Provisioning of Algo SIP Endpoints
Algo Communication Products
Version 1.6 – May 8, 2018

Contents

ALGO

1. Introduction

All Algo SIP products support provisioning. This allows system administrators to manage & configure large numbers of devices without the need to log into each individual web interface, which can save time and ensure consistent setups. It can also be used by installers to pre-configure units prior to installation on the final network.

The concept is simple: configuration files can be placed on a central server, and then the Algo devices can be instructed to fetch these files. Configuration files can be easily generated by downloading a backup of the settings currently on a device.

To provision an Algo endpoint, three steps are necessary:

- Provide the Algo endpoint device with the address of the provisioning server
- Create the desired configuration files, and place these on the provisioning server
- Reboot the device

These steps are described in more detail in the sections below.

1.1 Supported Devices

The following Algo products support provisioning:

- 8028 SIP Doorphone
- 8036 SIP Multimedia Intercom (*SIP configuration only; excluding UI configuration)
- 8039 SIP Video Intercom
- 8061 SIP Relay Module
- 8128 SIP Strobe Light
- 8180 SIP Audio Alerter
- 8186 SIP Horn Speaker
- 8188 SIP Ceiling Speaker
- 8201 SIP PoE Intercom
- 8300 Controller *(available soon)*
- 8301 Paging Adapter & Scheduler
- 8373 Zone Paging Adapter

2. Provisioning Basics

The default method of provisioning an Algo SIP Endpoint is as follows:

- **TFTP Server**: Set up a TFTP server on your network.
- **Configuration File**: Create the desired '.conf' file, and place it on the server.
- **Checksum File**: Generate a '.md5' file for the configuration, and also place it on the server.
- **DHCP Option 66**: On your DHCP server, populate Option 66 with the IP address of the TFTP server.
- **Boot**: Connect the Algo SIP Endpoint to the network, or reboot if already connected.

2.1 Server Address & Protocol

An Algo Endpoint can either be configured with the address for the Provisioning Server, or can obtain this address automatically at bootup via the DHCP Option 66 field. Using DHCP Option 66 allows the device to be provisioned with no initial configuration, while the manual provisioning option provides an alternative if access to the DHCP server is not available.

The following server types are supported: TFTP, FTP, HTTP & HTTPS*

Note: HTTPS is only supported in firmware v1.5 and above on the 8036, 8039, 8186, 8188, 8201, 8301 & 8373.

2.1.1   DHCP Option 66 (and Options 160 & 150)

The DHCP Option 66 field can be used to provide the address of the provisioning server to an Algo SIP Endpoint. In firmware v1.5 and above on the 8036, 8039, 8186, 8188, 8201, 8301 & 8373, a full URL can be provided in the Option 66 field, including the protocol, server address, and path.

If using older firmware, or a device not included in the list above, then only the server can be specified in the Option 66 field, so the following restrictions apply:

> Only the IP address (or DNS name) may be specified in the Option 66 field. Do not add any extra leading or trailing characters. I.e. do not add a protocol at the front nor an extra slash or path at the end. The protocol is TFTP by default, and cannot be changed with the Option 66 field.

In firmware v1.5 and above on the 8036, 8039, 8186, 8188, 8201, 8301 & 8373, DHCP Option 160 or Option 150 are also supported to specify the address of the provisioning server. The priority order is as follows, from highest to lowers: Option 66, 160, 150. The highest priority Option that is returned in the DHCP response is the one that will be used. A configuration option is also provided to manually specify which DHCP Option number to use if you wish to override the automatic selection.

2.1.2   Manual Configuration

As an alternative to using DHCP Option 66, a Static IP address can be provided for the provisioning server by logging into the web interface of the Algo endpoint and configuring the Static option in the tab "Advanced Settings > Provisioning". Other options also available in this tab include selection of the protocol between TFTP, FTP, HTTP & HTTPS, as well as an option to provide a path on the server from which to download the configuration files. There is also an option to disable Provisioning entirely.

2.2 Configuration Files

Configuration files contain all of the settings used by the Algo SIP endpoints. <u>Complete configuration files must be provided to the device; any settings omitted from the configuration file will be set to their factory default value</u>.

Note that in firmware v1.5 and above on the 8036, 8039, 8186, 8188, 8201, 8301 & 8373, support has also been added for "Partial Provisioning" using incremental files (described in a separate section below).

For TFTP provisioning, a checksum file in MD5 format is also required in order to validate the file transfer.

2.2.1   Generating Configuration Files

<u>The simplest way to generate a configuration file is by downloading a settings backup from a device using the web interface</u>. This file is in the same format as is used for provisioning, and will contain all the settings as currently applied on the device. This file can then be modified by hand in a text editor, or by using custom programming scripts to populate values from a database.

Two types of configuration files can be used: generic, or device-specific. Device-specific files will only be retrieved by the device with the matching MAC address. If a device-specific file is not found on the provisioning server, then the endpoint will attempt to download the generic file instead.

Advanced Tip:

> The easiest way to identify the name of a parameter is to do a lookup from the web interface, to view which parameter a given option is associated with. Right-click on the desired option in the web interface, and then select "Inspect" or "Inspect Element". In the code window that appears, you may need to click on the triangle at the start of the highlighted line to expand the block if it is not already visible (depends on the browser used). Look for the 'name' tag to find the parameter name. For example, for the SIP Server field, you'll see ' name="sip.proxy" '.

2.2.2 Filenames for Configuration Files

The following filenames are examples for the 8188 SIP Ceiling Speaker. **For other products, replace the "8188" portion of the filename with the appropriate 4 digit number for your product**. Some of the firmware components apply to specific products only.

Configuration files:

Generic configuration file	**algop8188.conf**
Device-specific configuration file	algomMACADDRESS.conf *Example*: **algom0022EE010203.conf** ➔ Note:MAC address must be in all uppercase

Other optional files related to provisioning:

Tones file	algop8188-tones.zip
Firmware image	algo-8188-1.5.fw
Kernel image (related to firmware)	algo-pa-kernel-r5.fw
Base image (related to firmware)	algo-pb-base-r1.5.fw
System image (related to firmware)	algo-pb-sys-r1.5.fw
Incremental provisioning file*	algop8188-i.conf algomMACADDRESS-i.conf

* *Note: incremental provisioning is only supported in firmware v1.5 and above on the 8036, 8039, 8186, 8188, 8201, 8301 & 8373.*

➔ Note: The names of the configuration files are all **case-sensitive.**

These filenames will be combined with the other provisioning parameters to form the full request that is sent to the provisioning server. For example, the final result would be:

Download {tftp,ftp,http,https}://server/[Config_Download_Path]/algop8188.conf

2.2.3   Generic Configuration Files

The generic configuration file will be downloaded by all devices of the matching product model (if a device-specific file is not provided). Generic configuration files can be used to apply settings that are common to all devices, but of course cannot be used to apply unique SIP credentials to individual devices.

Note that if a generic file is used to first set common settings, and SIP credential are later applied manually via the web interface of each device, then it is important that this generic provisioning file is removed (or the provisioning option is disabled on the device) so that the provisioning file is not accidentally applied again later thus deleting the modified settings.

2.2.4   Device-Specific Configuration Files

Device-specific configuration files contain the MAC address in the filename, and are thus only retrieved by a single target device. These files can be used to completely configure a device. If the provisioning server is left active after initial configuration, they can also be used to ensure that the device configuration is restored to these intended settings at each reboot.

2.2.5   Partial Provisioning using Incremental Files

*** Note: Incremental provisioning files are only supported in firmware v1.5 and above on the 8036, 8039, 8186, 8188, 8201, 8301 & 8373.**

Incremental provisioning files can be either Generic or Device-Specific files, as described above, and are created by adding a "-i" suffix to either of these types of filename (before the file extension).

For example: **algop8188-i.conf**, or **algom0022EE010203-i.conf** (and the same for the .md5 if using TFTP).

In contrast to the regular provisioning method described previously, where omitted parameters are set to factory default values, any parameters omitted from an Incremental Provisioning file will retain their original settings as currently stored on the device.

This allows settings such as SIP credentials, or a new speaker volume level, to be assigned to a device without affecting the other settings already on the device.

Note that a given device will only download a single provisioning file from the server. Multiple file types (e.g. generic + incremental) cannot be combined on the same server. In the event that multiple files are found on the server, the search order is as follows: algom, algom-i, algop, algop-i. Only the first file that is found will be used. See "Detailed Provisioning Behavior" later in this document for more details.

Note: this feature is disabled by default. To use it, enabled it in the web interface at: "Advanced Settings > Provisioning > Partial Provisioning".

2.2.6   MD5 Checksum

An .md5 checksum file must also be generated and uploaded to the Provisioning server (for TFTP only). This checksum file is used to verify that the .conf file is transferred correctly without error.

On Linux systems, the following command can be used:

>   md5sum *<filename>*

Example:
>   md5sum *algop8180.conf*

On Windows systems, a third-party tool is required to generate the md5 checksum file. One such tool can be found at the website address below:

>   www.fourmilab.ch/md5

To use this specific Windows tool, simply unzip and run the application (md5) from a command prompt (no installation is required). The proper .md5 file will be generated in the same directory. When using the above tool, be sure to use the "-l" parameter to generate lower case letters:

>   md5 -l -o*<output filename> <filename>*

Example:
>   md5 -l -o*algop8180.md5 algop8180.conf*

If the md5 checksum file is not present on the server, or does not correctly match the .conf file, then this provisioning file will be ignored.

An .md5 file is always required when using provisioning to upload a tones .zip file (see section on Tones provisioning).

2.3 Reboot

Once the provisioning server is ready, each Algo SIP Endpoint must be rebooted in order to trigger a fetch of the configuration files. This can be done in a number of ways:

- Plug in a new unit to the network
- Remove and restore PoE power via the Ethernet switch/router
- Send a "CheckSync" NOTIFY command via the network (note: CheckSync is only supported when Provisioning Mode is enabled)

If a new provisioning file is successfully retrieved, the device will then reboot again automatically, and apply this new configuration. This allows multi-stage provisioning whereby the first provisioning server may contain a generic configuration file that contains settings that point to a second provisioning server where the device-specific configuration files are kept; see details in the Advanced section below.

2.4 Provisioning Parameters

The provisioning parameters can be seen in the web interface in the tab: "Advanced Settings > Provisioning"

The following parameters are available:

- **Provisioning Mode**: enables or disables all provisioning functions.

- **Server Method**: server address can be obtained either from DHCP server via Option 66 (default), or a static address can be specified.

- **Static Server**: additional support for a "full URL" has been recently added for this field.
    - *If using firmware v1.5 and above on the 8036, 8039, 8186, 8188, 8201, 8301 & 8373:*
        This field can contain the server address, along with the protocol and path. All of the following examples are valid:
        "192.168.1.1"
        "provserver.mydomain.com"
        "ftp://192.168.1.1"
        "192.168.1.1/files"
        "user:pwd@192.168.1.1"
    - *If using a device not listed above, or an older firmware version:*
        Enter server address only with no extra leading or trailing characters. I.e. do not add a protocol at the front nor extra slash or path at the end.
        Good example: "192.168.1.1", "provserver.mydomain.com"
        Bad examples: "ftp://192.168.1.1", "192.168.1.1/", "192.168.1.1/files",
        "user:pwd@192.168.1.1"
    - ➔ Note: These same requirements apply to the DHCP Option 66 field as well

- **Download Method**: Select protocol supported by server, either TFTP (default), FTP, HTTP, HTTPS.
    Note: HTTPS is only supported in firmware v1.5 and above on the 8036, 8039, 8186, 8188, 8201, 8301 & 8373.

- **Username / Password**: Login credentials to access the server (not applicable in TFTP mode).

- **Config Download Path**: Path on server to request config files from. Do not add any extra leading or trailing characters or slashes.

- **Firmware Download Path**: Path on server to request firmware files from. Do not add any extra leading or trailing characters or slashes.

3. Firmware Upgrade via Provisioning

Configuration files can also contain an optional parameter that specifies a desired firmware version. If this version number does not match the current firmware installed on the device, then the new firmware will be downloaded from the provisioning server and installed. The available parameters are shown below with example values. The first one, "prov.version.firmware" applies to all products, whereas the others apply only to specific devices.

Firmware parameter, with example value:

prov.version.firmware = 2.6

Other firmware components, applicable to specific products only (with examples values):

prov.version.kernel = r5
prov.version.base = r1.2
prov.version.sys = r1.2

The full filename will be generated automatically based on the parameter name, version number, and product that it is applied to. In the example above, "prov.version.firmware = 2.4" when applied on the 8180 will request the following file: "algo-8180-2.4.fw". Both this file, and the matching md5 checksum, must be present on the provisioning server for the upgrade to be successful. The correct .md5 file is always be provided by Algo for all firmware releases. These firmware files should always be used as-is; you will never need to change the filenames nor contents of the .fw & .md5 files that were provided by Algo when using provisioning. Only the version number portion needs to be specified in the provisioning parameter, the rest of the filename will be matched automatic.

Note: When upgrading the firmware via provisioning, the device will reboot two additional times as part of the process. First, the device will install the new configuration file and reboot to apply it. Next, it will parse the new parameter requesting the firmware version, and thus download and install the firmware. It will then reboot again at the end of the firmware upgrade process. Ensure that the provisioning feature is still enabled in the config file that specifies the target firmware version, otherwise the firmware changes will not be applied.

Note that the "bundle" firmware package (which contains the firmware, base & sys together in a single package) is not supported by the provisioning function at this time.

4. Tone Files Installation via Provisioning

Audio tone files can be installed on a device using provisioning by placing one or more .wav files within a .zip file, and then using the methods described below.

4.1 Tones Provisioning on 8180

On the 8180, the device will automatically attempt to download a tones file from the provisioning server, using the following name:

algop8180-tones.zip
(and matching .md5 checksum)

4.2 Tones Provisioning on all other products

On the 8188, 8186, and 8300-series products, the device will use the parameter "prov.data.tone" in the configuration file to specify the tones filename. To enable tone file provisioning on these products, add this parameter to the configuration file and provide the full file name of the desired tones package, including the .zip extension (note that the filename is case sensitive).

Example:
    prov.data.tone = TONE_FILE.zip

Also create a valid checksum (.md5) for TONE_FILE.zip, using the same process as described for the configuration files. The filename should be the same, just with a suffix of .md5

Example:
    TONE_FILE.md5

Note: As with the firmware upgrade process, the device will first download & apply the new configuration file, followed by a reboot, before attempting to download the specified tone files. Thus the provisioning feature must still be enabled in the newly applied config file.

The follow products do not support Tones provision: 8028, 8061

Note: The tone file .zip must be placed in the same folder on the provisioning server as the firmware files (i.e. the "Firmware Download Path" parameter will be used for the tones as well). Only the filename can be specified in 'prov.data.tone' - this parameter must not contain a path.

4.2.1 "New" versus "Same" Tones File

In order to identify whether the tones file is "new" or not each time the device boots and provisions, the .md5 file is saved internally on the device for future comparison. If no changes are made to the tones, then the device will not download and install the same .zip file next time.

There are three cases that will trigger the device to decide that the tones .zip file is new, and thus download & install it the next time provisioning occurs:

1) Different .md5 (and .zip) file found on provisioning server
2) Renaming or deleting any tone files from the web interface
3) Removing the "prov.data.tone" parameter from the config file (and then adding it again of course)

In the last case, if the device is provisioning successfully with no tones file specified, then the next time that it does see the tones file specified again, it will treat it as a new file and install it.

5. Advanced Provisioning

This section covers some more advanced details and techniques.

5.1 Detailed Provisioning Behavior

This section provides a detailed list of the steps that occur on the device as part of the provisioning process. Note that any time a changed provisioning file is successfully accepted, the device will apply these changes, reboot, and start from the top of the list again. On the next pass through these steps, the device should detect a match at the stage that was just completed, and thus continue further. Once all configuration and firmware on the device matches with that on the provisioning server, the device will proceed through all the steps without further action, and will begin normal operation.

At powerup/reboot, the device will go through the following steps:

1) Verify provisioning is enabled (on by default). Exit if disabled.

2) Obtain provisioning server address from DHCP Option 66 (default), or static address if set.

3) Download config files...

   TFTP case:

   a) Attempt download of MAC-specific config file
   b) Attempt download of md5 file for MAC-specific config (if config file was found)
   c) Attempt download of incremental MAC-specific config file (if previous file was not found; and if supported by device/firmware)
   d) Attempt download of md5 file for incremental MAC-specific config (if config file was found)
   e) Attempt download of generic config file (if previous file was not found)
   f) Attempt download of md5 file for generic config (if config file was found)
   g) Attempt download of incremental generic config file (if previous file was not found; and if supported by device/firmware)
   h) Attempt download of md5 file for incremental generic config (if config file was found)

   - If the downloaded config & md5 files do not match, then this config file is ignored.
   - If the config file matches the config file already on the device, then this config file is ignored (file is diff'ed with existing file)
   - If the downloaded config & md5 files do match, and the config is new, then the config file is applied and the device will immediately reboot (and thus start the process again).

FTP, HTTP & HTTPS case:
- a) Attempt download of MAC-specific config file
- b) Attempt download of incremental MAC-specific config file (if previous file was <u>not</u> found)
- c) Attempt download of generic config file (if previous file was <u>not</u> found)
- d) Attempt download of incremental generic config file (if previous file was <u>not</u> found)

- If the downloaded config file differs from that currently on the device, then the config file is applied and the device will immediately reboot (and thus start the process again)

4) Check config file for firmware version

- a) If firmware version parameter is present in config file, and this version does not match the currently installed firmware version, then attempt download of appropriate .fw & .md5 files.
- b) If firmware & md5 files match, then install firmware and reboot (and thus start the provisioning process again from the start)

5) Tones

8180:
- a) Attempt to download 'algop8180-tones.md5'
- b) Attempt to download 'algop8180-tones.zip' (if md5 was found, and different than last stored md5)
- If the downloaded .zip & md5 files match, then the tones are unzipped and saved on the device (any existing tone files with matching names are overwritten)

Other products:
- a) Attempt to download md5 file for tones file (if tones parameter present in config file)
- b) Attempt to download tones file .zip (if md5 was found, and different than last stored md5)
- If the downloaded .zip & md5 files match, then the tones are unzipped and saved on the device (any existing tone files with matching names are overwritten)

5.2 2-Stage Provisioning

Note: now that the Protocol & Path are also supported for Option 66 in firmware v1.5 and above on the 8036, 8039, 8186, 8188, 8201, 8301 & 8373, this often replaces the need for 2-Stage Provisioning.

Example: You have an existing provisioning server that supports only FTP (with password required) that you wish to use to provision a large number of Algo 8180 SIP Audio Alerters. This server has a static IP address.

Solution: Set up a temporary TFTP server, and use DHCP Option 66 to point the 8180's to this server. Note that this step could potentially be done at a different location, e.g. at a vendor's office prior to installation with the end customer. Log in to the web interface on one 8180, and set the options required to access the FTP provisioning server (i.e. provisioning server static IP address, the FTP mode, and login credentials). From the tab "System > Maintenance", use the "Backup" button to download the configuration file from this device, and save this on the TFTP server. This file then contains all the settings necessary to configure the remainder of the 8188's to use the regular FTP server, so next, create the necessary .md5 file. Once done, plug in or reboot all the other 8188's and they will obtain this configuration and then begin communicating with the desired provisioning server.

What actually happens: When a new 8180 is booted in this scenario, it will receive the address of the temporary TFTP via the DHCP Option 66 field. It will then contact this TFTP server, and download the configuration file that you created which contains the settings necessary to contact the existing FTP provisioning system. The 8180 will then apply these new settings and reboot, and now contact the FTP server this time and log in successfully. It will then pull whatever configuration files are stored on that server, which will typically be specific to each endpoint and contain the SIP account details and other desired settings. Once this first TFTP step is completed on all units, the TFTP server can be removed and the DHCP Option 66 field can be cleared.

5.3 Disabling Provisioning After Completion

It is important to consider whether you wish provisioning to be still enabled or not after completion. Often devices are provisioned once prior to installation, and then installed at a site. If a static provisioning server was configured, then the device will still attempt to contact this server, and may lead to a delay in booting since the server does not exist at this location.

If settings such as SIP accounts or volume control will be adjusted directly on each device after initial provisioning, then it is important that provisioning is disabled so that these settings are not overwritten at next bootup.

6. Troubleshooting

Troubleshooting issues with provisioning can sometimes be difficult as the device will not provide any output during this process. Be sure to observe the logs from your provisioning server to see any file requests that the device might be making, and see the list below for other hints:

- Verify that provisioning has not been disabled in the web interface.
- Verify that your provisioning server supports the protocol that is selected on the device (TFTP, FTP or HTTP).
- Compare the file request logs seen on your server with the steps outlined in the above sections to determine which step the device is getting to.
- **Verify that the md5 checksum file is correct. This is typically the most common source of failure.**
- Verify that provisioning is not still enabled when not expected. This can cause settings to get overwritten as any changes in the web interface will be lost the next time the device is rebooted as it will download the original provisioning file again.