. Because a resync can take several seconds, it is desirable to wait until the device has been idle for an extended period before resyncing. This allows a user to make calls in succession without interruption. The device has a timer that begins counting down when all of its lines become idle. This parameter is the initial value of the counter. Resync events are delayed until this counter decrements to zero. The default value is 14,400 seconds.
Resync_From_SIP	Enables a resync to be triggered via a SIP NOTIFY message. The default value is Yes.
Resync_After_Upgrade_Attempt	Triggers a resync after every firmware upgrade attempt. The default value is Yes.

Parameter Name	Description and Default Value
Resync_Trigger_1, Resync_Trigger_2	Configurable resync trigger conditions. A resync is triggered when the logic equation in these parameters evaluates to TRUE. The default value is (empty).
Resync_Fails_On_FNF	Determines whether a file-not-found response from the provisioning server constitutes a successful or a failed resync. A failed resync activates the error resync timer. The default value is Yes.
Profile_Rule	This parameter is a profile script that evaluates to the provisioning resync command. The command specifies the protocol (TFTP, HTTP, or HTTPS) and an associated URL. If the command is not specified, TFTP is assumed, and the address of the TFTP server is obtained through DHCP option 66. In the URL, either the IP address or the FQDN of the server can be specified. The file name can have macros, such as $MA, which expands to the device MAC address. The default value is /spa$PSN.cfg.
Profile_Rule_B, Profile_Rule_C, Profile_Rule_D	Defines second, third, and fourth resync commands and associated profile URLs. These profile scripts are executed sequentially after the primary Profile Rule resync operation has completed. If a resync is triggered and Profile Rule is blank, Profile Rule B, C, and D are still evaluated and executed. The default value is (empty).
Log_Resync_Request_Msg	This parameter contains the message that is sent to the syslog server at the start of a resync attempt. The default value is $PN $MAC – Requesting resync $SCHEME:// $SERVIP:$PORT$PATH.

Parameter Name	Description and Default Value
Log_Resync_Success_Msg	The syslog message that is issued upon successful completion of a resync attempt. The default value is $PN $MAC – Successful resync $SCHEME:// $SERVIP:$PORT$PATH -- $ERR.
Log_Resync_Failure_Msg	The syslog message that is issued after a failed resync attempt. The default value is $PN $MAC – Resync failed: $ERR.

Parameter Name	Description and Default Value
Report_Rule	The target URL to which configuration reports are sent. This parameter has the same syntax as the Profile_Rule parameter, and resolves to a TCP/IP command with an associated URL. A configuration report is generated in response to an authenticated SIP NOTIFY message, with Event: report. The report is an XML file containing the name and value of all the device parameters. This parameter may optionally contain an encryption key. For example: [--key $K] tftp://ps.callhome.net/$MA/rep.xml.enc Additionally, this parameter can trigger the reporting of configuration changes (deltas) to the server since the last resync, reboot, or upgrade using the --delta option. For example, to store delta configuration changes in a file with a name like SPA504G_<*MAC*>_<*serial#*>.xml, add the following to your provisioning file: [--delta] http://reportTargetServer/reportPath/$PN_$MA_$SN.xml </Report_Rule> The default value is (empty).

Firmware Upgrade Parameters

The following table defines the function and usage of each parameter in the Firmware Upgrade section of the Provisioning tab.

Parameter Name	Description and Default Value
Upgrade_Enable	Enables firmware upgrade operations independently of resync actions. The default value is Yes.
Upgrade_Error_Retry_Delay	The upgrade retry interval (in seconds) applied in case of upgrade failure. The device has a firmware upgrade error timer that activates after a failed firmware upgrade attempt. The timer is initialized with the value in this parameter. The next firmware upgrade attempt occurs when this timer counts down to zero. The default value is 3600 seconds.
Downgrade_Rev_Limit	Enforces a lower limit on the acceptable version number during a firmware upgrade or downgrade. The device does not complete a firmware upgrade operation unless the firmware version is greater than or equal to this parameter. The default value is (empty).
Upgrade_Rule	This parameter is a firmware upgrade script with the same syntax as Profile_Rule. Defines upgrade conditions and associated firmware URLs. The default value is (empty).
Log_Upgrade_Request_Msg	The syslog message that is issued at the start of a firmware upgrade attempt. The default value is $PN $MAC -- Requesting upgrade $SCHEME://$SERVIP:$PORT$PATH.

Parameter Name	Description and Default Value
Log_Upgrade_Success_Msg	The syslog message that is issued after a firmware upgrade attempt completes successfully. The default value is $PN $MAC -- Successful upgrade $SCHEME://$SERVIP:$PORT$PATH -- $ERR.
Log_Upgrade_Failure_Msg	The syslog message that is issued after a failed firmware upgrade attempt. The default value is $PN $MAC -- Upgrade failed: $ERR.

General Purpose Parameters

The following table defines the function and usage of each parameter in the General Purpose Parameters section of the Provisioning tab.

Parameter Name	Description and Default Value
GPP_SA, GPP_SB, GPP_SC, GPP_SD	Special purpose provisioning parameters, designed to hold encryption keys and passwords. To ensure the integrity of the encryption mechanism, these parameters must be kept secret. Therefore these parameters are not displayed on the device configuration web page, and they are not included in the configuration report sent in response to a SIP NOTIFY command. Note that these parameters are not available on the SPA500 Series phones. The default value is (empty).

Parameter Name	Description and Default Value
GPP_A through GPP_P	General purpose provisioning parameters. These parameters can be used as variables in provisioning and upgrade rules. They are referenced by prepending the variable name with a '$' character, such as $GPP_A. The default value is (empty).

Macro Expansion Variables

Certain macro variables are recognized within the following provisioning parameters:

- Profile_Rule
- Profile_Rule_*
- Resync_Trigger_*
- Log_Resync_*
- Upgrade_Rule
- Log_Upgrade_*
- GPP_* (under specific conditions)

Within these parameters, syntax types, such as $NAME or $(NAME), are recognized and expanded.

Macro variable substrings can be specified with the notation $(NAME:p) and $(NAME:p:q), where p and q are non-negative integers (available in revision 2.0.11 and above). The resulting macro expansion is the substring starting at character offset p, with length q (or else till end-of-string if q is not specified). For example, if GPP_A contains ABCDEF, then $(A:2) expands to CDEF, and $(A:2:3) expands to CDE.

An unrecognized name is not translated, and the $NAME or $(NAME) form remains unchanged in the parameter value after expansion.

Parameter Name	Description and Default Value
$	The form $$ expands to a single $ character.
A through P	Replaced by the contents of the general purpose parameters GPP_A through GPP_P.
SA through SD	Replaced by the contents of the special purpose parameters GPP_SA through GPP_SD. These parameters are meant to hold keys or passwords used in provisioning. Note that $SA through $SD are only recognized as arguments to the optional resync URL qualifier **--key**, as in the following example: [--key $SA] http://ps.callme.com/profiles/abcdefg.cfg These variables are not expanded outside of this limited context. Note that these variables are not available on the SPA500 Series phones.
MA	MAC address using lower case hex digits, for example, 000e08aabbcc.
MAU	MAC address using upper case hex digits, for example 000E08AABBCC.
MAC	MAC address using lower case hex digits, and colons to separate hex digit pairs, for example 00:0e:08:aa:bb:cc.
PN	Product Name, for example SPA962.
PSN	Product Series Number, for example 962.
SN	Serial Number string, for example 88012BA01234.
CCERT	SSL Client Certificate status: Installed or Not Installed.
IP	IP address of the ATA within its local subnet, for example 192.168.1.100.
EXTIP	External IP of the ATA, as seen on the Internet, for example 66.43.16.52.
SWVER	Software version string, for example 2.0.6(b).
HWVER	Hardware version string, for example 1.88.1.

Parameter Name	Description and Default Value
PRVST	Provisioning State, a numeric string: -1 = explicit resync request, 0 = power-up resync, 1 = periodic resync, 2 = resync failed, retry attempt
UPGST	Upgrade State, a numeric string: 1 = first upgrade attempt, 2 = upgrade failed, retry attempt
UPGERR	Result message (ERR) of previous upgrade attempt, for example http_get failed.
PRVTMR	Seconds since last resync attempt.
UPGTMR	Seconds since last upgrade attempt.
REGTMR1	Seconds since Line 1 lost registration with SIP server.
REGTMR2	Seconds since Line 2 lost registration with SIP server.
UPGCOND	Legacy macro name, always expands to true in firmware rev 2.0.6 and above.
SCHEME	File access scheme, one of TFTP, HTTP, or HTTPS, as obtained after parsing resync or upgrade URL.
METH	Deprecated alias for SCHEME, do not use.
SERV	Request target server host name, as obtained after parsing resync or upgrade URL.
SERVIP	Request target server IP address, as obtained after parsing resync or upgrade URL, possibly following DNS lookup.
PORT	Request target UDP/TCP port, as obtained after parsing resync or upgrade URL.
PATH	Request target file path, as obtained after parsing resync or upgrade URL.
ERR	Result message of resync or upgrade attempt. Only useful in generating result syslog messages. The value is preserved in the UPGERR variable in the case of upgrade attempts.

Parameter Name	Description and Default Value
UID1	The contents of the Line 1 User_ID configuration parameter (Firmware 2.0.11 and above).
UID2	The contents of the Line 2 User_ID configuration parameter (Firmware 2.0.11 and above).
ISCUST	Value=1 if unit is customized, 0 otherwise; customization status viewable on WebUI Info page.

Internal Error Codes

The ATA defines a number of internal error codes (X00–X99) to facilitate configuration in providing finer control over the behavior of the unit under certain error conditions.

Parameter Name	Description and Default Value
X00	Transport layer (or ICMP) error when sending a SIP request.
X20	SIP request times out while waiting for a response.
X40	General SIP protocol error (for example, unacceptable codec in SDP in 200 and ACK messages, or times out while waiting for ACK).
X60	Dialed number invalid according to given dial plan.

6

Voice Parameters

This chapter describes the voice parameters for the ATAs.

A note about parameter numbering:

Certain types of parameters apply to multiple elements, such as users and lines. In the configuration file, the parameter name is appended with a number, such as <Line_Enable_1_> and <Line_Enable_2_>. To understand this numbering system, use the following key:

- 1—User 1 or Line 1 (PHONE1 port, all models)
- 2—User 2 or Line 2 (PHONE2 port, SPA100 Series only)

<Restricted_Access_Domains>	This feature is not currently used.
<Enable_Web_Admin_Access>	This feature enables or disables access to the configuration utility from devices that are connected via the ETHERNET (LAN) port. Default setting: Yes (enabled)
<IVR_Admin_Password>	Password for the administrator to manage the ATA by using the built-in IVR through a connected phone.
<Network_Startup_Delay>	The number of seconds of delay between restarting the voice module and initializing network interface. Default setting: 3

<DNS_Query_TTL_Ignore<	In DNS packages, the server will suggest a TTL value to the client; if this parameter is set to yes, the value from the server will be ignored. Default setting: yes

<Syslog_Server>	Specify the syslog server name and port. This feature specifies the server for logging ATA system information and critical events. If both Debug Server and Syslog Server are specified, Syslog messages are also logged to the Debug Server. Default setting: blank
<Debug_Server>	The debug server name and port. This feature specifies the server for logging debug information. The level of detailed output depends on the debug level parameter setting. Default setting: blank
<Debug_Level>	Determines the level of debug information that will be generated. Select 0, 1, 2, 3 or 3+Router from the drop-down list. The higher the debug level, the more debug information will be generated. Level 0 means that no information will be collected. Levels 1, 2 & 3 generate messages related to the voice ports only. Level 3+Router generates debug content for both voice and router components. Default setting: 3

<Provision_Enable>	Controls all resync actions independently of firmware upgrade actions. Set to yes to enable remote provisioning. Default setting: yes
<Resync_On_Reset>	Triggers a resync after every reboot except for reboots caused by parameter updates and firmware upgrades. Default setting: yes
<Resync_Random_Delay>	The maximum value for a random time interval that the ATA waits before making its initial contact with the provisioning server. This delay is effective only on the initial configuration attempt following power-on or reset. The delay is a pseudo-random number between zero and this value. This parameter is in units of 20 seconds; the default value of 2 represents 40 seconds. This feature is disabled when this parameter is set to zero. This feature can be used to prevent an overload of the provisioning server when a large number of devices power-on simultaneously. Default setting: 2 (40 seconds)

<Resync_At_HHmm>	The time of day when the device tries to resync. The resync is performed each day. Used in conjunction with the Resync At Random Delay. Default setting: blank
<Resync_At_Random_Delay>	Used in conjunction with the Resync At (HHmm) setting, this parameter sets a range of possible values for the resync delay. The system randomly chooses a value from this range and waits the specified number of seconds before attempting to resync. This feature is intended to prevent the network jam that would occur if all resynchronizing devices began the resync at the exact same time of day. Default setting: 600
<Resync_Periodic>	The time interval between periodic resyncs with the provisioning server. The associated resync timer is active only after the first successful synchronization with the server. Setting this parameter to zero disables periodic resynchronization. Default setting: 3600 seconds
<Resync_Error_Retry_Delay>	Resync retry interval (in seconds) applied in case of resync failure. The ATA has an error retry timer that activates if the previous attempt to sync with the provisioning server fails. The ATA waits to contact the server again until the timer counts down to zero. This parameter is the value that is initially loaded into the error retry timer. If this parameter is set to zero, the ATA immediately retries to sync with the provisioning server following a failed attempt. Default setting: 3600 seconds

<Forced_Resync_Delay>	Maximum delay (in seconds) that the ATA waits before performing a resync. The ATA does not resync while one of its lines is active. Because a resync can take several seconds, it is desirable to wait until the ATA has been idle for an extended period before resynchronizing. This allows a user to make calls in succession without interruption. The ATA has a timer that begins counting down when all of its lines become idle. This parameter is the initial value of the counter. Resync events are delayed until this counter decrements to zero. Default setting: 14400 seconds
<Resync_From_SIP>	Enables a resync to be triggered via a SIP NOTIFY message. Default setting: yes
<Resync_After_Upgrade_Attempt>	Triggers a resync after every firmware upgrade attempt. Default setting: yes
<Resync_Trigger_1> <Resync_Trigger_2>	Configurable resync trigger conditions. A resync is triggered when the logic equation in these parameters evaluates to TRUE. Default setting: blank
<Resync_Fails_On_FNF>	Determines whether a file-not-found response from the provisioning server constitutes a successful or a failed resync. A failed resync activates the error resync timer. Default setting: yes
<Profile_Rule>	This parameter is a profile script that evaluates to the provisioning resync command. The command is a TCP/IP operation and an associated URL. The TCP/IP operation can be TFTP, HTTP, or HTTPS. If the command is not specified, TFTP is assumed, and the address of the TFTP server is obtained through DHCP option 66. In the URL, either the IP address or the FQDN of the server can be specified. The file name can have macros, such as $MA, which expands to the ATA MAC address. Default setting: /spa$PSN.cfg

<Profile_Rule_B:> <Profile_Rule_C:> <Profile_Rule_D>	Defines second, third, and fourth resync commands and associated profile URLs. These profile scripts are executed sequentially after the primary Profile Rule resync operation has completed. If a resync is triggered and Profile Rule is blank, Profile Rule B, C, and D are still evaluated and executed. Default setting: blank
<Log_Resync_Request_Msg>	This parameter contains the message that is sent to the Syslog server at the start of a resync attempt. Default setting: $PN $MAC -- Requesting resync $SCHEME:// $SERVIP:$PORT$PATH
<Log_Resync_Success_Msg>	Syslog message issued upon successful completion of a resync attempt. Default setting: $PN $MAC -- Successful resync $SCHEME:// $SERVIP:$PORT$PATH
<Log_Resync_Failure_Msg>	Syslog message issued after a failed resync attempt. Default setting: $PN $MAC -- Resync failed: $ERR
<Report_Rule>	The target URL to which configuration reports are sent. This parameter has the same syntax as the Profile_Rule parameter, and resolves to a TCP/IP command with an associated URL. A configuration report is generated in response to an authenticated SIP NOTIFY message, with Event: report. The report is an XML file containing the name and value of all the device parameters. This parameter may optionally contain an encryption key. For example: [--key $K] tftp://ps.callhome.net/$MA/rep.xml.enc Default setting: blank
<Upgrade_Enable>	Determines whether or not firmware upgrade operations can occur independently of resync actions. Default setting: yes

<Upgrade_Error_Retry_Delay>	The upgrade retry interval (in seconds) applied in case of upgrade failure. The ATA has a firmware upgrade error timer that activates after a failed firmware upgrade attempt. The timer is initialized with the value in this parameter. The next firmware upgrade attempt occurs when this timer counts down to zero. Default setting: 3600 seconds
<Downgrade_Rev_Limit>	Enforces a lower limit on the acceptable version number during a firmware upgrade or downgrade. The ATA does not complete a firmware upgrade operation unless the firmware version is greater than or equal to this parameter. Default setting: blank
<Upgrade_Rule>	This parameter is a firmware upgrade script with the same syntax as Profile_Rule. Defines upgrade conditions and associated firmware URLs. Default setting: blank
<Log_Upgrade_Request_Msg>	Syslog message issued at the start of a firmware upgrade attempt. Default setting: $PN $MAC -- Requesting upgrade $SCHEME:/ /$SERVIP:$PORT$PATH
<Log_Upgrade_Success_Msg>	Syslog message issued after a firmware upgrade attempt completes successfully. Default setting: $PN $MAC -- Successful upgrade $SCHEME:/ /$SERVIP:$PORT$PATH -- $ERR
<Log_Upgrade_Failure_Msg>	Syslog message issued after a failed firmware upgrade attempt. Default setting: $PN $MAC -- Upgrade failed: $ERR
<License_Keys>	This field is not currently used.

<Custom_CA_URL>	The URL of a file location for a custom Certificate Authority (CA) certificate. Either the IP address or the FQDN of the server can be specified. The file name can have macros, such as $MA, which expands to the ATA MAC address. Default setting: null

<GPP_A> to <GPP_P>	General purpose provisioning parameters. These parameters can be used as variables in provisioning and upgrade rules. They are referenced by prepending the variable name with a '$' character, such as $GPP_A. Default setting: blank
<GPP_SA> to <GPP_SD>	The two-letter upper-case macro names SA through SD identify GPP_SA through GPP_SD as a special case when used as arguments of the key URL option.

<Max_Forward>	The maximum times a call can be forwarded. The valid range is from 1 to 255. Default setting: 70
<Max_Redirection>	Number of times an invite can be redirected to avoid an infinite loop. Default setting: 5.
<Max_Auth_>	The maximum number of times (from 0 to 255) a request may be challenged. Default setting: 2
<SIP_User_Agent_Name>	The User-Agent header used in outbound requests. If empty, the header is not included. Macro expansion of $A to $D corresponding to GPP_A to GPP_D allowed. Default setting: $VERSION
<SIP_Server_Name>	The server header used in responses to inbound responses. Default setting: $VERSION
<SIP_Reg_User_Agent_Name>	The User-Agent name to be used in a REGISTER request. If this value is not specified, the SIP User Agent Name parameter is also used for the REGISTER request. Default setting: blank
<SIP_Accept_Language>	Accept-Language header used. There is no default (this indicates that the ATA does not include this header) If empty, the header is not included. Default setting: blank

<DTMF_Relay_MIME_Type>	The MIME Type used in a SIP INFO message to signal a DTMF event. Default setting: application/dtmf-relay.
<Hook_Flash_MIME_Type>	The MIME Type used in a SIP INFO message to signal a hook flash event. Default setting: application/hook-flash
<Remove_Last_Reg>	Determines whether or not the ATA removes the last registration before submitting a new one, if the value is different. Select yes to remove the last registration, or select no to omit this step. Default setting: no
<Use_Compact_Header>	Determines whether or not the ATA uses compact SIP headers in outbound SIP messages. Select yes or no from the drop-down list. Select yes to use compact SIP headers in outbound SIP messages. Select no to use normal SIP headers. If inbound SIP requests contain compact headers, the ATA reuses the same compact headers when generating the response regardless the settings of the Use Compact Header parameter. If inbound SIP requests contain normal headers, the ATA substitutes those headers with compact headers (if defined by RFC 261) if Use Compact Header parameter is set to yes. Default setting: no
<Escape_Display_Name>	Determines whether or not the Display Name is private. Select yes if you want the ATA to enclose the string (configured in the Display Name) in a pair of double quotes for outbound SIP messages. If the display name includes " or \, these will be escaped to \" and \\ within the double quotes. Otherwise, select no. Default setting: no
<RFC_2543_Call_Hold>	Configures the type of call hold: a:sendonly or 0.0.0.0. Do not use the 0.0.0.0 syntax in a HOLD SDP; use the a:sendonly syntax. Default setting: no
<Mark_all_AVT_Packets>	Select yes if you want all AVT tone packets (encoded for redundancy) to have the marker bit set for each DTMF event. Select no to have the marker bit set only for the first packet. Default setting: yes

<SIP_TCP_Port_Min>	The lowest TCP port number that can be used for SIP sessions. Default setting: 5060
<SIP_TCP_Port_Max>	The highest TCP port number that can be used for SIP sessions. Default setting: 5080
<CTI_Enable>	Enables or disables the Computer Telephone Interface feature provided by some servers. Default setting: no

<SIP_T1>	RFC 3261 T1 value (round-trip time estimate), which can range from 0 to 64 seconds. Default setting: 0.5
<SIP_T2>	RFC 3261 T2 value (maximum retransmit interval for non-INVITE requests and INVITE responses), which can range from 0 to 64 seconds. Default setting: 4
<SIP_T4>	RFC 3261 T4 value (maximum duration a message remains in the network), which can range from 0 to 64 seconds. Default setting: 5
<SIP_Timer_B>	INVITE time-out value, which can range from 0 to 64 seconds. Default setting: 32
<SIP_Timer_F>	Non-INVITE time-out value, which can range from 0 to 64 seconds. Default setting: 32
<SIP_Timer_H>	H INVITE final response, time-out value, which can range from 0 to 64 seconds. Default setting: 32
<SIP_Timer_D>	ACK hang-around time, which can range from 0 to 64 seconds. Default setting: 32
<SIP_Timer_J>	Non-INVITE response hang-around time, which can range from 0 to 64 seconds. Default setting: 32

<INVITE_Expires>	INVITE request Expires header value. If you enter 0, the Expires header is not included in the request. Range: 0–(2^{31}-1) Default setting: 240
<ReINVITE_Expires>	ReINVITE request Expires header value. If you enter 0, the Expires header is not included in the request. Range: 0–(2^{31}-1) Default setting: 30
<Reg_Min_Expires>	Minimum registration expiration time allowed from the proxy in the Expires header or as a Contact header parameter. If the proxy returns a value less than this setting, the minimum value is used. Default setting: 1
<Reg_Max_Expires>	Maximum registration expiration time allowed from the proxy in the Min-Expires header. If the value is larger than this setting, the maximum value is used. Default setting: 7200
<Reg_Retry_Intvl>	Interval to wait before the ATA retries registration after failing during the last registration. Default setting: 30
<Reg_Retry_Long_Intvl>	When registration fails with a SIP response code that does not match Retry Reg RSC, the ATA waits for the specified length of time before retrying. If this interval is 0, the ATA stops trying. This value should be much larger than the Reg Retry Intvl value, which should not be 0. Default setting: 1200
<Reg_Retry_Random_Delay>	Random delay range (in seconds) to add to Register Retry Intvl when retrying REGISTER after a failure. Default setting: 0 (disabled)
<Reg_Retry_Long_Random_Delay>	Random delay range (in seconds) to add to Register Retry Long Intvl when retrying REGISTER after a failure. Default setting: 0 (disabled)

<Reg_Retry_Intvl_Cap>	The maximum value to cap the exponential back-off retry delay (which starts at Register Retry Intvl and doubles on every REGISTER retry after a failure) In other words, the retry interval is always at Register Retry Intvl seconds after a failure. If this feature is enabled, Reg Retry Random Delay is added on top of the exponential back-off adjusted delay value. Default setting: 0, which disables the exponential backoff feature.

<SIT1_RSC> <SIT2_RSC> <SIT3_RSC> <SIT4_RSC>	SIP response status code for the corresponding Special Information Tone (SIT), SIT1 through SIT4. For example, if you set the SIT1 RSC to 404, when the user makes a call and a failure code of 404 is returned, the SIT1 tone is played. Reorder or Busy tone is played by default for all unsuccessful response status code for SIT 1 RSC through SIT 4 RSC. Default setting: blank
<Try_Backup_RSC>	SIP response code that retries a backup server for the current request. Default setting: blank
<Retry_Reg_RSC>	Interval to wait before the ATA retries registration after failing during the last registration. Default setting: blank

<RTP_Port_Min>	Minimum port number for RTP transmission and reception. The RTP Port Min and RTP Port Max parameters should define a range that contains at least 4 even number ports, such as 100 –106. Default setting: 16384.
<RTP_Port_Max>	Maximum port number for RTP transmission and reception. Default setting: 16482.
<RTP_Packet_Size>	Packet size in seconds, which can range from 0.01 to 0.16. Valid values must be a multiple of 0.01 seconds. Default setting: 0.030

<Max_RTP_ICMP_Err>	Number of successive ICMP errors allowed when transmitting RTP packets to the peer before the ATA terminates the call. If value is set to 0, the ATA ignores the limit on ICMP errors. Default setting: 0
<RTCP_Tx_Interval>	Interval for sending out RTCP sender reports on an active connection. It can range from 0 to 255 seconds. During an active connection, the ATA can be programmed to send out compound RTCP packet on the connection. Each compound RTP packet except the last one contains a SR (Sender Report) and a SDES (Source Description) The last RTCP packet contains an additional BYE packet. Each SR except the last one contains exactly 1 RR (Receiver Report); the last SR carries no RR. The SDES contains CNAME, NAME, and TOOL identifiers. The CNAME is set to <User ID>@<Proxy>, NAME is set to <Display Name> (or Anonymous if user blocks caller ID), and TOOL is set to the Vendor/Hardware-platform-software-version. The NTP timestamp used in the SR is a snapshot of the local time for the ATA, not the time reported by an NTP server. If the ATA receives a RR from the peer, it attempts to compute the round trip delay and show it as the Call Round Trip Delay value (ms) on the *Information* page. Default setting: 0
<No_UDP_Checksum>	Select yes if you want the ATA to calculate the UDP header checksum for SIP messages. Otherwise, select no. Default setting: no
<Stats_In_BYE>	Determines whether the ATA includes the P-RTP-Stat header or response in a BYE message. The header contains the RTP statistics of the current call. Select yes or no from the drop-down list. Default setting: yes The format of the P-RTP-Stat header is: P-RTP-State: PS=<packets sent>,OS=<octets sent> ,PR=<packets received>,OR=<octets received>,PL=<packets lost>,JI=<jitter in ms>,LA=<delay in ms>,DU=<call duration ins>,EN=<encoder>,DE=<decoder>.

<NSE_Dynamic_Payload>	NSE dynamic payload type. The valid range is 96-127. Default setting: 100

<AVT_Dynamic_Payload>	AVT dynamic payload type. The valid range is 96-127. Default setting: 101
<INFOREQ_Dynamic_Payload />	INFOREQ dynamic payload type. Default setting: blank
<G726r32_Dynamic_Payload>	G726r32 dynamic payload type. Default setting: 2
<EncapRTP_Dynamic_Payload>	EncapRTP Dynamic Payload type. Default setting: 112
<RTP-Start-Loopback_Dynamic_Payload>	RTP-Start-Loopback Dynamic Payload type. Default setting: 113
<RTP-Start-Loopback_Codec>	RTP-Start-Loopback Codec. Select one of the following: G711u, G711a, G726-32, G729a. Default setting: G711u
<NSE_Codec_Name>	NSE codec name used in SDP. Default setting: NSE
<AVT_Codec_Name>	AVT codec name used in SDP. Default setting: telephone-event
<G711u_Codec_Name>	G.711u codec name used in SDP. Default setting: PCMU
<G711a_Codec_Name>	G.711a codec name used in SDP. Default setting: PCMA
<G726r32_Codec_Name>	G.726-32 codec name used in SDP. Default setting: G726-32
<G729a_Codec_Name>	G.729a codec name used in SDP. Default setting: G729a
<G722_Codec_Name>	G.722 codec name used in SDP. Default setting: G722
<EncapRTP_Codec_Name>	EncapRTP codec name used in SDP. Default setting: encaprtp

<Handle_VIA_received>	If you select yes, the ATA processes the received parameter in the VIA header (this value is inserted by the server in a response to any one of its requests) If you select no, the parameter is ignored. Select yes or no from the drop-down menu. Default setting: no
<Handle_VIA_rport>	If you select yes, the ATA processes the rport parameter in the VIA header (this value is inserted by the server in a response to any one of its requests) If you select no, the parameter is ignored. Select yes or no from the drop-down menu. Default setting: no
<Insert_VIA_received>	Inserts the received parameter into the VIA header of SIP responses if the received-from IP and VIA sent-by IP values differ. Select yes or no from the drop-down menu. Default setting: no
<Insert_VIA_rport>	Inserts the rport parameter into the VIA header of SIP responses if the received-from IP and VIA sent-by IP values differ. Select yes or no from the drop-down menu. Default setting: no
<Substitute_VIA_Addr>	Lets you use NAT-mapped IP:port values in the VIA header. Select yes or no from the drop-down menu. Default setting: no
<Send_Resp_To_Src_Port>	Sends responses to the request source port instead of the VIA sent-by port. Select yes or no from the drop-down menu. Default setting: no
<STUN_Enable>	Enables the use of STUN to discover NAT mapping. Select yes or no from the drop-down menu. Default setting: no
<STUN_Test_Enable>	If the STUN Enable feature is enabled and a valid STUN server is available, the ATA can perform a NAT-type discovery operation when it powers on. It contacts the configured STUN server, and the result of the discovery is reported in a Warning header in all subsequent REGISTER requests. If the ATA detects symmetric NAT or a symmetric firewall, NAT mapping is disabled. Default setting: no

<STUN_Server>	IP address or fully-qualified domain name of the STUN server to contact for NAT mapping discovery. Default setting: blank
<EXT_IP>	External IP address to substitute for the actual IP address of the ATA in all outgoing SIP messages. If 0.0.0.0 is specified, no IP address substitution is performed. If this parameter is specified, the ATA assumes this IP address when generating SIP messages and SDP (if NAT Mapping is enabled for that line) However, the results of STUN and VIA received parameter processing, if available, supersede this statically configured value. This option requires that you have (1) a static IP address from your Internet Service Provider and (2) an edge device with a symmetric NAT mechanism. If the ATA is the edge device, the second requirement is met. Default setting: blank
<EXT_RTP_Port_Min>	External port mapping number of the RTP Port Min. number. If this value is not zero, the RTP port number in all outgoing SIP messages is substituted for the corresponding port value in the external RTP port range. Default setting: blank
<NAT_Keep_Alive_Intvl>	Interval between NAT-mapping keep alive messages. Default setting: 15
<Redirect_Keep_Alive>	Interval between NAT Redirect keep alive messages. Default setting: 15

<Linksys_Key_System>	To enable operation with the Cisco SPA9000, choose yes. Otherwise, choose no. Default setting: no
<Multicast_Address>	The multicast address for devices in the Cisco SPA9000 voice network. Default setting: 224.168.168.168:6061
<Key_System_Auto_Discovery>	To enable auto-discovery of the Cisco SPA9000 voice system, choose yes. Otherwise, choose no. Default setting: yes

<Key_System_IP_Address>	The IP address of the Cisco SPA9000. Default setting: blank
<Force_LAN_Codec>	If needed, specify a voice codec. Default setting: none

<Line_Enable_1_> <Line_Enable_2_> <Line_Enable_5_> through <Line_Enable_13_>	To enable this line for service, select yes. Otherwise, select no. Default setting: yes

<SAS_Enable_1_> <SAS_Enable_2_>	To enable the use of the line as a streaming audio source, select yes. Otherwise, select no. If enabled, the line cannot be used for outgoing calls. Instead, it auto-answers incoming calls and streams audio RTP packets to the caller. Default setting: no
<SAS_DLG_Refresh_Intvl_1_> <SAS_DLG_Refresh_Intvl_2_>	If this value is not zero, it is the interval at which the streaming audio server sends out session refresh (SIP re-INVITE) messages to determine whether the connection to the caller is still active. If the caller does not respond to the refresh message, the ATA ends this call with a SIP BYE message. The range is 0 to 255 seconds (0 means that the session refresh is disabled) Default setting: 30

<SAS_Inbound_RTP_Sink_1_> <SAS_Inbound_RTP_Sink_2_>	The purpose of this parameter is to work around devices that do not play inbound RTP if the SAS line declares itself as a send-only device and tells the client not to stream out audio. This parameter is an FQDN or IP address of an RTP sink to be used by the SAS line in the SDP of its 200 response to inbound INVITE from a client. It will appear in the c = line and the port number, if specified, will appear in the m = line of the SDP. If this value is not specified or is equal to 0, then c = 0.0.0.0 and a=sendonly will be used in the SDP to tell the SAS client to not to send any RTP to this SAS line. If a non-zero value is specified, then a=sendrecv and the SAS client will stream audio to the given address. Special case: If the value is $IP, then the SAS line's own IP address is used in the c = line and a=sendrecv. In that case the SAS client will stream RTP packets to the SAS line. Default setting: blank

<NAT_Mapping_Enable_1_> through <NAT_Mapping_Enable_13_>	To use externally mapped IP addresses and SIP/RTP ports in SIP messages, select yes. Otherwise, select no. Default setting: no
<NAT_Keep_Alive_Enable_1_> through <NAT_Keep_Alive_Enable_13_>	To send the configured NAT keep alive message periodically, select yes. Otherwise, select no. Default setting: no
<NAT_Keep_Alive_Msg_1_> through <NAT_Keep_Alive_Msg_13_>	Enter the keep alive message that should be sent periodically to maintain the current NAT mapping. If the value is $NOTIFY, a NOTIFY message is sent. If the value is $REGISTER, a REGISTER message without contact is sent. Default setting: $NOTIFY
<NAT_Keep_Alive_Dest_1_> through <NAT_Keep_Alive_Dest_13_>	Destination that should receive NAT keep alive messages. If the value is $PROXY, the messages are sent to the current proxy server or outbound proxy server. Default setting: $PROXY

<Blind_Attn-Xfer_Enable_1_> <Blind_Attn-Xfer_Enable_2_>	Enables the ATA to perform an attended transfer operation by ending the current call leg and performing a blind transfer of the other call leg. If this feature is disabled, the ATA performs an attended transfer operation by referring the other call leg to the current call leg while maintaining both call legs. To use this feature, select yes. Otherwise, select no. Default setting: no
<MOH_Server_1_> <MOH_Server_2_>	User ID or URL of the auto-answering streaming audio server. When only a user ID is specified, the current or outbound proxy is contacted. Music-on-hold is disabled if the MOH Server is not specified. Default setting: blank
<Xfer_When_Hangup_Conf_1_> <Xfer_When_Hangup_Conf_2_>	Makes the ATA perform a transfer when a conference call has ended. Select yes or no from the drop-down menu. Default setting: yes
<Conference_Bridge_URL_1_> <Conference_Bridge_URL_2_>	This feature supports external conference bridging for n-way conference calls (n>2), instead of mixing audio locally. To use this feature, set this parameter to that of the server's name. For example: **conf@mysefver.com:12345** or **conf** (which uses the Proxy value as the domain). Default setting: blank
<Conference_Bridge_Ports_1_> <Conference_Bridge_Ports_2_>	Select the maximum number of conference call participants. The range is 3 to 10. Default setting: 3
<Enable_IP_Dialing_1_> <Enable_IP_Dialing_2_>	Enable or disable IP dialing. If IP dialing is enabled, one can dial [userid@] a.b.c.d[:port], where '@', '.', and ':' are dialed by entering *, user-id must be numeric (like a phone number) and a, b, c, d must be between 0 and 255, and port must be larger than 255. If port is not given, 5060 is used. Port and User-Id are optional. If the user-id portion matches a pattern in the dial plan, then it is interpreted as a regular phone number according to the dial plan. The INVITE message, however, is still sent to the outbound proxy if it is enabled. Default setting: no

<Emergency_Number_1_> <Emergency_Number_2_>	Comma separated list of emergency number patterns. If outbound call matches one of the pattern, the ATA will disable hook flash event handling. The condition is restored to normal after the call ends. Blank signifies that there is no emergency number. Maximum number length is 63 characters. Default setting: blank
<Mailbox_ID_1_> <Mailbox_ID_2_>	Enter the ID number of the mailbox for this line. Default setting: blank

<Proxy_1_> through <Proxy_13_>	SIP proxy server for all outbound requests. Default setting: blank
<Outbound_Proxy_1_> through <Outbound_Proxy_13_>	SIP Outbound Proxy Server where all outbound requests are sent as the first hop. Default setting: blank
<Use_Outbound_Proxy_1_> through <Use_Outbound_Proxy_13_>	Enables the use of an Outbound Proxy. If set to no, the Outbound Proxy and Use OB Proxy in Dialog parameters are ignored. Default setting: no
<Use_OB_Proxy_In_Dialog_1_> through <Use_OB_Proxy_In_Dialog_13_>	Whether to force SIP requests to be sent to the outbound proxy within a dialog. Ignored if the parameter Use Outbound Proxy is no, or the Outbound Proxy parameter is empty. Default setting: yes
<Register_1_> through <Register_13_>	Enable periodic registration with the Proxy parameter. This parameter is ignored if Proxy is not specified. Default setting: yes
<Make_Call_Without_Reg_1_> through <Make_Call_Without_Reg_13_>	Allow making outbound calls without successful (dynamic) registration by the unit. If No, dial tone will not play unless registration is successful. Default setting: no
<Register_Expires_1_> through <Register_Expires_13_>	Expires value in sec in a REGISTER request. The ATA will periodically renew registration shortly before the current registration expired. This parameter is ignored if the Register parameter is no. Range: 0 – (2^{31} – 1) sec. Default setting: 3600

<Ans_Call_Without_Reg_1_> through <Ans_Call_Without_Reg_13_>	Allow answering inbound calls without successful (dynamic) registration by the unit. Default setting: no
<Use_DNS_SRV_1_> through <Use_DNS_SRV_13_>	Whether to use DNS SRV lookup for Proxy and Outbound Proxy. Default setting: no
<DNS_SRV_Auto_Prefix_1_> through <DNS_SRV_Auto_Prefix_13_>	If enabled, the ATA will automatically prepend the Proxy or Outbound Proxy name with _sip._udp when performing a DNS SRV lookup on that name. Default setting: no
<Proxy_Fallback_Intvl_1_> through <Proxy_Fallback_Intvl_13_>	After failing over to a lower priority server, the ATA waits for the specified Proxy Fallback Interval, in seconds, before retrying the highest priority proxy (or outbound proxy) servers. This parameter is useful only if the primary and backup proxy server list is provided to the ATA via DNS SRV record lookup on the server name. (Using multiple DNS A records per server name does not allow the notion of priority, so all hosts will be considered at the same priority and the ATA will not attempt to fall back after a failover.) Default setting: 3600
<Proxy_Redundancy_Method_1_> through <Proxy_Redundancy_Method_13_>	The method that the ATA uses to create a list of proxies returned in the DNS SRV records. If you select Normal, the list will contain proxies ranked by weight and priority. If you select Based on SRV port, the ATA also inspects the port number based on 1st proxy's port. Default setting: Normal
<Mailbox_Subscribe_URL_1_> <Mailbox_Subscribe_URL_2_> <Mailbox_Subscribe_URL_5_> through <Mailbox_Subscribe_URL_13_>	The URL or IP address of the voicemail server. Default setting: blank
<Mailbox_Subscribe_Expires_1_> through <Mailbox_Subscribe_Expires_13_>	The subscription interval for voicemail message waiting indication. When this time period expires, the ATA sends another subscribe message to the voice mail server. Default: 2147483647

<Display_Name_1_> through <Display_Name_13_>	Display name for caller ID. Default setting: blank
<User_ID_1_> through <User_ID_13_>	User ID for this line. Default setting: blank
<Password_1_> through <Password_13_>	Password for this line. Default setting: blank
<Use_Auth_ID_1_> through <Use_Auth_ID_13_>	To use the authentication ID and password for SIP authentication, select yes. Otherwise, select no to use the user ID and password. Default setting: no
<Auth_ID_1_> through <Auth_ID_13_>	Authentication ID for SIP authentication. Default setting: blank
<Resident_Online_Number_1_> through <Resident_Online_Number_13_>	This setting allows you to associate a "local" telephone number with this line using a valid Skype Online Number from Skype. Calls made to that number will ring your phone. Enter the number without spaces or special characters. Default setting: blank
SIP URI	The SIP URI, in the following format: sip:<username>@<WAN_IP>:<port> or sip:<username>@<domain>:<port>

<Call_Waiting_Serv_1_> <Call_Waiting_Serv_2_> <Call_Waiting_Serv_5_> through <Call_Waiting_Serv_13_>	Enable Call Waiting Service. Default setting: yes
<Block_CID_Serv_1_> <Block_CID_Serv_2_> <Block_CID_Serv_5_> through <Block_CID_Serv_13_>	Enable Block Caller ID Service. Default setting: yes
<Block_ANC_Serv_1_> <Block,_ANC_Serv_2_> <Block,_ANC_Serv_5_> through <Block_ANC_Serv_13_>	Enable Block Anonymous Calls Service Default setting: yes

<Dist_Ring_Serv_1_> <Dist_Ring_Serv_2_> <Block_ANC_Serv_5_> through <Block_ANC_Serv_13_>	Enable Distinctive Ringing Service Default setting: yes
<Cfwd_All_Serv_1_> <Cfwd_All_Serv_2_> <Cfwd_All_Serv_5_> through <Cfwd_All_Serv_14_>	Enable Call Forward All Service Default setting: yes
<Cfwd_Busy_Serv_1_> <Cfwd_Busy_Serv_2_> <Cfwd_Busy_Serv_5_> through <Cfwd_Busy_Serv_13_>	Enable Call Forward Busy Service Default setting: yes
<Cfwd_No_Ans_Serv_1_> <Cfwd_No_Ans_Serv_2_> <Cfwd_No_Ans_Serv_5_> through <Cfwd_No_Ans_Serv_13_>	Enable Call Forward No Answer Service Default setting: yes
<Cfwd_Sel_Serv_1_> <Cfwd_Sel_Serv_2_> <Cfwd_Sel_Serv_5_> through <Cfwd_Sel_Serv_13_>	Enable Call Forward Selective Service. Default setting: yes
<Cfwd_Last_Serv_1_> <Cfwd_Last_Serv_2_> <Cfwd_Last_Serv_5_> through <Cfwd_Last_Serv_13_>	Enable Forward Last Call Service Default setting: yes
<Block_Last_Serv_1_> <Block_Last_Serv_2_> <Block_Last_Serv_5_> through <Block_Last_Serv_13_>	Enable Block Last Call Service Default setting: yes
<Accept_Last_Serv_1_> <Accept_Last_Serv_2_> <Accept_Last_Serv_5_> through <Accept_Last_Serv_13_>	Enable Accept Last Call Service Default setting: yes

<DND_Serv_1_> <DND_Serv_2_> <DND_Serv_5_> through <DND_Serv_13_>	Enable Do Not Disturb Service Default setting: yes
<CID–Serv_1_> <CID–Serv_2_> <CID–Serv_5_> through <CID–Serv_13_>	Enable Caller ID Service Default setting: yes
<CWCID_Serv_1_> <CWCID_Serv_2_> <CWCID_Serv_5_> through <CWCID_Serv_13_>	Enable Call Waiting Caller ID Service Default setting: yes
<Call_Return_Serv_1_> <Call_Return_Serv_2_> <Call_Return_Serv_5_> through <Call_Return_Serv_13_>	Enable Call Return Service Default setting: yes
<Call_Redial_Serv_1_> <Call_Redial_Serv_2_> <Call_Redial_Serv_5_> through <Call_Redial_Serv_13_>	Enable Call Redial Service.
<Call_Back_Serv_1_> <Call_Back_Serv_2_> <Call_Back_Serv_5_> through <Call_Back_Serv_13_>	Enable Call Back Service.
<Three_Way_Call_Serv_1_> <Three_Way_Call_Serv_2_> <Three_Way_Call_Serv_5_> through <Three_Way_Call_Serv_13_>	Enable Three Way Calling Service. Three Way Calling is required for Three Way Conference and Attended Transfer. Default setting: yes
<Three_Way_Conf_Serv_1_> <Three_Way_Conf_Serv_2_> <Three_Way_Conf_Serv_5_> through <Three_Way_Conf_Serv_13_>	Enable Three Way Conference Service. Three Way Conference is required for Attended Transfer. Default setting: yes

<Attn_Transfer_Serv_1_> <Attn_Transfer_Serv_2_> <Attn_Transfer_Serv_5_> through <Attn_Transfer_Serv_13_>	Enable Attended Call Transfer Service. Three Way Conference is required for Attended Transfer. Default setting: yes
<Unattn_Transfer_Serv_1_> <Unattn_Transfer_Serv_2_> <Unattn_Transfer_Serv_5_> through <Unattn_Transfer_Serv_13_>	Enable Unattended (Blind) Call Transfer Service. Default setting: yes
<MWI_Serv_1_> <MWI_Serv_2_> <MWI_Serv_5_> through <MWI_Serv_13_>	Enable MWI Service. MWI is available only if a Voice Mail Service is set-up in the deployment. Default setting: yes
<VMWI_Serv_1_> <VMWI_Serv_2_> <VMWI_Serv_5_> through <VMWI_Serv_13_>	Enable VMWI Service (FSK) Default setting: yes
<Speed_Dial_Serv_1_> <Speed_Dial_Serv_2_> <Speed_Dial_Serv_5_> through <Speed_Dial_Serv_13_>	Enable Speed Dial Service. Default setting: yes
<Secure_Call_Serv_1_> <Secure_Call_Serv_2_> <Secure_Call_Serv_5_> through <Secure_Call_Serv_13_>	Secure Call Service. If this feature is enabled, a user can make a secure call by entering an activation code (*18 by default) before dialing the target number. Then audio traffic in both directions is encrypted for the duration of the call. Default setting: yes
<Referral_Serv_1_> <Referral_Serv_2_> <Referral_Serv_5_> through <Referral_Serv_13_>	Enable Referral Service. See the Referral Services Codes parameter For more information. Default setting: yes
<Feature_Dial_Serv_1_> <Feature_Dial_Serv_2_> <Feature_Dial_Serv_5_> through <Feature_Dial_Serv_13_>	Enable Feature Dial Service. See the Feature Dial Services Codes parameter For more information. Default setting: yes

<Service_Announcement_Serv_1_> <Service_Announcement_Serv_2_> <Service_Announcement_Serv_5_> through <Service_Announcement_Serv_13_>	Enable Service Announcement Service. Default setting: no
<Reuse_CID_Number_As_Name_1_> <Reuse_CID_Number_As_Name_2_> <Reuse_CID_Number_As_Name_5_> through <Reuse_CID_Number_As_Name_13_>	Use the Caller ID number as the caller name. Default settings: yes

<Preferred_Codec_1_> through <Preferred_Codec_13_> <Second_Preferred_Codec_1_> through <Second_Preferred_Codec_13_> <Third_Preferred_Codec_1_>through <Third_Preferred_Codec_13_>	Up to three codecs to be used for all calls from the specified line/handset, listed order of preference. The actual codec used in a call depends on the outcome of the codec negotiation protocol. Select one of the following: G711u, G711a, G726-32, G729a, or G722. Default setting for Preferred Codec: G711u Default setting for Second and Third Preferred Codec: Unspecified
<Use_Pref_Codec_Only_1_>through <Use_Pref_Codec_Only_13_>	To use only the preferred codec for all calls, select yes. (The call fails if the far end does not support this codec.) Otherwise, select no. Default setting: no
<Use_Remote_Pref_Codec_1_> through <Use_Remote_Pref_Codec_13_>	To use the preferred codec specified by the remote peer, select yes. Otherwise, select no. Default setting:
<Codec_Negotiation_1_> through <Codec_Negotiation_13_>	Specify the codecs for codec negotiation: Default or List All. Default setting: Default
<G729a_Enable_1_>through <G729a_Enable_13_>	To enable the use of the G.729a codec at 8 kbps, select yes. Otherwise, select no. Default setting: yes
<Silence_Supp_Enable_1_>through <Silence_Supp_Enable_13_>	To enable silence suppression so that silent audio frames are not transmitted, select yes. Otherwise, select no. Default setting: no

<G726-32_Enable_1_> through <G726-32_Enable_13_>	To enable the use of the G.726 codec at 32 kbps, select yes. Otherwise, select no. Default setting: yes
<Silence_Threshold_1_> <Silence_Threshold_2_> <Silence_Threshold_5_> through <Silence_Threshold_13_>	Select the appropriate setting for the threshold: high, medium, or low. Default setting: medium
<FAX_V21_Detect_Enable_1_> <FAX_V21_Detect_Enable_2_>	To enable detection of V21 fax tones, select yes. Otherwise, select no. Default setting: yes
<Echo_Canc_Enable_1_> through <Echo_Canc_Enable_13_>	To enable the use of the echo canceller, select yes. Otherwise, select no. Default setting: yes
<FAX_CNG_Detect_Enable_1_> <FAX_CNG_Detect_Enable_2_>	To enable detection of the fax Calling Tone (CNG), select yes. Otherwise, select no. Default setting: yes
<FAX_Passthru_Codec_1_> through <FAX_Passthru_Codec_2_>	Select the codec for fax passthrough, G711u or G711a. Default setting: G711u
<FAX_Codec_Symmetric_1_> through <FAX_Codec_Symmetric_2_>	To force the ATA to use a symmetric codec during fax passthrough, select yes. Otherwise, select no. Default setting: yes
<DTMF_Process_INFO_1_> through <DTMF_Process_INFO_13_>	To use the DTMF process info feature, select yes. Otherwise, select no. Default setting: yes
<FAX_Passthru_Method_1_> through <FAX_Passthru_Method_2_>	Select the fax passthrough method: None, NSE, or ReINVITE. Default setting: NSE
<DTMF_Process_AVT_1_> through <DTMF_Process_AVT_13_>	To use the DTMF process AVT feature, select yes. Otherwise, select no. Default setting: yes
<FAX_Process_NSE_1_> through <FAX_Process_NSE_2_>	To use the fax process NSE feature, select yes. Otherwise, select no. Default setting: yes

<DTMF_Tx_Method_1_> through <DTMF_Tx_Method_13_>	Select the method to transmit DTMF signals to the far end: InBand, AVT, INFO, or Auto. InBand sends DTMF by using the audio path. AVT sends DTMF as AVT events. INFO uses the SIP INFO method. Auto uses InBand or AVT based on the outcome of codec negotiation. Default setting: Auto
<FAX_Disable_ECAN_1_> through <FAX_Disable_ECAN_2_>	If enabled, this feature automatically disables the echo canceller when a fax tone is detected. To use this feature, select yes. Otherwise, select no. Default setting: no
<DTMF_Tx_Mode_1_> <DTMF_Tx_Mode_2_> <DTMF_Tx_Mode_5_> through <DTMF_Tx_Mode_13_>	DTMF Detection Tx Mode is available for SIP information and AVT. Options are: Strict or Normal. Default setting: Strict for which the following are true: ▪ A DTMF digit requires an extra hold time after detection. ▪ The DTMF level threshold is raised to -20 dBm. The minimum and maximum duration thresholds are: ▪ strict mode for AVT: 70 ms ▪ normal mode for AVT: 40 ms ▪ strict mode for SIP info: 90 ms ▪ normal mode for SIP info: 50 ms
<DTMF_Tx_Strict_Hold_Off_Time_1_> <DTMF_Tx_Strict_Hold_Off_Time_2_>	This parameter is in effect only when DTMF Tx Mode is set to strict, and when DTMF Tx Method is set to out-ofband; i.e. either AVT or SIP-INFO. The value can be set as low as 40 ms. There is no maximum limit. A larger value will reduce the chance of talk-off (beeping) during conversation, at the expense of reduced performance of DTMF detection, which is needed for interactive voice response systems (IVR) Default: 70 ms
<FAX_Enable_T38_1_> <FAX_Enable_T38_2_>	To enable the use of ITU-T T.38 standard for FAX Relay, select yes. Otherwise select no. Default setting: yes

<Hook_Flash_Tx_Method_1_> <Hook_Flash_Tx_Method_2_>	Select the method for signaling hook flash events: None, AVT, or INFO. None does not signal hook flash events. AVT uses RFC2833 AVT (event = 16) INFO uses SIP INFO with the single line signal=hf in the message body. The MIME type for this message body is taken from the Hook Flash MIME Type setting. Default setting: None
<FAX_T38_Redundancy_1_> <FAX_T38_Redundancy_2_>	Select the appropriate number to indicate the number of previous packet payloads to repeat with each packet. Choose 0 for no payload redundancy. The higher the number, the larger the packet size and the more bandwidth consumed. Default setting: 1
<FAX_T38_ECM_Enable_1_> <FAX_T38_ECM_Enable_2_>	Select yes to enable T.38 Error Correction Mode. Otherwise select no. Default setting: yes
<FAX_Tone_Detect_Mode_1_> <FAX_Tone_Detect_Mode_2_>	This parameter has three possible values: ▪ **caller or callee**: The ATA will detect FAX tone whether it is callee or caller ▪ **caller only:** The ATA will detect FAX tone only if it is the caller ▪ **callee only**: The ATA will detect FAX tone only if it is the callee Default setting: caller or callee.
<Symmetric_RTP_1_> <Symmetric_RTP_2_> <Symmetric_RTP_5_> through <Symmetric_RTP_13_>	Enable symmetric RTP operation. If enabled, the ATA sends RTP packets to the source address and port of the last received valid inbound RTP packet. If disabled (or before the first RTP packet arrives) the ATA sends RTP to the destination as indicated in the inbound SDP. Default setting: no
<FAX_T38_Return_to_Voice_1_> <FAX_T38_Return_to_Voice_2_>	When this feature is enabled, upon completion of the fax image transfer, the connection remains established and reverts to a voice call using the previously designated codec. Select yes to enable this feature, or select no to disable it. Default setting: no

<Dial_Plan_1_> through <Dial_Plan_13_>	The allowed number patterns for outbound calls. The default dial plan script for the line is as follows: (*xx|[3469]11|0|00|[2-9]xxxxxx|1xxx[2-9]xxxxxx|xxxxxxxxxxx.) Each parameter is separated by a semi-colon (;) **Example Dial Plan Entry** — **Functionality** (*xx — Allow arbitrary 2 digit star code [3469]11 — Allow x11 sequences 0 — Operator 00 — Int'l Operator [2-9]xxxxxx — US local number 1xxx[2-9]xxxxxx — US 1 + 10-digit long distance xxxxxxxxxxx. — Everything else

<Gateway_1_1_> through <Gateway_4_1_>	The first of 4 gateways that can be specified to be used in the <Dial Plan> to facilitate call routing specification (that overrides the given proxy information). This gateway is represented by gw1 in the <Dial Plan>. For example, the rule 1408xxxxxxx<:@gw1> can be added to the dial plan such that when the user dials 1408+7digits, the call will be routed to Gateway 1. Without the <:@gw1> syntax, all calls are routed to the given proxy by default (except IP dialing). Default setting: blank
<GW1_NAT_Mapping_Enable_1_> through<GW4_NAT_Mapping_Enable_1_>	If enabled, the ATA uses NAT mapping when contacting Gateway 1. Default setting: no
<GW1_Auth_ID_1_ > through <GW4_Auth_ID_1_ >	This value is the authentication user-id to be used by the ATA to authenticate itself to Gateway 1. Default setting: blank
<GW1_Password_1_> through <GW4_Password_1_>	This value is the password to be used by the ATA to authenticate itself to Gateway 1. Default setting: blank

<Auto_PSTN_Fallback_1_> <Auto_PSTN_Fallback_2_> <Auto_PSTN_Fallback_5_> through <Auto_PSTN_Fallback_13_>	If enabled, the ATA automatically routes all calls to the PSTN gateway when the SIP proxy is down (registration failure or network link down). Default setting: yes

<Cfwd_No_Ans_Dest_1_> <Cfwd_No_Ans_Dest_2_> <Cfwd_No_Ans_Dest_5_> through <Cfwd_No_Ans_Dest_13_>	Forward number for Call Forward No Answer Service. Same as Cfwd All Dest. Default setting: blank
<Cfwd_No_Ans_Delay_1_> <Cfwd_No_Ans_Delay_2_> <Cfwd_No_Ans_Delay_5_> through <Cfwd_No_Ans_Delay_13_>	Delay in sec before Call Forward No Answer triggers. Same as Cfwd All Dest. Default setting: 20
<Idle_Polarity_1_> <Idle_Polarity_2_>	Polarity before a call is connected: Forward or Reverse. Default setting: Forward
<Caller_Conn_Polarity_1_> <Caller_Conn_Polarity_2_>	Polarity after an outbound call is connected: Forward or Reverse. Default setting: Forward.
<Callee_Conn_Polarity_1_> <Callee_Conn_Polarity_2_>	Polarity after an inbound call is connected: Forward or Reverse. Default setting: Forward

<Cfwd_All_Dest_1_> <Cfwd_All_Dest_2_> <Cfwd_All_Dest_5_> through <Cfwd_All_Dest_13_>	Forward number for Call Forward All Service. Default setting: blank
<Cfwd_Busy_Dest_1_> <Cfwd_Busy_Dest_2_> <Cfwd_Busy_Dest_5_> through <Cfwd_Busy_Dest_13_>	Forward number for Call Forward Busy Service. Same as Cfwd All Dest. Default setting: blank

<Cfwd_Sel1_Caller_1_> through <Cfwd_Sel8_Caller_1_> <Cfwd_Sel1_Caller_2_> through <Cfwd_Sel8_Caller_2_>	Caller number pattern to trigger Call Forward Selective service. When the caller's phone number matches the entry, the call is forwarded to the corresponding Cfwd Selective Destination (Cfwd Sel1-8 Dest). ▪ Use *?* to match any single digit. ▪ Use * to match any number of digits. Example: 1408*, 1512???1234 In the above example, a call is forwarded to the corresponding destination if the caller ID either starts with 1408 or is an 11-digit numbering starting with 1512 and ending with 1234. Default setting: blank
<Cfwd_Sel1_Dest_1_> through <Cfwd_Sel8_Dest_1_> <Cfwd_Sel1_Dest_2_> through <Cfwd_Sel8_Dest_2_>	The destination for the corresponding Call Forward Selective caller pattern (Cfwd Sel1-8 Caller). Default setting: blank <FAX_CNG_Detect_Enable_1_>
<Cfwd_Last_Caller_1_> <Cfwd_Last_Caller_2_>	The number of the last caller; this caller is actively forwarded to the Cfwd Last Dest via the Call Forward Last service. Default setting: blank
<Cfwd_Last_Dest_1_> <Cfwd_Last_Dest_2_>	The destination for the Cfwd Last Caller.
<Block_Last_Caller_1_> <Block_Last_Caller_2_>	The number of the last caller; this caller is blocked via the Block Last Caller Service. Default setting: blank
<Accept_Last_Caller_1_> <Accept_Last_Caller_2_>	The number of the last caller; this caller is accepted via the Accept Last Caller Service. Default setting: blank

<Speed_Dial_2_1_> through <Speed_Dial_9_1_> <Speed_Dial_2_2_> through <Speed_Dial_9_2_> <FAX_CNG_Detect_Enable_1_>	Target phone number (or URL) assigned to speed dial 2, 3, 4, 5, 6, 7, 8, or 9. Default setting: blank

<CW_Setting_1_> <CW_Setting_2_>	Call Waiting on/off for all calls. Default setting: yes
<Block_CID_Setting_1_> <Block_CID_Setting_2_>	Block Caller ID on/off for all calls. Default setting: no
<Block_ANC_Setting_1_> <Block_ANC_Setting_2_>	Block Anonymous Calls on or off. Default setting: no
<DND_Setting_1_> <DND_Setting_2_>	DND on or off. Default setting: no
<CID_Setting_1_> <CID_Setting_2_>	Caller ID Generation on or off. Default setting: yes
<CWCID_Setting_1_> <CWCID_Setting_2_>	Call Waiting Caller ID Generation on or off. Default setting: yes
<Dist_Ring_Setting_1_> <Dist_Ring_Setting_2_>	Distinctive Ring on or off. Default setting: yes

<Secure_Call_Setting_1_> <Secure_Call_Setting_2_>	If yes, all outbound calls are secure calls by default, without requiring the user to dial a star code first. Default setting: no ▪ If Secure Call Setting is set to yes, all outbound calls are secure. However, a user can disable security for a call by dialing *19 before dialing the target number. ▪ If Secure Call Setting is set to No, the user can make a secure outbound call by dialing *18 before dialing the target number. ▪ A user cannot force inbound calls to be secure or not secure; that depends on whether the caller has security enabled or not. **Note:** This setting is applicable only if Secure Call Serv is set to yes on the line interface.
<Message_Waiting_1_> <Message_Waiting_2_>	Setting this value to yes can activate stutter tone and VMWI signal. This parameter is stored in long term memory and will survive after reboot or power cycle. Default setting: no
<Accept_Media_Loopback_Request_1_> <Accept_Media_Loopback_Request_2_> <Accept_Media_Loopback_Request_5_> through <Accept_Media_Loopback_Request_13_>	Controls how to handle incoming requests for loopback operation. Default setting: automatic ▪ **never:** Never accepts loopback calls; replies 486 to the caller. ▪ **automatic:** Automatically accepts the call without ringing. ▪ **manual:** Rings the phone first, and the call must be picked up manually before loopback starts. Default setting: Automatic
<Media_Loopback_Mode_1_> <Media_Loopback_Mode_2_> <Media_Loopback_Mode_5_> through <Media_Loopback_Mode_13_>	The loopback mode to assume locally when making call to request media loopback. Choices are: Source and Mirror. Default setting: source NOTE If the ATA answers the call, the mode is determined by the caller.

<Media_Loopback_Type_1_> <Media_Loopback_Type_2_> <Media_Loopback_Type_5_> through <Media_Loopback_Type_13_>	The loopback type to use when making call to request media loopback operation. Choices are Media and Packet. Default setting: media Note that if the ATA answers the call, then the loopback type is determined by the caller (the ATA always picks the first loopback type in the offer if it contains multiple type)

<Ring1_Caller_1_ > through <Ring8_ Caller_1_ > <Ring1_Caller_2_ > through <Ring8_ Caller_2_ > <FAX_CNG_Detect_Enable_1_>	Caller number pattern to play Distinctive Ring/CWT 1, 2, 3, 4, 5, 6, 7, or 8. Caller number patterns are matched from Ring 1 to Ring 8. The first match (not the closest match) will be used for alerting the subscriber. Default setting: blank

<Default_Ring_1_> <Default_Ring_2_>	Default ringing pattern, 1–8, for all callers. Default setting: 1
<Default_CWT_1_> <Default_CWT_2_>	Default CWT pattern, 1–8, for all callers. Default setting: 1
<Hold_Reminder_Ring_1_> <Hold_Reminder_Ring_2_>	Ring pattern for reminder of a holding call when the phone is on-hook. Default setting: 8
<Call_Back_Ring_1_> <Call_Back_Ring_2_>	Ring pattern for call back notification. Default setting: 7
<Cfwd_Ring_Splash_Len_1_> <Cfwd_Ring_Splash_Len_2_>	Duration of ring splash when a call is forwarded (0 – 10.0s) Default setting: 0
<Cblk_Ring_Splash_Len_1_> <Cblk_Ring_Splash_Len_2_>	Duration of ring splash when a call is blocked (0 – 10.0s) Default setting: 0

<VMWI_Ring_Policy_1_> <VMWI_Ring_Policy_2_>	The parameter controls when a ring splash is played when a the VM server sends a SIP NOTIFY message to the ATA indicating the status of the subscriber's mail box. Three settings are available. Default setting: New VM Available ▪ New VM Available: Ring as long as there new voicemail messages. ▪ New VM Becomes Available: Ring at the point when the first new voicemail message is received. ▪ New VM Arrives: Ring when the number of new voicemail messages increases.
<VMWI_Ring_Splash_Len_1_> <VMWI_Ring_Splash_Len_2_>	Duration of ring splash when new messages arrive before the VMWI signal is applied (0 – 10.0s) Default setting: 0
<Ring_On_No_New_VM_1_> <Ring_On_No_New_VM_2_>	If enabled, the ATA plays a ring splash when the voicemail server sends SIP NOTIFY message to the ATA indicating that there are no more unread voice mails. Some equipment requires a short ring to precede the FSK signal to turn off VMWI lamp. Default setting: no

<PSTN_Line_Enable_3_>	To enable this line for service, select yes. Otherwise, select no. Default setting: yes
<Incoming_Handset_List_3_> through <Incoming_Handset_List_13_>	The devices that ring when an incoming call is received on the specified line. Default setting: fxs,1,2,3,4,5,6,7,8,9,10

<SIP_ToS/DiffServ_Value_1_> through <SIP_ToS/DiffServ_Value_5_>	TOS/DiffServ field value in UDP IP packets carrying a SIP message. Default setting: 0x68
<SIP_CoS_Value_1_> through <SIP_CoS_Value_5_>	CoS value for SIP messages. Valid values are 0 through 7. Default setting: 3

<RTP_ToS/DiffServ_Value_1> through <RTP_ToS/DiffServ_Value_5_>	ToS/DiffServ field value in UDP IP packets carrying RTP data. Default setting: 0xb8
<RTP_CoS_Value_1_> through <RTP_CoS_Value_5_>	CoS value for RTP data. Valid values are 0 through 7. Default setting: 6
<Network_Jitter_Level_1_> through <Network_Jitter_Level_5_>	Determines how jitter buffer size is adjusted by the ATA. Jitter buffer size is adjusted dynamically. The minimum jitter buffer size is 30 milliseconds or (10 milliseconds + current RTP frame size), whichever is larger, for all jitter level settings. However, the starting jitter buffer size value is larger for higher jitter levels. This setting controls the rate at which the jitter buffer size is adjusted to reach the minimum. Select the appropriate setting: low, medium, high, very high, or extremely high. Default setting: high
<Jitter_Buffer_Adjustment_1_> through <Jitter_Buffer_Adjustment_5_>	Choose yes to enable or no to disable this feature. Default setting: yes

<SIP_Transport_1_> through <SIP_Transport_5_>	The TCP choice provides “guaranteed delivery”, which assures that lost packets are retransmitted. TCP also guarantees that the SIP packages are received in the same order that they were sent. As a result, TCP overcomes the main disadvantages of UDP. In addition, for security reasons, most corporate firewalls block UDP ports. With TCP, new ports do not need to be opened or packets dropped, because TCP is already in use for basic activities such as Internet browsing or e-commerce. Options are: UDP, TCP, TLS. Default setting: UDP
<SIP_Port_1_> through <SIP_Port_5_>	Port number of the SIP message listening and transmission port. Default setting: 5060
<SIP_100REL_Enable_1_> through <SIP_100REL_Enable_5_>	To enable the support of 100REL SIP extension for reliable transmission of provisional responses (18x) and use of PRACK requests, select yes. Otherwise, select no. Default setting: no
<EXT_SIP_Port_1_> through <EXT_SIP_Port_5_>	The external SIP port number. Default setting: blank

<Auth_Resync-Reboot_1_> through <Auth_Resync-Reboot_5_>	If this feature is enabled, the ATA authenticates the sender when it receives the NOTIFY resync reboot (RFC 2617) message. To use this feature, select yes. Otherwise, select no. Default setting: yes
<SIP_Proxy-Require_1_> through <SIP_Proxy-Require_5_>	The SIP proxy can support a specific extension or behavior when it sees this header from the user agent. If this field is configured and the proxy does not support it, it responds with the message, unsupported. Enter the appropriate header in the field provided. Default setting: blank
<SIP_Remote-Party-ID_1_> <SIP_Remote-Party-ID_2_>	To use the Remote-Party-ID header instead of the From header, select yes. Otherwise, select no. Default setting: yes
<SIP_GUID_1_> through <SIP_GUID_5_>	This feature limits the registration of SIP accounts. The Global Unique ID is generated for each line for each ATA. When it is enabled, the ATA adds a GUID header in the SIP request. The GUID is generated the first time the unit boots up and stays with the unit through rebooting and even factory reset. Default setting: no

<SIP_Debug_Option_1_> through <SIP_Debug_Option_5_>	SIP messages are received at or sent from the proxy listen port. This feature controls which SIP messages to log. The choices are described below. Default setting: none ▪ **none**—No logging. ▪ **1-line**—Logs the start-line only for all messages. ▪ **1-line excl. OPT**—Logs the start-line only for all messages except OPTIONS requests/responses. ▪ **1-line excl. NTFY**—Logs the start-line only for all messages except NOTIFY requests/responses. ▪ **1-line excl. REG**—Logs the start-line only for all messages except REGISTER requests/responses. ▪ **1-line excl. OPT|NTFY|REG**—Logs the start-line only for all messages except OPTIONS, NOTIFY, and REGISTER requests/responses. ▪ **full**—Logs all SIP messages in full text. ▪ **full excl. OPT**—Logs all SIP messages in full text except OPTIONS requests/responses. ▪ **full excl. NTFY**—Logs all SIP messages in full text except NOTIFY requests/responses. ▪ **full excl. REG**—Logs all SIP messages in full text except REGISTER requests/responses. ▪ **full excl. OPT|NTFY|REG**—Logs all SIP messages in full text except for OPTIONS, NOTIFY, and REGISTER requests/responses.
<RTP_Log_Intvl_1_> through <RTP_Log_Intvl_5_>	The interval for the RTP log. Default setting: 0
<Restrict_Source_IP_1_> through <Restrict_Source_IP_5_>	If configured, the ATA drops all packets sent to its SIP Ports from an untrusted IP address. A source IP address is untrusted if it does not match any of the IP addresses resolved from the configured Proxy (or Outbound Proxy if Use Outbound Proxy is yes) Default setting: no
<Referor_Bye_Delay_1_> through <Referor_Bye_Delay_5_>	The number of seconds to wait before sending a BYE to the referrer to terminate a stale call leg after a call transfer.

<Refer_Target_Bye_Delay_1_> through <Refer_Target_Bye_Delay_5_>	The number of seconds to wait before sending a BYE to the refer target to terminate a stale call leg after a call transfer.
<Referee_Bye_Delay_1_> through <Referee_Bye_Delay_5_>	The number of seconds to wait before sending a BYE to the referee to terminate a stale call leg after a call transfer.
<Refer-To_Target_Contact_1_> through <Refer-To_Target_Contact_5_>	To contact the refer-to target, select yes. Otherwise, select no. Default setting: no
<Sticky_183_1_> through <Sticky_183_5_>	If this feature is enabled, the ATA ignores further 180 SIP responses after receiving the first 183 SIP response for an outbound INVITE. To enable this feature, select yes. Otherwise, select no. Default setting: no
<Auth_INVITE_1_> through <Auth_INVITE_5_>	When enabled, authorization is required for initial incoming INVITE requests from the SIP proxy. Default setting: no
<Reply_182_On_Call_Waiting_1_> through <Reply_182_On_Call_Waiting_5_>	When enabled, the ATA replies with a SIP182 response to the caller if it is already in a call and the line is off-hook. To use this feature select yes. Default setting: no
<Use_Anonymous_With_RPID_1_> through <Use_Anonymous_With_RPID_5_>	Determines whether or not the ATA uses “Anonymous” when Remote Party ID is requested in the SIP message. Default setting: yes
<Use_Local_Addr_In_From_1_> through <Use_Local_Addr_In_From_5_>	Use the local ATA IP address in the SIP FROM message. Default setting: no

<Dial_Plan_1_3_> through <Dial_Plan_8_3_>	The PSTN dial plan pool. You can associate a dial plan with a VoIP Caller or a PSTN Caller by referencing the index number (1~8). Default setting: (xx.)

<VoIP-To-PSTN_Gateway_Enable_3_>	Choose yes to enable or choose no to disable the VoIP-To-PSTN Gateway functionality. Default setting: yes
<VoIP_Caller_Auth_Method_3_>	The method to authenticate a VoIP Caller to access the PSTN gateway. Choose from none, PIN, or HTTP Digest. Default setting: none
<VoIP_PIN_Max_Retry_3_>	The number of times that a VoIP caller can attempt to enter a PIN, if the VoIP Caller Auth Method is set to PIN. Default setting: 3
<One_Stage_Dialing_3_>	Choose yes to enable or choose no to disable one-stage dialing. This setting applies if the VoIP Caller Auth Method is none or HTTP Digest, or if caller is in the Access List. Default setting: yes
<Line_1_VoIP_Caller_DP_3_>	The index number of the dial plan to use when the VoIP Caller is calling from Line 1 of the same ATA during normal operation (in other words, not due to fallback to PSTN service when Line 1 VoIP service is down). the Authentication is skipped for Line 1 VoIP caller. Default setting: 1
<VoIP_Caller_Default_DP_3_>	The index number of the dial plan to use when the VoIP Caller is not authenticated. Default setting: 1
<Line_1_Fallback_DP_3_>	The index number of the dial plan to use when the VoIP Caller is calling from Line 1 of the same ATA due to fallback to PSTN service when Line 1 VoIP service is down. Default setting: none

<VoIP_Caller_ID_Pattern_3_>	A comma-separated list of caller phone number patterns that is used to allow or block access to the PSTN gateway based on the caller ID. If the caller ID does not match a specified pattern, access is rejected, regardless of the authentication method. This comparison is applied before the access list is applied. If this parameter is blank (not specified), all callers are considered for VoIP service. ▪ Use ? to match any single digit. ▪ Use * to match any number of digits. Example: 1408*, 1512???1234 In the above example, the caller ID either must start with 1408 or must be an 11-digit numbering starting with 1512 and ending with 1234. Default setting: blank
<VoIP_Access_List_3_>	A comma-separated list of number patterns that is used to allow or block access to the PSTN gateway based on the source IP address. If the IP address matches a specified pattern, service is allowed without further authentication. Example: 192.168.*.*, 66.43.12.1??. In the above example, the source IP address either must begin with 192.168 or must be in the range of 66.43.12.100-199. Default setting: blank
<VoIP_Caller_1_PIN_3_> through <VoIP_Caller_8_PIN_3_>	A PIN number that a VoIP caller can use to access the PSTN gateway, when the VoIP Caller Auth Method is set to PIN. Default setting: blank
<VoIP_Caller_1_DP_3_>through <VoIP_Caller_8_DP_3_>	The index number of the dial plan to use upon successful entry of the corresponding VoIP Caller PIN. Default setting: 1

<VoIP_User_1_Auth_ID_3_> through <VoIP_User_8_Auth_ID_3_>	A user ID that a VoIP Caller can use for authentication by using the HTTP Digest method (in other words, by embedding an Authorization header in the SIP INVITE message sent to the ATA. If the credentials are missing or incorrect, the ATA will challenge the caller with a 401 response). The VoIP caller whose authentication user-id equals to this ID is referred to VoIP User 1 of this ATA. **NOTE:** If the caller specifies an authentication user-id that does not match any of the VoIP User Auth ID's, the INVITE will be rejected with a 403 response. Default setting: blank.
<VoIP_User_1_Password_3_> through <VoIP_User_8_Password_3_>	The password to be used with VoIP User 1. The user assumes the identity of VoIP User 1 must therefore compute the credentials using this password, or the INVITE will be challenged with a 401 response Default setting: blank.
<VoIP_User_1_DP_3_>through <VoIP_User_8_DP_3_>	For up to 8 VoIP users, specify the index of the dial plan to be used after successful authentication. If authentication is disabled, the default dial plan is used for all unknown VoIP users. Default setting: 1.

<PSTN-To-VoIP_Gateway_Enable_3_>	Select yes to enable or select no to disable PSTN-To-VoIP Gateway functionality. Default setting: yes
<PSTN_Caller_Auth_Method_3_>	The method to authenticate a PSTN Caller to access the VoIP gateway. Choose from none or PIN. Default setting: none
<PSTN_Ring_Thru_1_3_>	To enable ring through to Line 1 based on caller number patterns, choose yes. Otherwise choose no. **Note:** For more information about PSTN Caller number patterns, see **<PSTN_Caller_ID_Pattern_3_>**. Default setting: yes

<PSTN_PIN_Max_Retry_3_>	The number of times that a PSTN caller can attempt to enter a PIN number, if the authentication method is set to PIN. Default setting: 3
<PSTN_CID_for_VoIP_CID_3_>	Choose yes or no. Default setting: no
<PSTN_CID_Number_Prefix_3_>	A dialing prefix, if needed, to add to the caller ID number on the PBX to ensure that a callback goes to the correct number. Default setting: blank
<PSTN_Caller_Default_DP_3_>	The index number of the dial plan that is used when the PSTN Caller Auth Method is set to none. Default settings: 1
<Line_1_Signal_Hook_Flash_to_PSTN_3_>	Specify the operation of the hook flash on the analog phone when a PSTN-to-VoIP call is active. Choose Disabled or Double Hook Flash. Default setting: Disabled
<PSTN_CID_Name_Prefix_3_>	The prefix to add to the caller ID name that is sent to the PBX. Enter the characters to add to the caller ID name. Default setting: blank
<PSTN_Caller_ID_Pattern_3_>	A comma-separated list of phone number patterns that is used to allow or block access to the VoIP gateway based on the caller ID. If the caller ID does not match a specified pattern, access is rejected, regardless of the authentication method. This comparison is applied before the access list is applied. If this parameter is blank (not specified), all callers are considered for VoIP service. ▪ Use ? to match any single digit. ▪ Use * to match any number of digits. Example: 1408*, 1512???1234 In the above example, the caller ID either must start with 1408 or must be an 11-digit numbering starting with 1512 and ending with 1234. Default setting: blank

<PSTN_Access_List_3_>	A comma-separated list of number patterns that is used to allow or block access to the VoIP gateway based on the destination IP address. If the destination IP address matches a specified pattern, service is allowed without further authentication. Example: 192.168.*.*, 66.43.12.1??. In the above example, the IP address either must begin with 192.168 or must be in the range of 66.43.12.100-199. The default is blank.
<PSTN_Caller_1_PIN_3_> through <PSTN_Caller_8_PIN_3_>	A PIN number that allows a PSTN caller to access to the VoIP gateway. Calls will be subject to the dial plan specified by the corresponding PSTN Caller DP setting (see below). These settings apply when the PSTN Caller Authentication Method parameter is set to PIN. Default setting: blank
<PSTN_Caller_1_DP_3_> through <PSTN_Caller_8_DP_3_>	The index number of the dial plan to use upon successful entry of the corresponding PSTN Caller PIN. Default setting: 1

<VoIP_Answer_Delay_3_>	The number of seconds to wait before auto-answering an inbound VoIP call for the FXO account. The range is 0-255. Default setting: 0
<VoIP_PIN_Digit_Timeout_3_>	After a VoIP caller is prompted for a PIN or enters a digit, the number of seconds to wait for an entry. The range is 0-255. Default setting: 10
<PSTN_Answer_Delay_3_>	After an inbound PSTN call starts ringing, the number of seconds to wait before auto-answering the call. The range is 0-255. Default setting: 16
<PSTN_PIN_Digit_Timeout_3_>	After a PSTN caller is prompted for a PIN or enters a digit, the number of seconds to wait for an entry. The range is 0-255. Default setting: 10
<PSTN-To-VoIP_Call_Max_Dur_3_>	The limit on the duration of a PSTN-To-VoIP Gateway Call. Unit is in seconds. 0 means unlimited. The range is 0-2147483647. Default setting: 0

<PSTN_Ring_Thru_Delay_3_>	After a PSTN call starts ringing, the number of seconds to wait before ring through to Line 1. In order for Line 1 to have the caller ID information, this value must be greater than the time required to complete the PSTN caller ID delivery. The range is 0-255. Default setting: 1
<VoIP-To-PSTN_Call_Max_Dur_3_>	The limit on the duration of a VoIP-To-PSTN Gateway Call. Unit is in seconds. 0 means unlimited. The range is 0-2147483647. Default setting: 0
<PSTN_Ring_Thru_CWT_Delay_3_>	When a call is active and a new PSTN call starts ringing, the number of seconds to wait before ring through to Line 1 with a Call WaitingTone. Default setting: 3
<VoIP_DLG_Refresh_Intvl_3_>	The interval between (SIP) Dialog refresh messages sent by the ATA to detect if the VoIP call-leg is still up. If this value is set to 0, the VoIP call-leg status will not be checked by the ATA. The refresh message is a SIP RelNVITE, and the VoIP peer must response with a 2xx response. If the VoIP peer does not reply or the response is not greater than 2xx, the ATA will disconnect both call legs automatically. The range is 0-255. Default setting: 0
<PSTN_Ring_Timeout_3_>	After a ring burst, the number of seconds to wait before concluding that PSTN ring has ceased. The range is 0-255. Default setting: 5
<PSTN_Dialing_Delay_3_>	After hook, the number of seconds to wait before dialing a PSTN number. The range is 0-255. Default setting: 1
<PSTN_Dial_Digit_Len_3_>	The on/off time when the Gateway transmits digits through the Line (FXO) port. The syntax is *on-time/off-time*, expressed in seconds. The permitted range is 0.05 to 3.00 (up to two decimal places only). Default setting: .1/.1
<PSTN_Hook_Flash_Len_3_>	The length of the hook flash in seconds. Default setting: .25

<Detect_CPC_3_>	Choose yes to enable or choose no to disable this feature. CPC is a brief removal of tip-and-ring voltage. If enabled, the ATA will disconnect both call legs when this signal is detected during a gateway call. Default setting: yes
<Detect_Polarity_Reversal_3_>	Choose yes to enable or choose no to disable this feature. If enabled, the ATA will disconnect both call legs when this signal is detected during a gateway call. If it is a PSTN gateway call, the first polarity reversal is ignored and the second one triggers the disconnection. For VoIP gateway call, the first polarity reversal triggers the disconnection. Default setting: yes
<Detect_PSTN_Long_Silence_3_>	Choose yes to enable or choose no to disable this feature. If enabled, the ATA will disconnect both call legs when the PSTN side has no voice activity for a duration longer than the length specified in the Long Silence Duration parameter during a gateway call. Default setting: no
<Detect_VoIP_Long_Silence_3_>	Choose yes to enable or choose no to disable this feature. If enabled, the ATA will disconnect both call legs when the VoIP side has no voice activity for a duration longer than the length specified in the Long Silence Duration parameter during a gateway call. Default setting: no
<PSTN_Long_Silence_Duration_3_>	This value is minimum length of PSTN silence (or inactivity) in seconds to trigger a gateway call disconnection if Detect Long Silence is enabled. Default setting: 30
<VoIP_Long_Silence_Duration_3_>	This value is minimum length of VoIP silence (or inactivity) in seconds to trigger a gateway call disconnection if Detect Long Silence is enabled. Default setting: 30
<PSTN_Silence_Threshold_3_>	This parameter adjusts the sensitivity of PSTN silence detection. Choose from {very low, low, medium, high, very high}. The higher the setting, the easier to detect silence and hence easier to trigger a disconnection. Default setting: medium

<Min_CPC_Duration_3_>	Specify the minimum duration of a low tip-and-ring voltage (below 1V) for the Gateway to recognize it as a CPC signal or PSTN line removal. Default setting: 0.2
<Detect_Disconnect_Tone_3_>	Choose yes to enable or choose no to disable this feature. If enabled, the ATA will disconnect both call legs when it detects the disconnect tone from the PSTN side during a gateway call. Disconnect tone is specified in the *Disconnect Tone* parameter, which depends on the region of the PSTN service. Default setting: yes
<Disconnect_Tone_3_>	This value is the tone script which describes to the ATA the tone to detect as a disconnect tone. The syntax follows a standard Tone Script with some restrictions. Default value is standard US reorder (fast busy) tone, for 4 seconds. Default setting: 480@-30,620@-30;4(.25/.25/1+2) Restrictions: ▪ Two frequency components must be given. If single frequency is desired, the same frequency is used for both ▪ The tone level value is not used. –30 (dBm) should be used for now. ▪ Only 1 segment set is allowed ▪ Total duration of the segment set is interpreted as the minimum duration of the tone to trigger detection ▪ 6 segments of on/off time (seconds) can be specified. A 10% margin is used to validated cadence characteristics of the tone.

<<Disconnect_Tone_3_>	Disconnect Tone Script values: US—480@-30,620@-30;4(.25/.25/1+2) UK—400@-30,400@-30; 2(3/0/1+2) France—440@-30,440@-30; 2(0.5/0.5/1+2) Germany—440@-30,440@-30; 2(0.5/0.5/1+2) Netherlands—425@-30,425@-30; 2(0.5/0.5/1+2) Sweden—425@-10; 10(0.25/0.25/1) Norway—425@-10; 10(0.5/0.5/1) Italy—425@-30,425@-30; 2(0.2/0.2/1+2) Spain—425@-10; 10(0.2/0.2/1,0.2/0.2/1,0.2/0.6/1) Portugal—425@-10; 10(0.5/0.5/1) Poland—425@-10; 10(0.5/0.5/1) Denmark—425@-10; 10(0.25/0.25/1) New Zealand—400@-15; 10(0.25/0.25/1) Australia—425@-13; 10(0.375/0.375/1)

<FXO_Country_Setting_3_>	The country of deployment. This setting applies the relevant regional settings for PSTN calls. Default setting: USA
<Tip_Ring_Voltage_Adjust_3_>	Voltage adjustment. The choices are 3.1V, 3.2V, 3.35V, and 3.5V. Default setting: 3.5V.
<Ring_Frequency_Min_3_>	The lower limit of the ring frequency used to detect the ring signal. Default setting: 10
<SPA_To_PSTN_Gain_3_>	dB of digital gain (or attenuation if negative) to be applied to the signal sent from the ATA to the PSTN side. The range is -15 to 12. Default setting: 0
<Ring_Frequency_Max_3_>	The higher limit of the ring frequency used to detect the ring signal. Default setting: 100
<PSTN_To_SPA_Gain_3_>	dB of digital gain (or attenuation if negative) to be applied to the signal sent from the PSTN side to the ATA. The range is -15 to 12. Default setting: 0

<Ring_Validation_Time_3_>	The minimum signal duration required by the Gateway for recognition as a ring signal. Default setting: 256 ms
<Ring_Indication_Delay_3_>	Choose from {0, 512, 768, 1024, 1280, 1536, 1792} (ms). Default setting: 512ms
<Ring_Timeout_3_>	Choose from {0, 128, 256, 384, 512, 640, 768, 896, 1024, 1152, 1280, 1408, 1536, 1664, 1792, 1920} (ms). Default setting: 640 ms
<Ring_Threshold_3_>	Choose from {13.5–16.5, 19.35–2.65, 40.5–49.5} (Vrms). Default setting: 13.5-16.5 Vrms
<Line-In-Use_Voltage_3_>	The voltage threshold at which the ATA assumes the PSTN is in use by another handset sharing the same line (and will declare PSTN gateway service not available to incoming VoIP callers). Default setting: 30

<Dial_Tone>	Prompts the user to enter a phone number. Reorder Tone is played automatically when Dial Tone or any of its alternatives times out. Default setting: 350@-5,440@-5;10(*/0/1+2)
<Second_Dial_Tone>	Alternative to the Dial Tone when the user dials a three-way call. Default setting: 420@-5,520@-5;10(*/0/1+2)
<Outside_Dial_Tone>	Alternative to the Dial Tone. It prompts the user to enter an external phone number, as opposed to an internal extension. It is triggered by a comma character encountered in the dial plan. Default setting: 420@-4;10(*/0/1)
<Prompt_Tone>	Prompts the user to enter a call forwarding phone number. Default setting: 520@-5,620@-5;10(*/0/1+2)
<Busy_Tone>	Played when a 486 RSC is received for an outbound call. Default setting: 480@-5,620@-5;10(.5/.5/1+2)

<Reorder_Tone>	Played when an outbound call has failed, or after the far end hangs up during an established call. Reorder Tone is played automatically when Dial Tone or any of its alternatives times out. Default setting: 480@-5,620@-5;10(.25/.25/1+2)
<Off_Hook_Warning_Tone>	Played when the caller has not properly placed the handset on the cradle. Off Hook Warning Tone is played when the Reorder Tone times out. Default setting: 480@-3,620@3;10(.125/.125/1+2)
<Ring_Back_Tone>	Played during an outbound call when the far end is ringing. Default setting: 440@-5,480@-5;*(2/4/1+2)
<Ring_Back_2_Tone>	Your ATA plays this ringback tone instead of Ring Back Tone if the called party replies with a SIP 182 response without SDP to its outbound INVITE request. Default setting: the same as Ring Back Tone, except the cadence is 1s on and 1s off. Default setting: 440@-5,480@-5;*(1/1/1+2)
<Confirm_Tone>	Brief tone to notify the user that the last input value has been accepted. Default setting: 600@-4;1(.25/.25/1)
<SIT1_Tone> through <SIT4_Tone>	Alternative to the Reorder Tone played when an error occurs as a caller makes an outbound call. The RSC to trigger this tone is configurable on the SIP screen. Default setting: 985@-4,1428@-4,1777@-4;20(.380/0/1,.380/0/2,.380/0/3,0/4/0)
<MWI_Dial_Tone>	Played instead of the Dial Tone when there are unheard messages in the caller's mailbox. Default setting: 350@-5,440@-5;2(.1/.1/1+2);10(*/0/1+2)
<Cfwd_Dial_Tone_>	Played when all calls are forwarded. Default setting: 350@-5,440@-5;2(.2/.2/1+2);10(*/0/1+2)
<Holding_Tone>	Informs the local caller that the far end has placed the call on hold. Default setting: 600@-5;*(.1/.1/1,.1/.1/1,.1/9.5/1)
<Conference_Tone>	Played to all parties when a three-way conference call is in progress. Default setting: 350@-5;20(.1/.1/1,.1/9.7/1)

<Secure_Call_Indication_Tone>	Played when a call has been successfully switched to secure mode. It should be played only for a short while (less than 30 seconds) and at a reduced level (less than -19 dBm) so it does not interfere with the conversation. Default setting: 397@-5,507@-5;15(0/2/0,.2/.1/1,.1/2.1/2)
<VoIP_PIN_Tone>	This tone is played to prompt a VoIP caller to enter a PIN number.
<PSTN_PIN_Tone>	This tone is played to prompt a PSTN caller to enter a PIN number.
<Feature_Invocation_Tone>	Played when a feature is implemented. Default setting: 350@-4;*(.1/.1/1)

<Ring1_Cadence>	Cadence script for distinctive ring 1. Default setting: 60(2/4)
<Ring2_Cadence>	Cadence script for distinctive ring 2. Default setting: 60(.8/.4,.8/4)
<Ring3_Cadence>	Cadence script for distinctive ring 3. Default setting: 60(.4/.2,.4/.2,.8/4)
<Ring4_Cadence>	Cadence script for distinctive ring 4. Default setting: 60(.3/.2,1/.2,.3/4)
<Ring5_Cadence>	Cadence script for distinctive ring 5. Default setting: 1(.5/.5)
<Ring6_Cadence>	Cadence script for distinctive ring 6. Default setting: 60(.2/.4,.2/.4,.2/4)
<Ring7_Cadence>	Cadence script for distinctive ring 7. Default setting: 60(.4/.2,.4/.2,.4/4)
<Ring8_Cadence>	Cadence script for distinctive ring 8. Default setting: 60(0.25/9.75)

<CWT1_Cadence>	Cadence script for distinctive CWT 1. Default setting: 30(.3/9.7)

<CWT2_Cadence>	Cadence script for distinctive CWT 2. Default setting: 30(.1/.1, .1/9.7)
<CWT3_Cadence>	Cadence script for distinctive CWT 3. Default setting: 30(.1/.1, .1/.1, .1/9.7)
<CWT4_Cadence>	Cadence script for distinctive CWT 4. Default setting: 30(.1/.1, .3/.1, .1/9.3)
<CWT5_Cadence>	Cadence script for distinctive CWT 5. Default setting: 1(.5/.5)
<CWT6_Cadence>	Cadence script for distinctive CWT 6. Default setting: 30(.3/.1,.3/.1,.1/9.1)
<CWT7_Cadence>	Cadence script for distinctive CWT 7. Default setting: 30(.3/.1,.3/.1,.1/9.1)
<CWT8_Cadence>	Cadence script for distinctive CWT 8. Default setting: 2.3(.3/2)

<Ring1_Name>	Name in an INVITE's Alert-Info Header to pick distinctive ring/CWT 1 for the inbound call. Default setting: Bellcore-r1
<Ring2_Name>	Name in an INVITE's Alert-Info Header to pick distinctive ring/CWT 2 for the inbound call. Default setting: Bellcore-r2
<Ring3_Name>	Name in an INVITE's Alert-Info Header to pick distinctive ring/CWT 3 for the inbound call. Default setting: Bellcore-r3
<Ring4_Name>	Name in an INVITE's Alert-Info Header to pick distinctive ring/CWT 4 for the inbound call. Default setting: Bellcore-r4
<Ring5_Name>	Name in an INVITE's Alert-Info Header to pick distinctive ring/CWT 5 for the inbound call. Default setting: Bellcore-r5
<Ring6_Name>	Name in an INVITE's Alert-Info Header to pick distinctive ring/CWT 6 for the inbound call. Default setting: Bellcore-r6

<Ring7_Name>	Name in an INVITE's Alert-Info Header to pick distinctive ring/CWT 7 for the inbound call. Default setting: Bellcore-r7
<Ring8_Name>	Name in an INVITE's Alert-Info Header to pick distinctive ring/CWT 8 for the inbound call. Default setting: Bellcore-r8

<Ring_Waveform>	Waveform for the ringing signal. Choices are Sinusoid or Trapezoid. Default setting: Sinusoid
<Ring_Frequency>	Frequency of the ringing signal. Valid values are 10–100 (Hz) Default setting: 20
<Ring_Voltage>	Ringing voltage. Choices are 60–90 (V) Default setting: 85
<CWT_Frequency>	Frequency script of the call waiting tone. All distinctive CWTs are based on this tone. Default setting: 440@-10
<Synchronized_Ring>	If this is set to yes, when the ATA is called, all lines ring at the same time (similar to a regular PSTN line) After one line answers, the others stop ringing. Default setting: no

<Hook_Flash_Timer_Min>	Minimum on-hook time before off-hook qualifies as hook flash. Less than this the on-hook event is ignored. Range: 0.1–0.4 seconds. Default setting: 0.1
<Hook_Flash_Timer_Max>	Maximum on-hook time before off-hook qualifies as hook flash. More than this the on-hook event is treated as on hook (no hook-flash event) Range: 0.4–1.6 seconds. Default setting: 0.9
<Callee_On_<Hook_Delay>	Phone must be on-hook for at this time in sec. before the ATA will tear down the current inbound call. It does not apply to outbound calls. Range: 0–255 seconds. Default setting: 0

<Reorder_Delay>	Delay after far end hangs up before reorder tone is played. 0 = plays immediately, inf = never plays. Range: 0–255 seconds. Default setting: 5.
<Call_Back_Expires>	Expiration time in seconds of a call back activation. Range: 0–65535 seconds. Default setting: 1800
<Call_Back_Retry_Intvl>	Call back retry interval in seconds. Range: 0–255 seconds. Default setting: 30
<Call_Back_Delay>	Delay after receiving the first SIP 18x response before declaring the remote end is ringing. If a busy response is received during this time, the ATA still considers the call as failed and keeps on retrying. Default setting: 0.5
<VMWI_Refresh_Intvl>	Interval between VMWI refresh to the device. Default setting: 0
<Interdigit_Long_Timer>	Long timeout between entering digits when dialing. The interdigit timer values are used as defaults when dialing. The Interdigit_Long_Timer is used after any one digit, if all valid matching sequences in the dial plan are incomplete as dialed. Range: 0–64 seconds. Default setting: 10
<Interdigit_Short_Timer>	Short timeout between entering digits when dialing. The Interdigit_Short_Timer is used after any one digit, if at least one matching sequence is complete as dialed, but more dialed digits would match other as yet incomplete sequences. Range: 0–64 seconds. Default setting: 3

<CPC_Delay>	Delay in seconds after caller hangs up when the ATA starts removing the tip-and-ring voltage to the attached equipment of the called party. The range is 0–255 seconds. This feature is generally used for answer supervision on the caller side to signal to the attached equipment when the call has been connected (remote end has answered) or disconnected (remote end has hung up) This feature should be disabled for the called party (in other words, by using the same polarity for connected and idle state) and the CPC feature should be used instead. Without CPC enabled, reorder tone will is played after a configurable delay. If CPC is enabled, dial tone will be played when tip-to-ring voltage is restored. Resolution is 1 second. Default setting: 2
<CPC_Duration>	Duration in seconds for which the tip-to-ring voltage is removed after the caller hangs up. After that, tip-to-ring voltage is restored and the dial tone applies if the attached equipment is still off-hook. CPC is disabled if this value is set to 0. Range: 0 to 1.000 second. Resolution is 0.001 second. Default setting: 0 (CPC disabled)

<Call_Return_Code>	Call Return Code This code calls the last caller. Default setting: *69
<Call_Redial_Code>	Redials the last number called. Default setting: *07
<Blind_Transfer_Code>	Begins a blind transfer of the current call to the extension specified after the activation code. Default setting: *98
<Call_Back_Act_Code>	Starts a callback when the last outbound call is not busy. Default setting: *66
<Call_Back_Deact_Code>	Cancels a callback. Default setting: *86
<Call_Back_Busy_Act_Code>	Starts a callback when the last outbound call is busy. Default setting: *05

<Cfwd_All_Act_Code>	Forwards all calls to the extension specified after the activation code. Default setting: *72
<Cfwd_All_Deact_Code>	Cancels call forwarding of all calls. Default setting: *73
<Cfwd_Busy_Act_Code>	Forwards busy calls to the extension specified after the activation code. Default setting: *90
<Cfwd_Busy_Deact_Code>	Cancels call forwarding of busy calls. Default setting: *91
<Cfwd_No_Ans_Act_Code>	Forwards no-answer calls to the extension specified after the activation code. Default setting: *92
<Cfwd_No_Ans_Deact_Code>	Cancels call forwarding of no-answer calls. Default setting: *93
<Cfwd_Last_Act_Code>	Forwards the last inbound or outbound call to the number that the user specifies after entering the activation code. Default setting: *63
<Cfwd_Last_Deact_Code>	Cancels call forwarding of the last inbound or outbound call. Default setting: *83
<Block_Last_Act_Code>	Blocks the last inbound call. Default setting: *60
<Block_Last_Deact_Code>	Cancels blocking of the last inbound call. Default setting: *80
<Accept_Last_Act_Code>	Accepts the last outbound call. It lets the call ring through when do not disturb or call forwarding of all calls are enabled. Default setting: *64
<Accept_Last_Deact_Code>	Cancels the code to accept the last outbound call. Default setting: *84
<CW_Act_Code>	Enables call waiting on all calls. Default setting: *56
<CW_Deact_Code>	Disables call waiting on all calls. Default setting: *57

<CW_Per_Call_Act_Code>	Enables call waiting for the next call. Default setting: *71
<CW_Per_Call_Deact_Code>	Disables call waiting for the next call. Default setting: *70
<Block_CID_Act_Code>	Blocks caller ID on all outbound calls. Default setting: *67
<Block_CID_Deact_Code>	Removes caller ID blocking on all outbound calls. Default setting: *68
<Block_CID_Per_Call_Act_Code>	Blocks caller ID on the next outbound call. Default setting: *81
<Block_CID_Per_Call_Deact_Code>	Removes caller ID blocking on the next inbound call. Default setting: *82
<Block_ANC_Act_Code>	Blocks all anonymous calls. Default setting: *77
<Block_ANC_Deact_Code>	Removes blocking of all anonymous calls. Default setting: *87
<DND_Act_Code>	Enables the do not disturb feature. Default setting: *78
<DND_Deact_Code>	Disables the do not disturb feature. Default setting: *79
<CID_Act_Code>	Enables caller ID generation. Default setting: *65
<CID_Deact_Code>	Disables caller ID generation. Default setting: *85
<CWCID_Act_Code>	Enables call waiting, caller ID generation. Default setting: *25
<CWCID_Deact_Code>	Disables call waiting, caller ID generation. Default setting: *45
<Dist_Ring_Act_Code>	Enables the distinctive ringing feature. Default setting: *26
<Dist_Ring_Deact_Code>	Disables the distinctive ringing feature. Default setting: *46

<Speed_Dial_Act_Code>	Assigns a speed dial number. Default setting: *74
<Paging_Code>	Used for paging other clients in the group. Default setting: *96
<Secure_All_Call_Act_Code>	Makes all outbound calls secure. Default setting: *16
<Secure_No_Call_Act_Code>	Makes all outbound calls not secure. Default setting: *17
<Secure_One_Call_Act_Code>	Makes the next outbound call secure. (It is redundant if all outbound calls are secure by default.) Default setting: *18
<Secure_One_Call_Deact_Code>	Makes the next outbound call not secure. (It is redundant if all outbound calls are not secure by default.) Default setting: *19
<Conference_Act_Code>	If this code is specified, the user must enter it before dialing the third party for a conference call. Enter the code for a conference call. Default setting: blank
<Attn-Xfer_Act_Code>	If the code is specified, the user must enter it before dialing the third party for a call transfer. Enter the code for a call transfer. Default setting: blank
<Modem_Line_Toggle_Code>	Toggles the line to a modem. Modem passthrough mode can be triggered only by pre-dialing this code. Default setting: *99
<FAX_Line_Toggle_Code>	Toggles the line to a fax machine. Default setting: #99
<Media_Loopback_Code>	Use for media loopback. Default setting: *03

<Referral_Services_Code>s	These codes tell the ATA what to do when the user places the current call on hold and is listening to the second dial tone. One or more *codes can be configured into this parameter, such as *98, or *97|*98|*123, etc. The maximum length is 79 characters. This parameter applies when the user places the current call on hold by pressing the hook flash button. Each *code (and the following valid target number according to current dial plan) triggers the ATA to perform a blind transfer to a target number that is prepended by the service *code. For example, after the user dials *98, the ATA plays a special dial tone called the Prompt Tone while waiting for the user the enter a target number (which is checked according to dial plan as in normal dialing). When a complete number is entered, the ATA sends a blind REFER to the holding party with the Refer-To target equal to *98 target_number. This feature allows the ATA to hand off a call to an application server to perform further processing, such as call park. The *codes should not conflict with any of the other vertical service codes internally processed by the ATA. You can empty the corresponding *code that you do not want the ATA to process. Default setting: blank

<Feature_Dial_Services_Code>s	These codes tell the ATA what to do when the user is listening to the first or second dial tone. One or more *codes can be configured into this parameter, such as *72, or *72|*74|*67|*82, etc. The maximum length is 79 characters. This parameter applies when the user has a dial tone (first or second dial tone) After receiving dial tone, a user enters the *code and the target number according to current dial plan. For example, after user dials *72, the ATA plays a special tone called a Prompt tone while awaiting the user to enter a valid target number. When a complete number is entered, the ATA sends a INVITE to *72 target_number as in a normal call. This feature allows the proxy to process features like call forward (*72) or Block Caller ID (*67) The *codes should not conflict with any of the other vertical service codes internally processed by the ATA. You can remove a corresponding *code that you do not want to the ATA to process. You can add a parameter to indicate which tone plays after the *code is entered, such as *72‘c‘|*67‘p‘. Below is a list of allowed tone parameters (note the use of open quotes surrounding the parameter, without spaces) ‘c‘ = <Cfwd Dial Tone> ‘d‘ = <Dial Tone> ‘m‘ = <MWI Dial Tone> ‘o‘ = <Outside Dial Tone> ‘p‘ = <Prompt Dial Tone> ‘s‘ = <Second Dial Tone> ‘x‘ = No tones are place, x is any digit not used above If no tone parameter is specified, the ATA plays Prompt tone by default. If the *code is not to be followed by a phone number, such as *73 to cancel call forwarding, do not include this parameter. Instead, add the *code in the dial plan and the ATA send INVITE *73@..... as usual when user dials *73. Default setting: blank

<Service_Annc_Base_Number>	Base number for service announcements. Default setting: blank

<Service_Annc_Extension_Codes>	Extension codes for service announcements. Default setting: blank

<Prefer_G711u_Code>	Dial prefix to make G.711u the preferred codec for the call. Default setting: *017110
<Force_G711u_Code>	Dial prefix to make G.711u the only codec that can be used for the call. Default setting: *027110
<Prefer_G711a_Code>	Dial prefix to make G.711a the preferred codec for the call. Default setting: *017111
<Force_G711a_Code>	Dial prefix to make G.711a the only codec that can be used for the call. Default setting: *027111
<Prefer_G726r32_Code>	Dial prefix to make G.726r32 the preferred codec for the call. Default setting: *0172632
<Force_G726r32_Code>	Dial prefix to make G.726r32 the only codec that can be used for the call. Default setting: *0272632
<Prefer_G729a_Code>	Dial prefix to make G.729a the preferred codec for the call. Default setting: *01729
<Force_G729a_Code>	Dial prefix to make G.729a the only codec that can be used for the call. Default setting: *02729
<Prefer_G722_Code>	Dial prefix to make G.722 the preferred codec for the call. Default setting: *01722
<Force_G722_Code>	Dial prefix to make G.722 the only codec that can be used for the call. Default setting: *02722

<FXS_Port_Impedance>	Sets the electrical impedance of the PHONE port. Choices are: 600, 900, 600+2.16uF, 900+2.16uF, 270+750‖150nF, 220+850‖120nF, 220+820‖115nF, or 200+600‖100nF. Default setting: 600. **NOTE** For New Zealand impedance (370+620‖310nF), use 270+750‖150nF.
<FXS_Port_Input_Gain>	Input gain in dB, up to three decimal places. The range is 6.000 to -12.000. Default setting: -3.
<FXS_Port_Output_Gain>	Output gain in dB, up to three decimal places. The range is 6.000 to -12.000. The Call Progress Tones and DTMF playback level are not affected by the FXS Port Output Gain parameter. Default setting: -3.
<DTMF_Playback_Level>	Local DTMF playback level in dBm, up to one decimal place. Default setting: -16.0.
<DTMF_Playback_Length>	Local DTMF playback duration in milliseconds. Default setting: .1.
<Detect_ABCD>	To enable local detection of DTMF ABCD, select yes. Otherwise, select no. Default setting: yes This setting has no effect if DTMF Tx Method is INFO; ABCD is always sent OOB regardless in this setting.
<Playback_ABCD>	To enable local playback of OOB DTMF ABCD, select yes. Otherwise, select no. Default setting: yes

<Caller_ID_Method>	The choices are described below. Default setting: Bellcore (N.Amer, China) ▪ **Bellcore(N.Amer,China)**: CID, CIDCW, and VMWI. FSK sent after first ring (same as ETSI FSK sent after first ring) (no polarity reversal or DTAS) ▪ **DTMF(Finland, Sweden)**: CID only. DTMF sent after polarity reversal (and no DTAS) and before first ring. ▪ **DTMF(Denmark)**: CID only. DTMF sent before first ring with no polarity reversal and no DTAS. ▪ **ETSI DTMF:** CID only. DTMF sent after DTAS (and no polarity reversal) and before first ring. ▪ **ETSI DTMF With PR**: CID only. DTMF sent after polarity reversal and DTAS and before first ring. ▪ **ETSI DTMF After Ring**: CID only. DTMF sent after first ring (no polarity reversal or DTAS) ▪ **ETSI FSK: CID, CIDCW, and VMWI**. FSK sent after DTAS (but no polarity reversal) and before first ring. Waits for ACK from a device after DTAS for CIDCW. ▪ **ETSI FSK With PR (UK)**: CID, CIDCW, and VMWI. FSK is sent after polarity reversal and DTAS and before first ring. Waits for ACK from a device after DTAS for CIDCW. Polarity reversal is applied only if equipment is on hook. ▪ **DTMF (Denmark) with PR:** CID only. DTMF sent after polarity reversal (and no DTAS) and before first ring. Default setting: Bellcore(N.Amer, China)
<FXS_Port_Power_Limit>	The choices are from 1 to 8. Default setting: 3
<Caller_ID_FSK_Standard>	The ATA supports bell 202 and v.23 standards for caller ID generation. Default setting: bell 202
<Feature_Invocation_Method>	Select the method you want to use, Default or Sweden default. Default setting: Default

<DECT_Enable>	To enable this handset for service, select yes. Otherwise, select no. Default setting: yes
<Call_Park_Enable>	Enables or disables Call Park. Default setting: No
<Call_Pickup_Enable>	Enables or disables Call Pickup. Default setting: No
<Call_Group_Pickup_Enable>	Enables or disables Group Pickup. Default setting: No

<Outgoing_Lines>	A comma-separated list of the index numbers (1~10) for the lines that are available from this handset for an outgoing call. These lines will be listed on the phone screen when the user displays the call options or holds down the green call button. Example: 1,2,8 In this example, a user can select DECT line 1, 2, or 8 for an outbound call. Default setting: 1 **Note:** You also can choose these lines from the DECT Handset Outgoing Line Selection section of the *Quick Setup* page.
<Failover>	When this feature is enabled and a call fails through the selected line, the ATA automatically attempts to place the call over another enabled DECT line. Select yes to enable this feature or select no to disable it. Default setting: no
<Deregister>	To deregister a handset, select yes. After you submit the settings and the voice module reboots, then the handset is deregistered. At that point, this parameter is reset to the default value. Default setting: no

<Bound_IPEI>	Enter the device's IPEI number (a unique hardware identifier comparable to a MAC address) if you want to bind this device to the specified handset ID, such as Handset 3. The IPEI can be found in the Settings > Phone Info menu on the handset. Default setting: blank

7

Router Configuration Parameters

This chapter provides descriptions and examples of the parameters in the <router-configuration> section of the config.xml file for Cisco SPA100 ATAs.

See these topics:

Nested Structure

- All items in the <router_configuration> section of the XML file need to be nested under <router-configuration> and the section headings as shown in Nested Sections, page 148.
- The </router-configuration> tag must appear at the end of the section.
- In the XML file, each section can be opened or closed by clicking the section heading. A + symbol indicates that a section is closed, and a - symbol indicates that it is open.
- To enter a null value, enter a backslash at the end of the parameter name, as show in this example: `<MAC_Address_Clone_Address />`

Nested Sections

```
- <flat-profile>
    ...
    ...
- <router-configuration>
  + <WAN_Interface>
  + <PHY_Port_Setting>
  + <MAC_Address_Clone>
  + <Internet_Option>
  + <DHCP_Server_Pool>
  + <WAN_VLAN_Setting>
  + <CLDP_Setting>
  + <SNMP>
  + <Time_Setup>
  + <QoS_Bandwidth_Control>
  + <Software_DMZ>
    <Bonjour_Enable>1</Bonjour_Enable>
    <Reset_Button_Enable>0</Reset_Button_Enable>
    <Router_Mode>1</Router_Mode>
  + <VPN_Passthrough>
  + <Web_Management>
  + <TR_069>
  + <Log_Configuration>
    <Web_Login_Admin_Name>admin</Web_Login_Admin_Name>
    <!--  <Web_Login_Admin_Password></Web_Login_Admin_Password>  -->
    <Web_Login_Guest_Name>cisco</Web_Login_Guest_Name>
    <!--  <Web_Login_Guest_Password></Web_Login_Guest_Password>  -->
    <Firmware_Version>1.2.1</Firmware_Version>
    <Firmware_Build_Version>004</Firmware_Build_Version>
    <System_Model_Number>SPA122</System_Model_Number>
  + <About_Product>
  </router-configuration>
</flat-profile>
```

<WAN_Interface> WAN Interface Parameters

This section describes the parameters in the <WAN_Interface> section of the config.xml file.

TIP: You can click the <WAN_Interface> heading in the XML file to expand or collapse the nested parameters in this section.

NOTE In addition to the descriptions, also refer to **WAN Example 1: DHCP with automatic MTU mode, page 153**, **WAN Example 2: Static IP with manual MTU mode, page 154**, and **WAN Example 3: PPPoE with Connect on Demand, page 154**.

Parameter	Details
<WAN_Connection_Type>	**Description:** Defines the connection/addressing mode used for the INTERNET (WAN) port. **User Interface:** *Network Setup > Basic Setup > Internet Settings* page, *Connection Type* field **Values**: ▪ dh: DHCP ▪ st: Static ▪ pp: PPPoE **Default**: dh **Example: Static connection type** `<WAN_Connection_Type>st</WAN_Connection_Type>`

Parameter	Details
<WAN_DHCP_MTU_Mode> <WAN_Static_MTU_Mode> <WAN_PPPoE_MTU_Mode>	**Description:** MTU mode. Use the parameter corresponding to the configured connection type. **User Interface:** *Network Setup > Basic Setup > Internet Settings* page, *MTU* drop-down list **Values**: ▪ 0: Auto ▪ 1: Manual **Default**: 0 **Example: Manual MTU mode for a static connection** `<WAN_Static_MTU_Mode>1</WAN_Static_MTU_Mode>`
<WAN_DHCP_MTU_Size> <WAN_Static_MTU_Size> <WAN_PPPoE_MTU_Size>	**Description:** MTU size. Use the parameter corresponding to the configured connection type. **User Interface:** *Network Setup > Basic Setup > Internet Settings* page, *MTU* text box **Values**: 576 to1492 **Default**: 0 **Example: Customized MTU size for PPPoE** `<WAN_PPPoE_MTU_Size>1492</WAN_PPPoE_MTU_Size>`

Parameter	Details
<WAN_Static_IP_NET>	**Description:** Specifies the IPv4 address for the Static IP connection. **User Interface:** *Interface Setup > Basic Setup > Internet Settings* page, *Internet IPv4 address* text box (available when Static IP is the Connection Type) **Parameters:** Internet_IP:Subnet_Mask:Default_Gateway [:DNS1[:DNS2[:DNS3]]] **Values**: Internet_IP: IPv4 address Subnet_Mask: IPv4 mask address Default_Gateway: IPv4 address DNS_1: IPv4 address DNS_2: IPv4 address DNS_3: IPv4 address **Default**: 0.0.0.0:0.0.0.0:0.0.0.0:0.0.0.0:0.0.0.0:0.0.0.0 **Example:** `<WAN_Static_IP_NET> 10.1.1.1:255.255.255.0:10.1.1.254:10.1.1.2:10.1.1.3</WAN_Static_IP_NET>`
<WAN_PPPoE_User_Name>	**Description**: Username for PPTP session through the INTERNET (WAN) port. **User Interface:** *Interface Setup > Basic Setup > Internet Settings* page, *User Name* field (available when PPPoE is the Connection Type) **Values**: (up to 64 characters), Printable ASCII characters **Default**: null **Example:** `<WAN_PPPoE_User_Name>test@example.net</WAN_PPPoE_User_Name>`

Parameter	Details
<WAN_PPPoE_Password>	**Description**: Configures the interface settings for defined VLAN sub interfaces. VLAN ID *n* must be previously defined in the VLAN_ID_ Index tag. This tag defines the password for PPPoE session configured over the sub interface. Note: the value of this field is hidden when reading the config.xml file from the device. **User Interface:** *Interface Setup > Basic Setup > Internet Settings* page, *Password* field (available when PPPoE is the Connection Type) **Values**: password (up to 64 characters) **Default**: commented out, `<!-- <WAN_PPPoE_Password></WAN_PPPoE_Password>-->` **Example:** `<WAN_PPPoE_Password>my-password</WAN_PPPoE_Password>`
<WAN_PPPoE_Service_Name>	**Description:** Descriptive service name (provided by the ISP), for a PPPoE session. **User Interface:** *Interface Setup > Basic Setup > Internet Settings* page, *Service Name* field (available when PPPoE is the Connection Type) **Parameter:** Service name **Values**: name (up to 64 characters) **Default**: null **Example:** `<WAN_PPPoE_Service_Name>ServiceX_PPP</WAN_PPPoE_Service_Name>`

Parameter	Details
<WAN_PPPoE_Keep_Alive>	**User Interface:** *Interface Setup* > *Basic Setup* > *Internet Settings* page, *Keep Alive* field and *Connect on Demand* field (available when PPPoE is the Connection Type) **Description:** Keep Alive or Connect on Demand settings for a PPPoE session configured. **Parameter:** Type:Max_Idle_Time:30 **Values**: 0 (Keep Alive) 0 (Connect on Demand) Max_Idle_Time (Minutes)= 1...9999 (for Connect on Demand) 30 is a static value **Default**: 0:0:30 **Example:** <WAN_PPPoE_Keep_Alive>1:5:30</WAN_PPPoE_Keep_Alive>

WAN Example 1: DHCP with automatic MTU mode

```
<router-configuration>
  <WAN_Interface>
    <WAN_Connection_Type>dh</WAN_Connection_Type>
    <WAN_DHCP_MTU_Mode>0</WAN_DHCP_MTU_Mode>
    <WAN_DHCP_MTU_Size>0</WAN_DHCP_MTU_Size>
    <WAN_Static_IP_NET>0.0.0.0:0.0.0.0:0.0.0.0</WAN_Static_IP_NET>
    <WAN_Static_MTU_Mode>0</WAN_Static_MTU_Mode>
    <WAN_Static_MTU_Size>0</WAN_Static_MTU_Size>
    <WAN_PPPoE_User_Name />
    <WAN_PPPoE_Service_Name />
    <WAN_PPPoE_Password />
    <WAN_PPPoE_Keep_Alive>0:0:30</WAN_PPPoE_Keep_Alive>
    <WAN_PPPoE_MTU_Mode>0</WAN_PPPoE_MTU_Mode>
    <WAN_PPPoE_MTU_Size>0</WAN_PPPoE_MTU_Size>
  </WAN_Interface>
...
</router-configuration>
```

WAN Example 2: Static IP with manual MTU mode

```
<router-configuration>
...
  <WAN_Interface>
    <WAN_Connection_Type>st</WAN_Connection_Type>
    <WAN_DHCP_MTU_Mode>0</WAN_DHCP_MTU_Mode>
    <WAN_DHCP_MTU_Size>0</WAN_DHCP_MTU_Size>
    <WAN_Static_IP_NET>10.1.1.1:255.255.255.0:10.1.1.254:10.1.1.2:10.1.1.3</
WAN_Static_IP_NET>
    <WAN_Static_MTU_Mode>1</WAN_Static_MTU_Mode>
    <WAN_Static_MTU_Size>1492</WAN_Static_MTU_Size>
  </WAN_Interface>
...
</router-configuration>
```

WAN Example 3: PPPoE with Connect on Demand

```
<router-configuration>
...
  <WAN_Interface>
    <WAN_Connection_Type>pppoe</WAN_Connection_Type>
    <WAN_DHCP_MTU_Mode>0</WAN_DHCP_MTU_Mode>
    <WAN_DHCP_MTU_Size>0</WAN_DHCP_MTU_Size>
    <WAN_Static_IP_NET>0.0.0.0:0.0.0.0:0.0.0.0</WAN_Static_IP_NET>
    <WAN_Static_MTU_Mode>0</WAN_Static_MTU_Mode>
    <WAN_Static_MTU_Size>0</WAN_Static_MTU_Size>
    <WAN_PPPoE_User_Name>test@example.net</WAN_PPPoE_User_Name>
    <WAN_PPPoE_Password>my-password</WAN_PPPoE_Password>
    <WAN_PPPoE_Service_Name>ServiceX_PPP</WAN_PPPoE_Service_Name>
    <WAN_PPPoE_Keep_Alive>1:5:30</WAN_PPPoE_Keep_Alive>
    <WAN_PPPoE_MTU_Mode>0</WAN_PPPoE_MTU_Mode>
    <WAN_PPPoE_MTU_Size>0</WAN_PPPoE_MTU_Size>
  </WAN_Interface>
...
</router-configuration>
```

<PHY_Port_Setting> Parameters

This section describes the parameters in the <PHY_Port_Setting> section of the config.xml file.

TIP: You can click the <PHY_Port_Setting> heading in the XML file to expand or collapse the nested parameters in this section.

NOTE In addition to the descriptions, also refer to**<PHY_Port_Setting> Example: Flow control enabled with auto-negotiated duplex mode, page 156**.

Parameter	Details
<Flow_Control>	**Description:** Enables or disables flow control **User Interface:** *Interface Setup* > *Advanced Settings* > *Port Setting* page, *Flow Control* field **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example: Flow control enabled** `<Flow_Control>1</Flow_Control>`

Parameter	Details
<Speed_Duplex>	**Description:** The port speed and duplex mode **User Interface:** *Interface Setup > Advanced Settings > Port Setting* page, *Speed Duplex* field **Values**: auto 10h 10f 100h 100f **Default**: auto **Example: 100 Mbps, half-duplex mode** `<Speed_Duplex>100h</Speed_Duplex>`

<PHY_Port_Setting> Example: Flow control enabled with auto-negotiated duplex mode

```
<router-configuration>
...
  <PHY_Port_Setting>
    <Flow_Control>1</Flow_Control>
    <Speed_Duplex>auto</Speed_Duplex>
  </PHY_Port_Setting>
...
</router-configuration>
```

<MAC_Address_Clone> Parameters

This section describes the parameters in the <MAC_Address_Clone> section of the config.xml file.

TIP: You can click the <MAC_Address_Clone> heading in the XML file to expand or collapse the nested parameters in this section.

NOTE In addition to the descriptions, also refer to**<MAC_Address_Clone> Example: MAC Address Clone enabled, page 158**.

Parameter	Details
<MAC_Address_Clone_Enabled>	**Description:** Enables or disables MAC address cloning. **User Interface:** *Interface Setup > Advanced Settings > MAC Address Clone* page, *MAC Clone* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: MAC clone enabled** `<MAC_Address_Clone_Enabled>1</MAC_Address_Clone_Enabled>`
<MAC_Address_Clone_Address>	**Description:** MAC address to assign (clone) to this ATA **User Interface:** *Interface Setup > Advanced Settings > MAC Address Clone* page, *MAC Address* field (available when MAC Clone is enabled) **Values**: MAC address **Default**: null **Example:** `<MAC_Address_Clone_Address>00:22:68:19:EF:83</MAC_Address_Clone_Address>`

<MAC_Address_Clone> Example: MAC Address Clone enabled

```
<router-configuration>
...
  <MAC_Address_Clone>
    <MAC_Address_Clone_Enabled>1</MAC_Address_Clone_Enabled>
    <MAC_Address_Clone_Address>00:22:68:19:EF:83</MAC_Address_Clone_Address>
  </MAC_Address_Clone>
...
</router-configuration>
```

<Internet_Option> Parameters

This section describes the parameters in the <Internet_Option> section of the config.xml file.

TIP: You can click the <Internet_Option> heading in the XML file to expand or collapse the nested parameters in this section.

NOTE In addition to the descriptions, also refer to **<Internet_Option> Example:, page 160**.

Parameter	Details
<Host_Name>	**Description:** The name of the ATA **User Interface:** *Network Setup > Basic Setup > Internet Settings* page, *Host Name* field **Values**: name **Default**: model number **Example:** <Host_Name>SPA112</Host_Name>

Parameter	Details
<Domain_Name>	**Description:** A domain name specified by the ISP, if applicable **User Interface:** *Network Setup > Basic Setup > Internet Settings* page, *Domain Name* field **Values**: name **Default**: null **Example:** `<Domain_Name>My ISP</Domain_Name>`
<DNS_Order>	**Description:** Method for choosing a DNS server **User Interface:** *Network Setup > Basic Setup > Internet Settings* page, *DNS Server Order* field **Values**: 0:Manual 1:Manual-DHCP 2:DHCP-Manual **Default**: 2 **Example: Manual-DHCP order** `<DNS_Order>2</DNS_Order>`
<DNS>	**Description:** For manual DNS server order, the IPv4 address of a DNS server; optionally, a secondary server can be specified **User Interface:** *Network Setup > Basic Setup > Internet Settings* page, *Primary DNS* and *Secondary DNS* fields **Values**: DNS1[:DNS2] **Default**: null **Example: Primary and secondary DNS server** `<DNS>209.165.201.1:209.165.201.2</DNS>`

<Internet_Option> Example:

```
<router-configuration>
...
  <Internet_Option>
    <Host_Name>SPA112</Host_Name>
    <Domain_Name>My ISP</Domain_Name>
    <DNS_Order>2</DNS_Order>
    <DNS>209.165.201.1:209.165.201.2</DNS>
  </Internet_Option>
...
</router-configuration>
```

<DHCP_Server_Pool> Parameters

This section describes the parameters in the <DHCP_Server_Pool> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<DHCP_Server_Pool> Example: DHCP enabled with two DHCP reservations, page 166**.

<Rule>

All parameters in the <DHCP_Server> section of the XML file are nested between <Rule> and </Rule>.

Parameter	Details
<DHCP_Server>	**Description:** Enables or disables the DHCP server **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *DHCP Server* field **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example: DHCP server enabled** <DHCP_Server>1</DHCP_Server>

Parameter	Details
<Local_IP>	**Description:** The IPv4 address of the LAN interface **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Local IPv4 address* field **Values**: IPv4 address **Default**: 192.168.15.1 **Example:** `<Local_IP>192.168.15.1</Local_IP>`
<Subnet_Mask>	**Description:** The subnet mask for the local network **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Subnet Mask* field **Values**: Class C subnet mask 255.255.255.0 255.255.255.128 255.255.255.192 255.255.255.224 255.255.255.240 255.255.255.248 255.255.255.252 **Default**: 255.255.255.0 **Example:** `<Subnet_Mask>255.255.255.0</Subnet_Mask>`

Parameter	Details
<DHCP_Client_Table>	**Description:** Clients with reserved IPv4 addresses **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Clients Already Reserved* list (available after clicking the Show DHCP Reservation button **Values**: Semi-colon separated list of client information in the following order: *<MAC address> <ip_address>* **on** *<client_name>* **Default**: null **Example:** `<DHCP_Client_Table>58:8D:09:72:73:DA 192.168.15.100 on Computer-1;00:22:68:19:EF:83 192.168.15.101 on Computer-2;</DHCP_Client_Table>`
<Option_66>	**Description:** Method for specifying a TFTP server for remote configuration of the ATA **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Option 66* field **Values**: 0: None 2: Remote TFTP Server 3: Manual TFTP Server **Default**: 0 **Example: Remote TFTP server** `<Option_66>2</Option_66>`

Parameter	Details
<TFTP_IP>	**Description:** IPv4 address of a TFTP server, if Option 66 is set to Manual **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *TFTP Server* field **Values**: IPv4 address **Default**: 0.0.0.0 **Example:** `<TFTP_IP>209.165.202.129</TFTP_IP>`
<Option_67>	**Description:** Provides a configuration/bootstrap filename to hosts that request this option **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Option 67* field **Values**: filename **Default**: null Example: `<Option_67>MyDirectory/MyFile.cfg</Option_67>`
<Option_159 >	**Description:** Provides a configuration URL to hosts that request this option **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Option 159* field **Values**: URL **Default**: null **Example:** `<Option_159>http://MyDomain.com/MyDirectory/MyFile.cfg></Option_159>`

Parameter	Details
<Option_160 >	**Description:** Provides a configuration URL to hosts that request this option **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Option 160* field **Values**: filename **Default**: null **Example:** `<Option_67>MyDirectory/MyFile.cfg</Option_67>`
<DNS_Proxy>	**Description:** Enables or disables the DNS proxy, which relays DNS requests to the current public network DNS server for the proxy, and replies as a DNS resolver to the client device on the network **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *DNS Proxy* field **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example: DNS proxy enabled** `<DNS_Proxy>1</DNS_Proxy>`
<Starting_IP>	**Description:** The first IPv4 address in the range of IPv4 addresses that are assigned by the DHCP server **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Starting IPv4 address* field **Values**: IPv4 address **Default**: 192.168.15.100 **Example:** `<Starting_IP>192.168.15.110</Starting_IP>`

Parameter	Details
<Max_DHCP_User>	**Description:** The maximum number of devices that can receive DHCP addresses from the DHCP server **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Maximum DHCP Users* field **Values**: number **Default**: 50 **Example: 10-device maximum** `<Max_DHCP_User>10</Max_DHCP_User>`
<Client_Lease_Time>	**Description**: The number of minutes that a dynamically assigned IPv4 address can be in use, or "leased" **User Interface:** *Network Setup > Basic Setup > Network Settings* page, *Client Lease Time* field **Values**: number Enter the number of minutes. Enter 0 to represent 1 day. Enter 9999 to never expire. **Default**: 0 (1 day) **Example: No expiration** `<Client_Lease_Time>9999</Client_Lease_Time>`
<Static_DNS>	**Description**: Defines a DNS server address that will be provided to DHCP clients. If DNS Proxy is enabled, clients will automatically be issued the Local IPv4 address to use for DNS. **Values**: IPv4 address **Default**: 0.0.0.0 **Example:** `<Static_DNS>209.165.202.129</Static_DNS>`

Parameter	Details
<Default_Gateway>	**Description**: Enter the IPv4 address of the default gateway to be used by the DHCP clients. **Default:** 192.168.15.1 **Example:** `<Default_Gateway>192.168.15.1</Default_Gateway>`

<DHCP_Server_Pool> Example: DHCP enabled with two DHCP reservations

```
<router-configuration>
...
  <DHCP_Server_Pool>
    <Rule>
      <DHCP_Server>1</DHCP_Server>
      <Local_IP>192.168.15.1</Local_IP>
      <Subnet_Mask>255.255.255.0</Subnet_Mask>
      <DHCP_Client_Table>58:8D:09:72:73:DA 192.168.15.100 on Computer-
1;00:22:68:19:EF:83 192.168.15.101 on Computer-2;</DHCP_Client_Table>
      <TFTP_IP>0.0.0.0</TFTP_IP>
      <Starting_IP>192.168.15.100</Starting_IP>
      <Max_DHCP_User>50</Max_DHCP_User>
      <Client_Lease_Time>0</Client_Lease_Time>
      <Default_Gateway>192.168.15.1</Default_Gateway>
    </Rule>
  </DHCP_Server_Pool>
...
</router-configuration>
```

<WAN_VLAN_Setting> Parameters

This section describes the parameters in the <WAN_VLAN_Setting> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<WAN_VLAN_Setting> Example: VLAN Enabled with ID 100, page 168**.

Parameter	Details
<WAN_VLAN_Enable>	**Description:** Enables or disables a VLAN on your network **User Interface:** *Network Setup* > *Advanced Settings* > *VLAN* page, *Enable VLAN* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: VLAN enabled** `<WAN_VLAN_Enable>1</WAN_VLAN_Enable>`
<WAN_VALN_ID>	**Description:** A number that identifies the VLAN **User Interface:** *Network Setup* > *Advanced Settings* > *VLAN* page, *VLAN ID* field **Values**: 1~4094 **Default**: 1 **Example: VLAN ID 100** `<WAN_VALN_ID>100</WAN_VALN_ID>`

<WAN_VLAN_Setting> Example: VLAN Enabled with ID 100

```
<router-configuration>
...
  <WAN_VLAN_Setting>
    <WAN_VLAN_Enable>1</WAN_VLAN_Enable>
    <WAN_VALN_ID>100</WAN_VALN_ID>
  </WAN_VLAN_Setting>
...
</router-configuration>
```

<CLDP_Setting> Parameters

This section describes the parameters in the <CLDP_Setting> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<CLDP_Setting> Example: CDP, LLDP, and Layer 2 logging enabled, page 169**.

Parameter	Details
<CDP_ENABLE>	**Description:** Enables or disables Cisco Discovery Protocol **User Interface:** *Network Setup > Advanced Settings > CDP & LLDP* page, *Enable CDP* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: CDP enabled** <CDP_ENABLE>1</CDP_ENABLE>

Parameter	Details
<LLDP_ENABLE>	**Description:** Enables or disables LLDP **User Interface:** *Network Setup > Advanced Settings > CDP & LLDP* page, *Enable LLDP-MED* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: LLDP enabled** `<LLDP_ENABLE>1</LLDP_ENABLE>`
<LAYER2_LOGGING_ENABLE>	**Description:** Enables Layer 2 logging, which is used by CDP and LLDP for debugging purposes **User Interface:** *Network Setup > Advanced Settings > CDP & LLDP* page, *Layer 2 Logging* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: Layer 2 logging enabled** `<LAYER2_LOGGING_ENABLE>1</LAYER2_LOGGING_ENABLE>`

<CLDP_Setting> Example: CDP, LLDP, and Layer 2 logging enabled

```
<router-configuration>
...
  <CLDP_Setting>
    <CDP_ENABLE>1</CDP_ENABLE>
    <LLDP_ENABLE>1</LLDP_ENABLE>
    <LAYER2_LOGGING_ENABLE>1</LAYER2_LOGGING_ENABLE>
  </CLDP_Setting>
...
</router-configuration>
```

<SNMP> Parameters

This section describes the parameters in the <SNMP> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<SNMP> Example 1: SNMP Enabled from Any IP Address, page 174** and **<SNMP> Example 2: SNMPv3 Enabled from Trusted IP Address, page 175**.

Parameter	Details
<SNMP_Enabled>	**Description:** Enables or disables SNMP **User Interface:** *Administration > Management > SNMP* page, *SNMP* section, *Enabled* and *Disabled* options **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: SNMP enabled** `<SNMP_Enabled>1</SNMP_Enabled>`
<SNMP_Trusted_IP>	**Description:** IPv4 address and subnet mask of a single SNMP manager or trap agent that can access the ATA through SNMP **User Interface:** *Administration > Management > SNMP* page, *SNMP* section, *Trusted IP* field **Values**: IPv4 address and subnet mask in this order: 0.0.0.0/0.0.0.0 **Default**: 0.0.0.0/0.0.0.0 (Any IP address) **Example:** `<SNMP_Trusted_IP>209.165.202.129/255.255.255.0</SNMP_Trusted_IP>`

Parameter	Details
<Get_Community>	**Description:** A community string for authentication for SNMP GET commands. **User Interface:** *Administration > Management > SNMP* page, *SNMP* section, *Get/Trap Community* field **Values**: string **Default**: public **Example:** `<Get_Community>MyGet</Get_Community>`
<Set_Community>	**Description:** A community string for authentication for SNMP GET commands. **User Interface:** *Administration > Management > SNMP* page, *SNMP* section, *Set Community* field **Values**: string **Default**: private **Example:** `<Set_Community>MySet</Set_Community>`
<SNMPV3>	**User Interface:** *Administration > Management > SNMP* page, *SNMPV3* section, *Enable* and *Disable* fields **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: SNMPv3 enabled** `<SNMPV3>1</SNMPV3>`

Parameter	Details
<RW_User>	**Description:** A username for SNMP authentication **User Interface:** *Administration > Management > SNMP* page, *SNMPV3* section, *R/W User* field **Values**: username **Default**: v3rwuser **Example:** `<RW_User>MyUsername</RW_User>`
<Auth_Protocol>	**Description:** SNMPv3 authentication protocol **User Interface:** *Administration > Management > SNMP* page, *SNMPV3* section, *Auth-Protocol* field **Values**: MD5 SHA **Default**: MD5 **Example: SHA enabled** `<Auth_Protocol>SHA</Auth_Protocol>`
<Auth_Password>	**Description:** Password for SNMPv3 authentication **User Interface:** *Administration > Management > SNMP* page, *Auth-Password* field for SNMPv3 **Values**: string **Default**: 1111111111 **Example:** `<Auth_Password>MyPassword</Auth_Password>`

Parameter	Details
<Privacy_Protocol>	**Description:** Privacy authentication protocol for SNMPv3 **User Interface:** *Administration > Management > SNMP* page, *SNMPV3* section, *privprotocol* field **Values**: None DES **Default**: DES **Example: DES enabled** `<Privacy_Protocol>DES</Privacy_Protocol>`
<Privacy_Password>	**Description:** Privacy authentication password for SNMPv3 **User Interface:** *Administration > Management > SNMP* page, *SNMPV3* section, *Privacy Password* field **Values**: string **Default**: 1111111111 **Example:** `<Privacy_Password>MyPrivacyPassword</Privacy_Password>`
<TRAP_IP_Address>	**Description:** The IP Address of the SNMP manager or trap agent **User Interface:** *Administration > Management > SNMP* page, *Trap Configuration* section, *IP Address* field **Values**: IPv4 address **Default**: 192.168.15.100 **Example:** `<TRAP_IP_Address>209.165.202.129</TRAP_IP_Address>`

Parameter	Details
<TRAP_Port>	**Description:** The SNMP trap port used by the SNMP manager or trap agent to receive the trap messages **User Interface:** *Administration* > *Management* > *SNMP* page, *Trap Configuration* section, *Port* field **Values**: 162 or 1025~65535 **Default**: 162 **Example:** <TRAP_Port>162</TRAP_Port>
<TRAP_SNMP_Version>	**Description:** The SNMP version in use by the SNMP manager or trap agent **User Interface:** *Administration* > *Management* > *SNMP* page, *Trap Configuration* section, *SNMP Version* field **Values**: One of the SNMP version number listed below v1 v2c v3 **Default**: v1 **Example:** <TRAP_SNMP_Version>v3</TRAP_SNMP_Version>

<SNMP> Example 1: SNMP Enabled from Any IP Address

```
<router-configuration>
...
    <SNMP>
      <SNMP_Enabled>1</SNMP_Enabled>
      <SNMP_Trusted_IP>0.0.0.0/0.0.0.0</SNMP_Trusted_IP>
      <Get_Community>MyGet</Get_Community>
      <Set_Community>MySet</Set_Community>
      <TRAP_IP_Address>209.165.202.129</TRAP_IP_Address>
      <TRAP_Port>162</TRAP_Port>
      <TRAP_SNMP_Version>v3</TRAP_SNMP_Version>
    </SNMP>
...
```

```
</router-configuration>
```

<SNMP> Example 2: SNMPv3 Enabled from Trusted IP Address

```
<router-configuration>
...
    <SNMP>
      <SNMP_Enabled>1</SNMP_Enabled>
      <SNMP_Trusted_IP>209.165.202.129/255.255.255.0</SNMP_Trusted_IP>
      <Get_Community>MyGet</Get_Community>
      <Set_Community>MySet</Set_Community>
      <SNMPV3>1</SNMPV3>
      <RW_User>MyUsername</RW_User>
      <Auth_Protocol>SHA</Auth_Protocol>
      <Auth_Password>MyPassword</Auth_Password>
      <Privacy_Protocol>DES</Privacy_Protocol>
      <Privacy_Password>MyPrivacyPassword</Privacy_Password>
      <TRAP_IP_Address>209.165.201.1</TRAP_IP_Address>
      <TRAP_Port>162</TRAP_Port>
      <TRAP_SNMP_Version>v3</TRAP_SNMP_Version>
    </SNMP>
...
<router-configuration>
```

<Time_Setup> Parameters

This section describes the parameters in the <SNMP> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<Time_Setup> Example: Germany Time Zone with Daylight Savings and Auto-Recovery Enabled, page 178**.

Parameter	Details
<Time_Zone>	**Description:** The time zone for the site where the ATA is in operation **User Interface:** *Network Setup > Basic Setup > Time Settings* page, Time Zone field **Values**: number identifying the time zone See **Appendix B, "Time Zone Settings."**. **Default**: -08 1 1 **Example: Germany** `<Time_Zone>+01 2 2</Time_Zone>`
<Auto_Adjust_Clock>	**Description:** Enables or disables automatic time adjustments for daylight savings time **User Interface:** *Network Setup > Basic Setup > Time Settings* page, *Adjust Clock for Daylight Saving Changes* field **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example: Automatic Daylight Saving adjustment enabled** `<Auto_Adjust_Clock>1</Auto_Adjust_Clock>`

Parameter	Details
<Time_Server_Mode>	**Description:** The method for specifying an NTP time server Time Server Address **User Interface:** *Network Setup > Basic Setup > Time Settings* page, *Time Server* field **Values**: manual auto **Default**: auto **Example: Manual mode** `<Time_Server_Mode>manual</Time_Server_Mode>`
<Time_Server>	**Description:** IPv4 address or domain name of an NTP server **User Interface:** *Network Setup > Basic Setup > Time Settings* page, *Time Server Address* field **Values**: IPv4 address or domain name **Default**: 0.ciscosb.pool.ntp.org **Example: European pool** `<Time_Server>server 0.europe.pool.ntp.org </Time_Server>`
<Resync_Timer>	**Description:** The interval, in seconds, at which the ATA resynchronizes with the NTP server **User Interface:** *Network Setup > Basic Setup > Time Settings* page, *Resync Timer* field **Values**: number **Default**: 3600 **Example:** `<Resync_Timer>3600</Resync_Timer>`

Parameter	Details
<Auto_Recovery_System_Time>	**Description:** When enabled, allows the ATA to automatically reconnect to the time server after a system reboot **User Interface:** *Network Setup > Basic Setup > Time Settings* page, *Auto Recovery After System Reboot* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: Auto Recovery enabled** `<Auto_Recovery_System_Time>1</Auto_Recovery_System_Time>`
<Time_Mode>	**Description:** The method of specifying a time server **User Interface:** *Network Setup > Basic Setup > Time Settings* page, *Time Server* field **Values**: 0: Manual 1: Auto **Default**: 1 **Example: Automatic mode** `<Time_Mode>1</Time_Mode>`

<Time_Setup> Example: Germany Time Zone with Daylight Savings and Auto-Recovery Enabled

```
<router-configuration>
...
  <Time_Setup>
    <Time_Zone>+01 2 2</Time_Zone>
    <Auto_Adjust_Clock>1</Auto_Adjust_Clock>
    <Time_Server_Mode>auto</Time_Server_Mode>
    <Time_Server>0.ciscosb.pool.ntp.org</Time_Server>
    <Resync_Timer>3600</Resync_Timer>
    <Auto_Recovery_System_Time>1</Auto_Recovery_System_Time>
    <Time_Mode>1</Time_Mode>
```

```
  </Time_Setup>
...
<router-configuration>
```

<QoS_Bandwidth_Control> Parameters

This section describes the parameters in the <QoS_Bandwidth_Control> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<QoS_Bandwidth_Control> Example: QoS always on, maximum bandwidth of 20,000 kbps, page 180**.

<WAN>

All parameters in the <Qos_Bandwidth_Control> section are nested between <WAN> and </WAN>.

Parameter	Details
<QoS_Always_ON>	**Description:** Determines whether QoS settings are enabled at all times or only when there is voice traffic **User Interface:** *Network Setup* > *Application* > *QoS* page, *QoS Policy* field **Values**: 0: On When Phone In Use 1: Always On **Default**: 0 **Example: On when phone is in use** <QoS_Always_ON>0</QoS_Always_ON>

Parameter	Details
<Upstream_Bandwidth>	**Description:** The maximum available upstream bandwidth, in kbps, as specified by the Internet Service Provider **User Interface:** *Network Setup* > *Application* > *QoS* page, *Upstream Bandwidth* field **Values**: number **Default**: 10000 **Example:** <Upstream_Bandwidth>20000</Upstream_Bandwidth>

<QoS_Bandwidth_Control> Example: QoS always on, maximum bandwidth of 20,000 kbps

```
<router-configuration>
...
  <QoS_Bandwidth_Control>
    <WAN>
      <QoS_Always_ON>1</QoS_Always_ON>
      <Upstream_Bandwidth>20000</Upstream_Bandwidth>
    </WAN>
  </QoS_Bandwidth_Control>
...
</router-configuration>
```

<Software_DMZ> Parameters

This section describes the parameters in the <SNMP> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<Software_DMZ> Example: DMZ allowing Internet traffic to access 192.168.15.101, page 182**.

<Rule1>

All parameters in the <Software_DMZ> section are nested between <Rule1> and </Rule1>. Only one DMZ rule is allowed on this device.

Parameter	Details
<Status>	**Description:** Enables or disables exposing a local device to the Internet for a special-purpose service **User Interface:** *Network Setup > Application > DMZ* page, *Status* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: DMZ enabled** `<Status>1</Status>`
<Private_IP>	**Description:** The local IPv4 address of the device that can be accessed through the DMZ **User Interface:** *Network Setup > Application > DMZ* page, *Private IP* field **Values**: IPv4 address **Default**: 0.0.0.0 **Example:** `<Private_IP>192.168.15.1</Private_IP>`
<Rule_Number>	**Description:** A static setting used to define the DMZ rule **User Interface:** not applicable **Values**: 1 (do not change this number) **Default**: 1

<Software_DMZ> Example: DMZ allowing Internet traffic to access 192.168.15.101

```
<router-configuration>
...
    <Software_DMZ>
      <Rule1>
        <Status>1</Status>
        <Private_IP>192.168.15.1</Private_IP>
      </Rule1>
      <Rule_Number>1</Rule_Number>
    </Software_DMZ>
...
</router-configuration>
```

<Bonjour_Enable>

Parameter	Details
<Bonjour_Enable>	**Description:** Enables or disables the Bonjour service discovery protocol, which may be required by network management systems that you use **User Interface:** Administ*ration* > *Management* > *Bonjour* page, *Enabled* and *Disabled* fields **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example: Bonjour enabled** <Bonjour_Enable>1</Bonjour_Enable>

<Reset_Button_Enable>

NOTE No other settings are nested below <Reset_Button_Enable>.

Parameter	Details
<Reset_Button_Enable>	**Description:** Enables or disables the RESET button **User Interface:** **Values**: 0: Disabled (button) 1: Enabled (button can be pressed for 1-2 seconds for reboot and 5-6 seconds for a factory reset) **Default**: 1 **Example: Button disabled** `<Reset_Button_Enable>0</<Reset_Button_Enable>`

<Router_Mode>

Parameter	Details
<Router_Mode>	**Description:** The operating mode of the router **User Interface:** *Network Setup > Basic Setup > Network Service* page, *Networking Service* field **Values**: 0: Bridge 1: NAT **Default**: 1 **Example: Bridge mode enabled** `<Router_Mode>0<Router_Mode>`

<VPN_Passthrough>

This section describes the parameters in the <VPN_Passthrough> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<VPN_Passthrough> Example: All passthrough options enabled, page 186**.

Parameter	Details
<IPSec_Passthrough>	**Description:** Enables or disables VPN passthrough for Internet Protocol Security (IPsec) **User Interface:** *Network Setup > Advanced Settings > VPN Passthrough* page, *IPsec Passthrough* field **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example:** `<IPSec_Passthrough>1</IPSec_Passthrough>`
<PPTP_Passthrough>	**Description:** Enables or disables VPN passthrough for Point-to-Point Tunneling Protocol (PPTP) **User Interface:** *Network Setup > Advanced Settings > VPN Passthrough* page, *PPTP Passthrough* field **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example:** `<PPTP_Passthrough>1</PPTP_Passthrough>`

Parameter	Details
<L2TP_Passthrough>	**Description:** Enables or disables VPN passthrough for Layer 2 Tunneling Protocol (L2TP) **User Interface:** *Network Setup > Advanced Settings > VPN Passthrough* page, *L2TP Passthrough* field **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example:** `<L2TP_Passthrough>1</L2TP_Passthrough>`

<VPN_Passthrough> Example: All passthrough options enabled

```
<router-configuration>
...
    <VPN_Passthrough>
      <IPSec_Passthrough>1</IPSec_Passthrough>
      <PPTP_Passthrough>1</PPTP_Passthrough>
      <L2TP_Passthrough>1</L2TP_Passthrough>
    </VPN_Passthrough>
...
</router-configuration>
```

<Web_Management>

This section describes the parameters in the <Web_Management> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<Web_Management> Example: Remote Management and Remote Upgrade enabled, page 190**.

Parameter	Details
<Web_Utility_Access_HTTP>	**Description:** Enables or disables access to the web-based configuration utility via HTTP, from a computer on the LAN **User Interface:** *Administration > Management > Web Access Management* page, *Web Utility Access* field, *HTTP* option **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example:** `<Web_Utility_Access_HTTP>1</Web_Utility_Access_HTTP>`
<Web_Utility_Access_HTTPS>	**Description:** Enables or disables access to the web-based configuration utility via HTTPS, from a computer on the LAN **User Interface:** *Administration > Management > Web Access Management* page, *Web Utility Access* field, *HTTPS* option **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example:** `<Web_Utility_Access_HTTPS>1</Web_Utility_Access_HTTPS>`

Parameter	Details
<Web_Remote_Management>	**Description:** Enables or disables access to the web-based configuration utility through the WAN interface (INTERNET port) **User Interface:** *Administration > Management > Web Access Management* page, *Remote Management* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example:** `<Web_Remote_Management>1</Web_Remote_Management>`
<Remote_Web_Utility_Access>	**Description:** Specifies the protocol that can be used to access the web-based configuration utility through the WAN interface (INTERNET port), when Remote Management is enabled **User Interface:** *Administration > Management > Web Access Management* page, *Web Utility Access* field **Values**: 0: HTTP 1: HTTPS **Default**: 0 **Example:** `<Remote_Web_Utility_Access>1</Remote_Web_Utility_Access>`

Parameter	Details
<Web_Remote_Upgrade>	**Description:** Enables or disables upgrading the firmware from a computer on the WAN, when Remote Management is enabled **User Interface:** *Administration > Management > Web Access Management* page, *Remote Upgrade* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example:** `<Web_Remote_Upgrade>1</Web_Remote_Upgrade>`
<Allowed_Remote_IP_Type>	**Description:** Specifies a method for identifying remote devices that are allowed access to the web-based configuration utility, when Remote Management is enabled **User Interface:** *Administration > Management > Web Access Management* page, *Allowed Remote IPv4 Address* field, *Any IP Address* option **Values**: 0: Specified IP Address 1: Any IP Address **Default**: 1 **Example:** `<Allowed_Remote_IP_Type>0</Allowed_Remote_IP_Type>`

Parameter	Details
<Allowed_Remote_IP_Address>	**Description:** Specifies a remote IPv4 address that is allowed access to the web-based configuration utility, when Remote Management is enabled **User Interface:** *Administration > Management > Web Access Management* page, *Allowed Remote IPv4 Address* field, unlabeled text box **Values:** IPv4 address **Default**: 0.0.0.0 **Example:** `<Allowed_Remote_IP_Address>209.165.201.129 129</Allowed_Remote_IP_Address>`
<Remote_Management_Port>	**Description:** Specifies the port to use for access to the web-based configuration utility through the WAN interface (INTERNET port) **User Interface:** *Administration > Management > Web Access Management* page, *Remote Management Port* field **Values**: port number **Default**: 80 **Example:** `<Remote_Management_Port>443</Remote_Management_Port>`

<Web_Management> Example: Remote Management and Remote Upgrade enabled

```
<router-configuration>
...
   <Web_Management>
     <Web_Utility_Access_HTTP>1</Web_Utility_Access_HTTP>
     <Web_Utility_Access_HTTPS>1</Web_Utility_Access_HTTPS>
     <Web_Remote_Management>1</Web_Remote_Management>
     <Remote_Web_Utility_Access>1</Remote_Web_Utility_Access>
     <Web_Remote_Upgrade>1</Web_Remote_Upgrade>
     <Allowed_Remote_IP_Type>0</Allowed_Remote_IP_Type>
     <Allowed_Remote_IP_Address>209.165.201.129 129</Allowed_Remote_IP_Address>
     <Remote_Management_Port>443</Remote_Management_Port>
   </Web_Management>
...
```

```
</router-configuration>
```

<TR_069> Parameters

This section describes the parameters in the <TR_069> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<TR-069> Example: TR-069 Enabled, page 194**.

Parameter	Details
<TR_069_Status>	**Description:** Enables or disables remote provisioning via TR-069 CPE WAN Management Protocol **User Interface:** *Administration > Management > TR-069* page, *Status* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example:** `<TR_069_Status>1</TR_069_Status>`
<TR_069_ACS_URL>	**Description:** The URL of the Auto-Configuration Server (ACS) **User Interface:** *Administration > Management > TR-069* page, *ACS URL* field **Values**: Domain name or IP address, starting with http:// or https://, and optionally ending with a port number **Default**: null **Example:** `<TR_069_ACS_URL>http://ACS-example.com</TR_069_ACS_URL>`

Parameter	Details
<TR_069_ACS_Username>	**Description:** The username for HTTP-based authentication to the ACS **User Interface:** *Administration > Management > TR-069* page, *ACS Username* field **Values**: username **Default**: null **Example:** `<TR_069_ACS_Username>MyUsername</TR_069_ACS_Username>`
<TR_069_ACS_Password>	**Description:** The password for HTTP-based authentication to the ACS **User Interface:** *Administration > Management > TR-069* page, *ACS Password* field **Values**: password **Default**: commented out: `<!-- <TR_069_ACS_Password></TR_069_ACS_Password> -->` **Example:** `<TR_069_ACS_Password>MyACSPassword</TR_069_ACS_Password>`
<TR_069_Connection_Request_URL>	**Description:** This field will be auto-filled and does not need to be entered manually **User Interface:** *Administration > Management > TR-069* page, *Connection Request URL* field **Values**: URL **Default**: null **Example:** not applicable, value is auto-filled

Parameter	Details
<TR_069_Connection_Request_Username>	**Description:** This field will be auto-filled and does not need to be entered manually **User Interface:** *Administration > Management > TR-069* page, *Connection Request Username* field **Values**: username **Default**: null **Example:** not applicable, value is auto-filled
<TR_069_Connection_Request_Password>	**Description:** This field will be auto-filled and does not need to be entered manually **User Interface:** *Administration > Management > TR-069* page, *Connection Request Password* field **Values**: password **Default**: commented out, `<!--<TR_069_Connection_Request_Password></TR_069_Connection_Request_Password>-->` **Example:** `<TR_069_Connection_Request_Password>MyPassword</TR_069_Connection_Request_Password>`
<TR_069_Periodic_Inform_Interval>	**Description:** When Periodic Information is enabled, the duration, in seconds, between CPE attempts to connect to the ACS **User Interface:** *Administration > Management > TR-069* page, *Periodic Inform Interval* field **Values**: number **Default**: 86400 **Example: Interval of 36000 seconds (10 minutes)** `<TR_069_Periodic_Inform_Interval>36000</TR_069_Periodic_Inform_Interval>`

Parameter	Details
<TR_069_Periodic_Inform_ Enable>	**Description:** Enables or disables CPE connection requests to the ACS **User Interface:** *Administration > Management > TR-069* page, *Periodic Inform Enable* field **Values**: 0: Disabled 1: Enabled **Default**: 1 **Example: Periodic Inform enabled** `<TR_069_Periodic_Inform_Enable>1</TR_069_Periodic_Inform_Enable>`

<TR-069> Example: TR-069 Enabled

```
<router-configuration>
...
    <TR_069>
      <TR_069_Status>1</TR_069_Status>
      <TR_069_ACS_URL>http://ACS-example.com</TR_069_ACS_URL>
      <TR_069_ACS_Username>MyUsername</TR_069_ACS_Username>
      <!--  <TR_069_ACS_Password></TR_069_ACS_Password> -->
      <TR_069_Connection_Request_URL />
      <TR_069_Connection_Request_Username>MyCPEUsername</TR_069_Connection_
Request_Username>
      <!--  <TR_069_Connection_Request_Password></TR_069_Connection_Request_
Password>-->
      <TR_069_Periodic_Inform_Interval>86400</TR_069_Periodic_Inform_
Interval>
      <TR_069_Periodic_Inform_Enable>1</TR_069_Periodic_Inform_Enable>
      <TR_069_Loopback_Binding>0</TR_069_Loopback_Binding>
     </TR_069>
...
</router-configuration>
```

<Log_Configuration> Parameters

This section describes the parameters in the <Log_Configuration> section of the config.xml file.

NOTE In addition to the descriptions, also refer to **<Log_Configuration> Example: Logging Critical Kernel Events and Info System Events to Viewer, Email, and a Syslog Server, page 202**.

Parameter	Details
<Status>	**Description:** Enables or disables logging **User Interface:** *Administration* > *Log* > *Log Module* page, *Status* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: Logging enabled** `<Status>1</Status>`
<RAM_Log_Size>	**Description:** The maximum size of the log file in kilobytes **User Interface:** *Administration* > *Log* > *Log Setting* page, *Log Size* field **Values**: number from 128~1024 **Default**: 200 **Example:** `<RAM_Log_Size>200</RAM_Log_Size>`

Parameter	Details
<Syslog_Server_IP>	**Description:** The IP address of the syslog server where the messages will be sent **User Interface:** *Administration* > *Log* > *Log Setting* page, *Syslog Server* section, *IP Address* field **Values**: IPv4 address **Default**: null **Example:** `<Syslog_Server_IP>209.165.202.129</Syslog_Server_IP>`
<Syslog_Server_Port>	**Description:** The port to use on the specified syslog server **User Interface:** *Administration* > *Log* > *Log Setting* page, *Syslog Server* section, *Port* field **Values**: number from 1~65535 **Default**: 514 **Example:** `<Syslog_Server_Port>514</Syslog_Server_Port>`
<MAIL_Sender>	**Description:** A valid email address from which to send the logs **User Interface:** *Administration* > *Log* > *Log Setting* page, *E-Mail* section, *Sender* field **Values**: email address, including domain, such as user1@example.com **Default**: null **Example:** `<MAIL_Sender>user1@example.com</MAIL_Sender>`

Parameter	Details
<MAIL_Receiver>	**Description:** A valid email address where the logs will be sent **User Interface:** *Administration > Log > Log Setting* page, *E-Mail* section, *Receiver* field **Values**: email address, including domain, such as user1@example.com **Default**: null **Example:** `<MAIL_Receiver>user2@example.com</MAIL_Receiver>`
<MAIL_Subject>	**Description:** Text to identify the subject of the email that will be sent **User Interface:** *Administration > Log > Log Setting* page, *E-Mail* section, *Subject* field **Values**: text **Default**: null **Example:** `<MAIL_Subject>My ATA Logs</MAIL_Subject>`
<MAIL_Smtp_Server>	**Description:** The IP address or domain name of the mail server that will be used to send the logs **User Interface:** *Administration > Log > Log Setting* page, *E-Mail* section, *SMTP Server* field **Values**: IPv4 address or domain name **Default**: null **Example:** `<MAIL_Smtp_Server>smtp.example.com</MAIL_Smtp_Server>`

Parameter	Details
<MAIL_Smtp_Port>	**Description:** The port to use on the SMTP server, as specified by the email server administrator or service provider **User Interface:** *Administration > Log > Log Setting* page, *E-Mail* section, *SMTP Port* field **Values**: number **Default**: 25 **Example:** `<MAIL_Smtp_Port>25</MAIL_Smtp_Port>`
<MAIL_Log_Count>	**Description:** The maximum number of logs to include in each email **User Interface:** *Administration > Log > Log Setting* page, *E-Mail* section, *Number of Logs* field **Values**: number, 10~200 **Default**: 100 **Example:** `<MAIL_Log_Count>100</MAIL_Log_Count>`
<MAIL_Interval>	**Description:** The interval, in minutes, at which to send emails **User Interface:** *Administration > Log > Log Setting* page, *E-Mail* section, *Interval* field **Values**: number, 1~1440 **Default**: 60 **Example:** `<MAIL_Interval>60</MAIL_Interval>`

Parameter	Details
<MAIL_Smtp_User>	**Description:** A valid email address from which to send the logs **User Interface:** *Administration > Log > Log Setting* page, *E-Mail* section, *Sender* field **Values**: email address, including domain, such as user1@example.com **Default**: null **Example:** `<MAIL_Smtp_User>MyUserName</MAIL_Smtp_User>`
<MAIL_Smtp_Passwd>	**Description:** A valid email address from which to send the logs **User Interface:** *Administration > Log > Log Setting* page, *E-Mail* section, *Password* field **Values**: email address, including domain, such as user1@example.com **Default**: null **Example:** `<MAIL_Smtp_Passwd>MyPassword</MAIL_Smtp_Passwd>`

Parameter		Description
<KERNEL> <System>		Under <KERNEL>, set the parameters for kernel logging. Under <System>, set the parameters for system logging.
	<Enable>	**Description:** Enables or disables logging for the specified service (kernel or system) **User Interface:** *Administration* > *Log* > *Log Module* page, *Kernel* and *System* check boxes **Values**: 0: Disabled 1: Enabled **Default:** 0 **Example: Logging enabled for this service (kernel or system)** `<Enable>1</Enable>`
	<Priority>	**Description:** The types of events to include in the logs, from Emergency (0) to Debugging (7) **User Interface:** *Administration* > *Log* > *Log Module* page, *Priority* field **Values**: 0: Emergency 1: Alert 2: Critical 3: Error 4: Warning 5: Notice 6: Info 7: Debug **Default**: 3 **Example: Include Critical, Alert, and Emergency events** `<Priority>2</Priority>`

Parameter		Description
	<RAM>	**Description:** Enables or disables capturing the specified logs in the Log Viewer **User Interface:** *Administration* > *Log* > *Log Module* page, *Local* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: Logs will be available in the Log Viewer** `<RAM>1</RAM>`
	<Syslog>	**Description:** Enables or disables including the specified logs in the file that is sent to the Syslog Server **User Interface:** *Administration* > *Log* > *Log Module* page, *Syslog Server* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: Logs will be sent to a Syslog Server** `<Syslog>1</Syslog>`

Parameter		Description
	<MAIL>	**Description:** Enables or disables including the specified logs in the file that is sent via email **User Interface:** *Administration* > *Log* > *Log Module* page, *E-Mail* field **Values**: 0: Disabled 1: Enabled **Default**: 0 **Example: Logs will be sent via email** `<MAIL>1</MAIL>`

<Log_Configuration> Example: Logging Critical Kernel Events and Info System Events to Viewer, Email, and a Syslog Server

```
<router-configuration>
...
    <Log_Configuration>
      <Log_Configuration>
      <Status>1</Status>
      <RAM_Log_Size>200</RAM_Log_Size>
      <Syslog_Server_IP>209.165.202.129</Syslog_Server_IP>
      <Syslog_Server_Port>514</Syslog_Server_Port>
      <MAIL_Sender>user1@example.com</MAIL_Sender>
      <MAIL_Receiver>user2@example.com</MAIL_Receiver>
      <MAIL_Subject>My ATA Logs</MAIL_Subject>
      <MAIL_Smtp_Server>smtp.example.com</MAIL_Smtp_Server>
      <MAIL_Smtp_Port>25</MAIL_Smtp_Port>
      <MAIL_Log_Count>100</MAIL_Log_Count>
      <MAIL_Interval>60</MAIL_Interval>
      <MAIL_Smtp_User>user1</MAIL_Smtp_User>
      <!-- <MAIL_Smtp_Passwd></MAIL_Smtp_Passwd> -->
      <KERNEL>
        <Enable>1</Enable>
        <Priority>2</Priority>
        <RAM>1</RAM>
        <Syslog>1</Syslog>
        <MAIL>1</MAIL>
      </KERNEL>
      <System>
        <Enable>1</Enable>
```

```
            <Priority>6</Priority>
            <RAM>1</RAM>
            <Syslog>1</Syslog>
            <MAIL>1</MAIL>
          </System>
        </Log_Configuration>
...
<router-configuration>
```

<Web_Login_Admin_Name>

Parameter	Details
<Web_Login_Admin_Name>	**Description:** The username for the administrator login, which has full read-write access to all parameters **User Interface:** *Administration* > *Management* > *User List* page, *Username* field **Values**: username **Default**: admin

<Web_Login_Admin_Password>

Parameter	Details
<Web_Login_Admin_Password>	**Description:** The password for the administrator login **User Interface:** *Administration* > *Management* > *User List* page **Values**: password **Default:** commented out `<!--<Web_Login_Admin_Password></Web_Login_Admin_Password>-->` **Example:** `<Web_Login_Admin_Password>MyPassword</Web_Login_Admin_Password>`

<Web_Login_Guest_Name>

Parameter	Details
<Web_Login_Guest_Name>	**Description:** The username for the guest login, which has limited access to view or change parameters **User Interface:** *Administration > Management > User List* page **Values**: username **Default**: cisco **Example:** `<Web_Login_Guest_Name>MyUsername</Web_Login_Guest_Name>`

<Web_Login_Guest_Password>

Parameter	Details
<Web_Login_Guest_Password>	**Description:** **User Interface:** *Administration > Management > User List* page, **Values**: password **Default:** commented out, `<!--<Web_Login_Guest_Password></Web_Login_Guest_Password>-->` **Example:** `<Web_Login_Guest_Password>MyPassword</Web_Login_Guest_Password>`

Additional Information in the <router-configuration> section

The following parameters display information about the ATA:

- **<Firmware_Version>:** The firmware version number
- **<System_Model_Number>:** The model number
- **<About_Product>**
  - **<Firmware_Version>:** The firmware version number
  - **<Model>:** The model number and description
  - **<Product_ID>**: The model number
  - **<Version_ID>:** The hardware version number
  - **<Serial_Number>:** The serial number

A

Acronyms

A/D	Analog To Digital Converter
ANC	Anonymous Call
B2BUA	Back to Back User Agent
Bool	Boolean Values. Specified as yes and no, or 1 and 0 in the profile
CA	Certificate Authority
CAS	CPE Alert Signal
CDR	Call Detail Record
CID	Caller ID
CIDCW	Call Waiting Caller ID
CNG	Comfort Noise Generation
CPC	Calling Party Control
CPE	Customer Premises Equipment
CWCID	Call Waiting Caller ID
CWT	Call Waiting Tone
D/A	Digital to Analog Converter
dB	decibel
dBm	dB with respect to 1 milliwatt
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System

DRAM	Dynamic Random Access Memory
DSL	Digital Subscriber Loop
DSP	Digital Signal Processor
DTAS	Data Terminal Alert Signal (same as CAS)
DTMF	Dual Tone Multiple Frequency
FQDN	Fully Qualified Domain Name
FSK	Frequency Shift Keying
FXS	Foreign eXchange Station
GW	Gateway
ITU	International Telecommunication Union
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	HTTP over SSL
ICMP	Internet Control Message Protocol
IGMP	Internet Group Management Protocol
ILEC	Incumbent Local Exchange Carrier
IP	Internet Protocol
ISP	Internet Service Provider
ITSP	Internet Telephony Service Provider
IVR	Interactive Voice Response
LAN	Local Area Network
LBR	Low Bit Rate
LBRC	Low Bit Rate Codec
MC	Mini-Certificate
MGCP	Media Gateway Control Protocol

MOH	Music On Hold
MOS	Mean Opinion Score (1-5, the higher the better)
ms	Millisecond
MSA	Music Source Adaptor
MWI	Message Waiting Indication
OSI	Open Switching Interval
PCB	Printed Circuit Board
PR	Polarity Reversal
PS	Provisioning Server
PSQM	Perceptual Speech Quality Measurement (1-5, the lower the better)
PSTN	Public Switched Telephone Network
NAT	Network Address Translation
OOB	Out-of-band
REQT	(SIP) Request Message
RESP	(SIP) Response Message
RSC	(SIP) Response Status Code, such as 404, 302, 600
RTP	Real Time Protocol
RTT	Round Trip Time
SAS	Streaming Audio Server
SDP	Session Description Protocol
SDRAM	Synchronous DRAM
sec	seconds
SIP	Session Initiation Protocol
SLA	Shared line appearance
SLIC	Subscriber Line Interface Circuit

SP	Service Provider
SSL	Secure Socket Layer
TFTP	Trivial File Transfer Protocol
TCP	Transmission Control Protocol
UA	User Agent
uC	Micro-controller
UDP	User Datagram Protocol
URL	Uniform Resource Locator
VM	Voicemail
VMWI	Visual Message Waiting Indication/Indicator
VQ	Voice Quality
WAN	Wide Area Network
XML	Extensible Markup Language

B

Time Zone Settings

Time Zone	Setting	Example
(GMT) England	+00 2 2	<Time_Zone>**+00 2 2**</Time_Zone>
(GMT) Gambia, Liberia, Morocco	+00 1 0	<Time_Zone>**+00 1 0**</Time_Zone>
(GMT+01:00) France, Germany, Italy	+01 2 2	<Time_Zone>**+01 2 2**</Time_Zone>
(GMT+01:00) Tunisia	+01 1 6	<Time_Zone>**+01 1 6**</Time_Zone>
(GMT+02:00) Greece, Ukraine, Romania, Turkey	+02 2 2	<Time_Zone>**+02 2 2**</Time_Zone>
(GMT+02:00) South Africa	+02 1 0	<Time_Zone>**+02 1 0**</Time_Zone>
(GMT+03:00) Iraq, Kuwait	+03 1 8	<Time_Zone>**+03 1 8**</Time_Zone>
(GMT+03:00) Jordan	+03 2 9	<Time_Zone>**+03 2 9**</Time_Zone>
(GMT+04:00) ABU Dhabi, Muscat, Armenia	+04 1 0	<Time_Zone>**+04 1 0**</Time_Zone>
(GMT+05:00) Pakistan, Russia	+05 1 7	<Time_Zone>**+05 1 7**</Time_Zone>
(GMT+05:30) Bombay, Calcutta, Madras, New Delhi	+05.5 1 0	<Time_Zone>**+05.5 1 0**</Time_Zone>
(GMT+06:00) Bangladesh, Russia	+06 1 7	<Time_Zone>**+06 1 7**</Time_Zone>
(GMT+07:00) Thailand, Russia	+07 1 7	<Time_Zone>**+07 1 7**</Time_Zone>
(GMT+08:00) Australia Western	+08 1 4	<Time_Zone>**+08 1 4**</Time_Zone>
(GMT+08:00) China, Hong Kong	+08 3 0	<Time_Zone>**+08 3 0**</Time_Zone>
(GMT+08:00) Russia	+08 2 7	<Time_Zone>**+08 2 7**</Time_Zone>
(GMT+08:00) Singapore, Taiwan	+08 4 0	<Time_Zone>**+08 4 0**</Time_Zone>
(GMT+09:00) Japan, Korea	+09 1 0	<Time_Zone>**+09 1 0**</Time_Zone>
(GMT+09:30) South Australia	+09.5 1 10	<Time_Zone>**+09.5 1 10**</Time_Zone>
(GMT+10:00) Australia	+10 2 4	<Time_Zone>**+10 2 4**</Time_Zone>

Time Zone	Setting	Example
(GMT+10:00) Guam, Russia	+10 1 7	<Time_Zone>**+10 1 7**</Time_Zone>
(GMT+11:00) Soloman Islands	+11 1 0	<Time_Zone>**+11 1 0**</Time_Zone>
(GMT+12:00) Fiji	+12 1 0	<Time_Zone>**+12 1 0**</Time_Zone>
(GMT+12:00) Kwajalein	+12 3 0	<Time_Zone>**+12 3 0**</Time_Zone>
(GMT+12:00) New Zealand	+12 2 4	<Time_Zone>**+12 2 4**</Time_Zone>
(GMT-01:00) Azores	-01 1 2	<Time_Zone>**-01 1 2**</Time_Zone>
(GMT-02:00) Mid-Atlantic	-02 1 0	<Time_Zone>**-02 1 0**</Time_Zone>
(GMT-03:00) Brazil East, Greenland	-03 1 1	<Time_Zone>**-03 1 1**</Time_Zone>
(GMT-03:30) Newfoundland	-03.5 1 1	<Time_Zone>**-03.5 1 1**</Time_Zone>
(GMT-04:00) Atlantic Time (Canada), Brazil West	-04 2 1	<Time_Zone>**-04 2 1**</Time_Zone>
(GMT-04:00) Bolivia, Venezuela	-04 1 0	<Time_Zone>**-04 1 0**</Time_Zone>
(GMT-04:00) Guyana	-04 3 0	<Time_Zone>**-04 3 0**</Time_Zone>
(GMT-05:00) Eastern Time (USA & Canada)	-05 2 1	<Time_Zone>**-05 2 1**</Time_Zone>
(GMT-05:00) Indiana East, Columbia, Panama	-05 1 0	<Time_Zone>**-05 1 0**</Time_Zone>
(GMT-06:00) Central Time (USA & Canada)	-06 2 1	<Time_Zone>**-06 2 1**</Time_Zone>
(GMT-06:00) Mexico	-06 1 5	<Time_Zone>**-06 1 5**</Time_Zone>
(GMT-07:00) Arizona	-07 1 0	<Time_Zone>**-07 1 0**</Time_Zone>
(GMT-07:00) Mountain Time (USA & Canada)	-07 2 1	<Time_Zone>**-07 2 1**</Time_Zone>
(GMT-08:00) Pacific Time (USA & Canada)	-08 1 1	<Time_Zone>-**08 1 1**</Time_Zone>
(GMT-09:00) Alaska	-09 1 1	<Time_Zone>**-09 1 1**</Time_Zone>
(GMT-10:00) Hawaii	-10 1 0	<Time_Zone>-**10 1 0**</Time_Zone>
(GMT-11:00) Midway Island, Samoa	-11 1 0	<Time_Zone>**-11 1 0**</Time_Zone>

C

Where to Go From Here

Cisco provides a wide range of resources to help you and your customer obtain the full benefits of the ATA.

Product Resources

Support	
Cisco Small Business Support Community	www.myciscocommunity.com/community/smallbizsupport/voiceandconferencing
Online Technical Support and Documentation (Login Required)	www.cisco.com/support
Cisco Small Business Support and Resources	www.cisco.com/go/smallbizhelp
Downloads and Documentation	
Firmware	www.cisco.com/go/software
Technical Documentation and Open Source Documentation	IP Phones: www.cisco.com/en/US/products/ps10033/tsd_products_support_series_home.html Voice Gateways/Analog Telephone Adapters: www.cisco.com/go/smallbizvoicegateways Voice System (SPA400): www.cisco.com/en/US/products/ps10030/tsd_products_support_series_home.html

Cisco Small Business	
Cisco Partner Central for Small Business (Partner Login Required)	www.cisco.com/web/partners/sell/smb
Cisco Small Business Home	www.cisco.com/smb