

Auto Provisioning

- [Definition](#)
- [Step 1: Get the Setting Server URL](#)
- [Step 2: Get the setting file\(s\)](#)
 - [What URL/s to fetch](#)
 - [Syntax](#)
- [Step 3: Get the firmware file\(s\)](#)
 - [What URL/s to fetch](#)
 - [Syntax](#)
- [Examples](#)
 - [Related articles:](#)

Definition

Auto Provisioning (or Mass Deployment) is a feature implemented in the standard firmware of Snom phones to allow remote administration (configuration and maintenance) of unlimited number of distinct Snom phones. Auto Provisioning is particularly useful for out-of-the-box scenarios in larger phone installations and can be used to provide general and specific configuration parameters ("Settings") to the phones and to manage firmware updates.

Step 1: Get the Setting Server URL

At boot time, a Snom phone automatically tries to find out where to go and retrieve its settings from. You can configure your phone to retrieve its settings using one or more of the following methods:

- **From the cloud**: using the [Snom Redirection Service/SRAPS](#) (by changing parameter [setting_server](#)). After booting, the phone tries first to fetch the file defined by its parameter **setting_server**. The hardcoded initial setting server that the phone tries to get its provisioning file from is <https://secure-provisioning.snom.com> (or for older phones <http://provisioning.snom.com>), which is the Snom SRAPS server. You can use the SRAPS service to change the value of parameter [setting_server](#)., i.e. to **redirect** the phone to your own provisioning server. Unless parameter [setting_server](#) or the [provisioning_order](#) gets changed, the phone will always try the initial setting server.
- **From the local network** :
 - using [SIP PnP](#) : setting [pnp_server](#)
 - using [DHCP options](#): settings [update_server](#) and [update_filename](#)



Using [SIP PnP](#) and [DHCP](#) provisioning does not automatically change the [setting_server](#). Even after you have configured the phone to fetch the configuration file through [SIP PnP](#) or [DHCP options](#), the phone will still try the cloud(Snom Redirection Service) provisioning type first. If you wish to deactivate the Snom Redirection Service you can either change parameter **setting_server** from your provisioning file to direct to your server or simply remove value "redirection:stop" from parameter [provisioning_order](#).

- **Manual**: it is also possible to change the Setting URL([setting_server](#)) manually
- Using [TR-069 - Full Endpoint Management](#)

The parameter [provisioning_order](#) defines in which order the phone will attempt from these given provisioning types. The default order is: **Snom Redirection Service; SIP PnP; DHCP; TR69**

Step 2: Get the setting file(s)

What URL/s to fetch

Once the phone has received the Setting Server URL, it will go to that server to fetch its settings.

If the Setting Server URL does not contain a file name, the phone will try to "guess" the configuration file name(s) by creating their name from its phone type and MAC address. The first file it tries to fetch is *<phoneType>.htm*, where phoneType is the same string as the vendor identifier which you can find in [this table](#) . Apart from phoneType.htm, the phone also tries to fetch *<phoneType>-<macAddress>.htm*.

Example: if server is set to "http://192.168.137.1", file name is not provided and the phone is a D345 with MAC address 000413A11FA7, the phone will try to fetch the following files:

- <http://192.168.137.1/snomD345.htm>
- <http://192.168.137.1/snomD345-000413A11FA7.htm>

You can also use replacement variables, which will be replaced during the provisioning. They are written inside of curly brackets - syntax: *{variable}*. The following variables are supported:

- **mac**: the content of the setting mac, e.g. 000413920A74
- **phone_type**: the content of the setting phone_type, e.g. snomD735 (**available since firmware version 10.1.47**)
- **firmware_version**: the last field of the content of the setting firmware_version, e.g. 10.1.47 (**available since firmware version 10.1.47**)

If the filename in the Setting Server URL does not already contain the placeholder **{mac}**, the phone will also try to fetch a MAC-specific URL that it creates using the given Setting Server URL.

Examples:

If the provisioned Setting Server URL is: `http://example.com/test.xml` or `http://example.com`, the phone will try to fetch the following setting files:

- `http://example.com/ test .xml`
- `http://example.com/ test - 000413920A74.xml` (000413920A74 is just an example, the phone will replace this with **its own** MAC address)

If the provisioned Setting Server URL is: `http://example.com/ test {mac}.xml`, the phone will try to fetch just one setting file:

- `http://example.com/ test 000413920A74.xml` (000413920A74 is just an example, the phone will replace this with **its own** MAC address)

Syntax

Continue to read here more about the [syntax of the configuration files](#).

Step 3: Get the firmware file(s)

What URL/s to fetch

Once the phone has received the Setting Server URL, it will also try to fetch its [firmware file\(s\)](#). A firmware file is used to configure which firmware version the phone should use.



In order for the phone to automatically install the firmware update version according to the firmware file, setting [update_policy](#) must be set to "auto_update". To set this setting automatically, you can add it in the setting file from Step 2:

```
<update_policy perm="RW">auto_update</update_policy>
```

If the setting file(s) have **not** set the parameters "firmware_status" or "firmware", the phone will try by default to fetch the firmware file using two specific URLs that it creates using the given Setting Server URL and the phone type.

Examples:

If the provisioned Setting Server URL is `http://example.com/test.xml`, a **D765** phone will try to fetch the following firmware file:

- `http://example.com/snomD765/snomD765-firmware.htm`
- in case file "`http://example.com/snomD765/snomD765-firmware.htm`" was **not found**, the phone also tries to fetch: `http://example.com/snomD765-firmware.htm`

If the setting file(s) have set the "firmware_status" or "firmware" parameters, the phone will only try to fetch the URL configured by these parameters - see [Firmware Update via Autoprovisioning](#) for more details on how to use these parameters.

Syntax

Continue to read here more about the [syntax of the firmware configuration files](#).

Examples

See some examples here:

[-Provisioning example via DHCP](#)

[-Provisioning example via PnP](#)



Further Information

- [VoIP Essentials](#)
- [Visit the Snom Forum](#)
- [Open a support ticket](#)
- [Find a local partner](#)

Related articles:

- [<certificates> tag](#)
- [<dialplan> tag](#)
- [<functionKeys> tag](#)
- [<gui-languages>, <web-languages>tag](#)
- [<phone-settings> tag](#)
- [<ReplacementPlan> tag](#)
- [<Setting-Files> tag](#)
- [<tbook>, <phone-book> tag](#)
- [<uploads> tag](#)
- [Auto Provisioning](#)
- [Basic setting provisioning via DHCP](#)
- [D3xx Function Key Index Ranges](#)
- [D7xx Function Key Index Ranges](#)
- [Desk Phone Configuration Files](#)
- [DHCP options](#)
- [How to trigger the phone to reset or synchronize its settings via provisioning - mass deployment](#)