Yealink

Yealink SIP IP Phones Auto Provisioning Guide

Version 83.11
Aug. 2018

Table of Contents

Table of Contents iii

Introduction 1

Supported Phones 1

Getting Started 3

Obtaining Boot, Configuration and Resource Files 3

Boot Files 3

Configuration Files 3

Resource Files 4

Obtaining Template Files 5

Obtaining Phone Information 5

Provisioning Yealink IP Phones 7

Interoperating with Provisioning Server 7

Auto Provisioning Process 8

Old Mechanism – Without Boot Files 8

New Mechanism – With Boot Files 9

Major Tasks for Auto Provisioning 11

An Instance of Auto Provision Configuration 12

Managing Boot Files 15

Editing Common Boot File 15

Creating MAC-Oriented Boot File 17

Managing Configuration Files 19

Editing Common CFG File 19

Editing MAC-Oriented CFG File 21

Creating a New CFG File 23

Managing MAC-local CFG File 23

Encrypting Configuration Files 24

Managing Resource Files 25

Customizing Resource Files 25

Configuring a Provisioning Server 27

Preparing a Root Directory 27
Configuring a TFTP Server 28

Obtaining the Provisioning Server Address 31

Zero Touch 31
Plug and Play (PnP) Server 33
DHCP Options 34
Phone Flash 35
Configuring Wildcard of the Provisioning Server URL 36

Triggering the IP Phone to Perform Auto Provisioning 39

Power On 39
Repeatedly 40
Weekly 41
Flexible Auto Provision 42
Auto Provision Now 43
Multi-mode Mixed 44
SIP NOTIFY Message 44
Auto Provisioning via Activation Code 45

Downloading and Verifying Configurations 49

Downloading Boot, Configuration and Resource Files 49
Resolving and Updating Configurations 49
Using MAC-local CFG File 50
Verifying Configurations 50

Troubleshooting 53

Glossary 55

Appendix 57

Configuring an FTP Server 57
Preparing a Root Directory 57
Configuring an FTP Server 58
Configuring an HTTP Server 60
Preparing a Root Directory 60

Configuring an HTTP Server 61

Introduction

Yealink IP phones are full-featured telephones that can be plugged directly into an IP network and can be used easily without manual configuration.

This guide provides instructions on how to provision Yealink IP phones with the minimum settings required. Yealink IP phones support FTP, TFTP, HTTP, and HTTPS protocols for auto provisioning and are configured by default to use the TFTP protocol.

Supported Phones

The purpose of this guide is to serve as a basic guidance for provisioning Yealink IP phones.

The following table lists product names and available firmware versions for IP phones that use auto provisioning process outlined in this guide.

Product Name	Boot File (Available Firmware Version)	Exclude Mode (Available Firmware Version)
SIP-T58V/A	Yes (80 or later)	Yes (83 or later)
SIP-T56A	Yes (80 or later)	Yes (83 or later)
SIP-T54S	Yes (81 or later)	Yes (83 or later)
SIP-T52S	Yes (81 or later)	Yes (83 or later)
SIP-T48G/S	Yes (81 or later)	Yes (83 or later)
SIP-T46G/S	Yes (81 or later)	Yes (83 or later)
SIP-T42G/S	Yes (81 or later)	Yes (83 or later)
SIP-T41P/S	Yes (81 or later)	Yes (83 or later)
SIP-T40P/G	Yes (81 or later)	Yes (83 or later)
SIP-T29G	Yes (81 or later)	Yes (83 or later)
SIP-T27G	Yes (81 or later)	Yes (83 or later)
SIP-T23P/G	Yes (81 or later)	Yes (83 or later)
SIP-T21(P) E2	Yes (81 or later)	Yes (83 or later)
SIP-T19(P) E2	Yes (81 or later)	Yes (83 or later)
CP860	Yes (81 or later)	No
CP960	Yes (80 or later)	Yes (83 or later)
CP920	Yes (81 or later)	No

Product Name	Boot File (Available Firmware Version)	Exclude Mode (Available Firmware Version)
W60P	Yes (81 or later)	No
W53P	Yes (83 or later)	No
W52P W56P	Yes (81 or later)	No

We recommend that IP phones running the latest firmware should not be downgraded to an earlier firmware version. The new firmware is compatible with old configuration parameters, but not vice versa.

Getting Started

This section provides instructions on how to get ready for auto provisioning. To begin the auto provisioning, the following steps are required:

- Obtaining Boot, Configuration and Resource Files
- Obtaining Phone Information

Obtaining Boot, Configuration and Resource Files

Boot Files

The IP phone tries to download the boot file first, and then download the configuration files referenced in the boot file during auto provisioning. You can select whether to use the boot file or not according to your deployment scenario. If required, you need to obtain the template boot file named as "y000000000000.boot" before auto provisioning.

You can use a boot file to specify which configuration files to be downloaded for specific phone groups by phone model identity, and customize the download sequence of configuration files. It is efficient for you to provision IP phones in different deployment scenarios, including all IP phones, specific phone groups, or a single phone.

The configuration files referenced in the boot file are flexible: you can rearrange the configuration parameters within the Yealink-supplied template configuration files or create your own configuration files from configuration parameters you want. You can create and name as many configuration files as you want and your own configuration files can contain any combination of configuration parameters.

Configuration Files

Before provisioning, you also need to obtain template configuration files. There are two configuration files both of which are CFG-formatted. We call these two files Common CFG file and MAC-Oriented CFG file.

The configuration files contain parameters that affect the features of the phone. You can use the configuration files to deploy and maintain a mass of Yealink IP phones automatically.

You can create and name as many configuration files as you want (e.g., account.cfg, sip.cfg, features.cfg) by using the template configuration files. The custom configuration files can contain the configuration parameters of the same feature modules for all phones.

Resource Files

When configuring some particular features, you may need to upload resource files to IP phones, such as personalized AutoDST file, language package file and local contact file. Resource files are optional, but if the particular feature is being employed, these files are required.

Yealink supplies the following resource file templates:

Feature	Template File Name
DST	AutoDST.xml
Language Packs	For example, 000.GUI.English.lang 1.English_note.xml 1.English.js
Replace Rule	dialplan.xml
Dial-now	dialnow.xml
Softkey Layout (not applicable to CP960/W52P/W53P/W56P/W60P IP phones)	CallFailed.xml CallIn.xml Connecting.xml Dialing.xml (not applicable to T58V/T58A/T56A/T48G/T48S IP phones) RingBack.xml Talking.xml
Directory	favorite_setting.xml
Super Search in dialing	super_search.xml
Local Contact File	contact.xml
Remote XML Phone Book	Department.xml Menu.xml
Screen Saver (not applicable to T58V/T58A/T56A/CP960/W52P/W53P/W56P/W60P IP phones)	CustomScreenSaver.xml
Firmware	For T54S/T52S/T48G/T48S/T46G/T46S/T42G/T42S/T41P/T41S/T40P/T29G/T27G/T23P/T23G/T21(P) E2/T19(P) E2: X.81.0.XX.rom

Feature	Template File Name
	For example, 44.81.0.15.rom
	For T58V/T58A/T56A/CP960: X.80.0.XX.rom For example, 58.83.0.1.rom

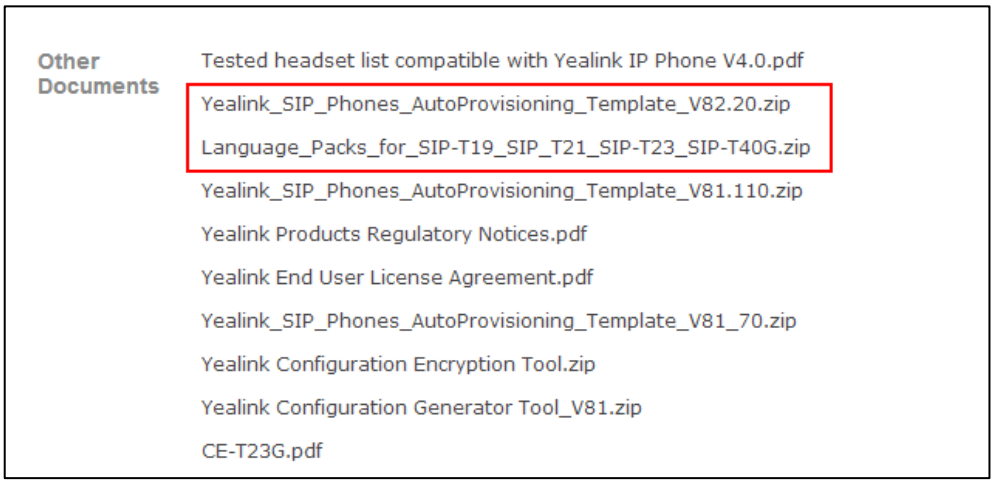
Obtaining Template Files

You can ask the distributor or Yealink FAE for template files. You can also obtain them online: http://support.yealink.com/documentFront/forwardToDocumentFrontDisplayPage.

To download template boot, configuration and resource files:

1. Go to Yealink Document Download page and select the desired phone model.
2. Download and extract the combined template files to your local system.

   For example, the following illustration shows the template files available for SIP-T23G IP phones running firmware version 82.



3. Open the folder you extracted and identify the files you want to edit.

Obtaining Phone Information

Before provisioning, you also need the IP phone information. For example, MAC address and the SIP account information of the IP phone.

MAC Address: The unique 12-digit serial number of the IP phone. You can obtain it from the bar code on the back of the IP phone.

SIP Account Information: This may include SIP credentials such as user name, password and IP

address of the SIP server. Ask your system administrator for SIP account information.

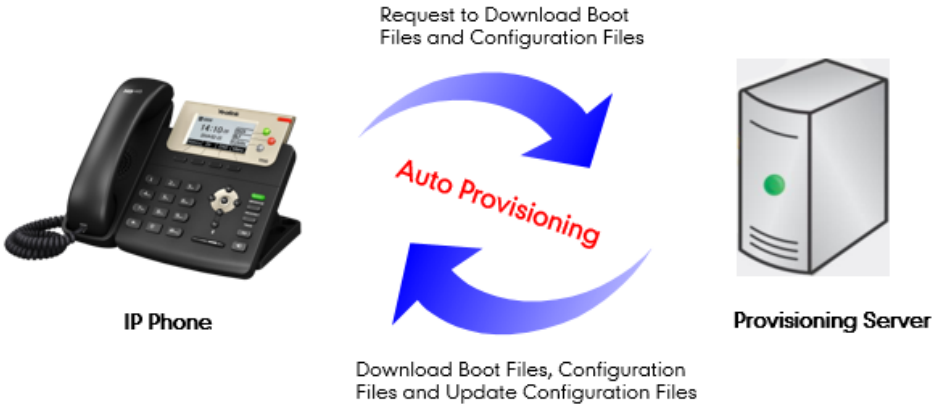
Provisioning Yealink IP Phones

This section provides instructions on how IP phones interoperate with provisioning server for auto provisioning, and shows you the auto provisioning process and the four major tasks to provision the phones. It will help users who are not familiar with auto provisioning to understand this process more easily and quickly.

Interoperating with Provisioning Server

When IP phones are triggered to perform auto provisioning, they will request to download the boot files and configuration files from the provisioning server. During the auto provisioning, the IP phone will download and update configuration files to the phone flash.

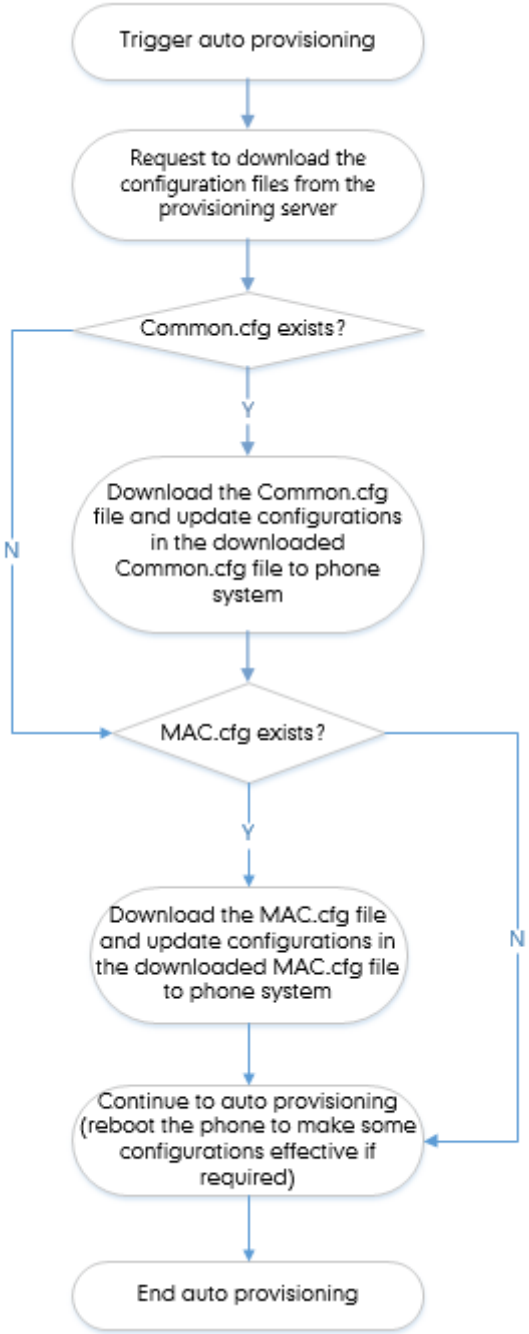
The following figure shows how the IP phone interoperates with the provisioning server:



Auto Provisioning Process

Old Mechanism – Without Boot Files

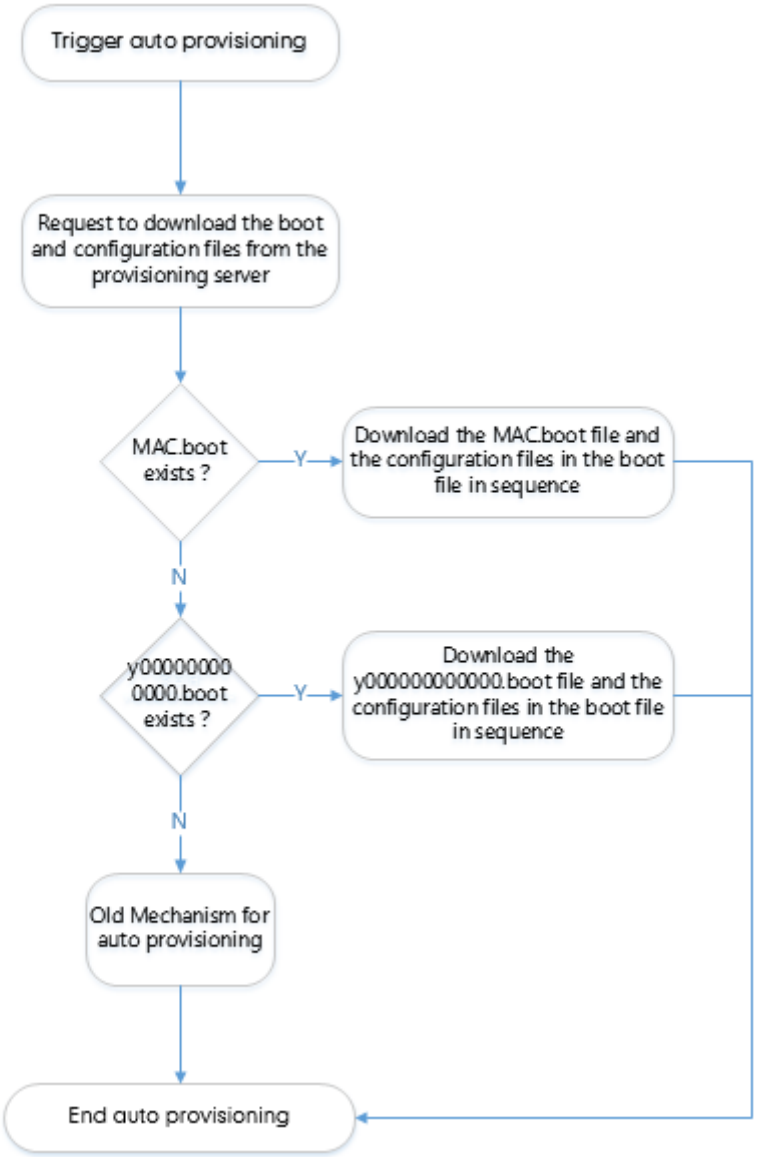
The following flowchart shows how Yealink IP phones perform auto provisioning when using configuration files only:



New Mechanism – With Boot Files

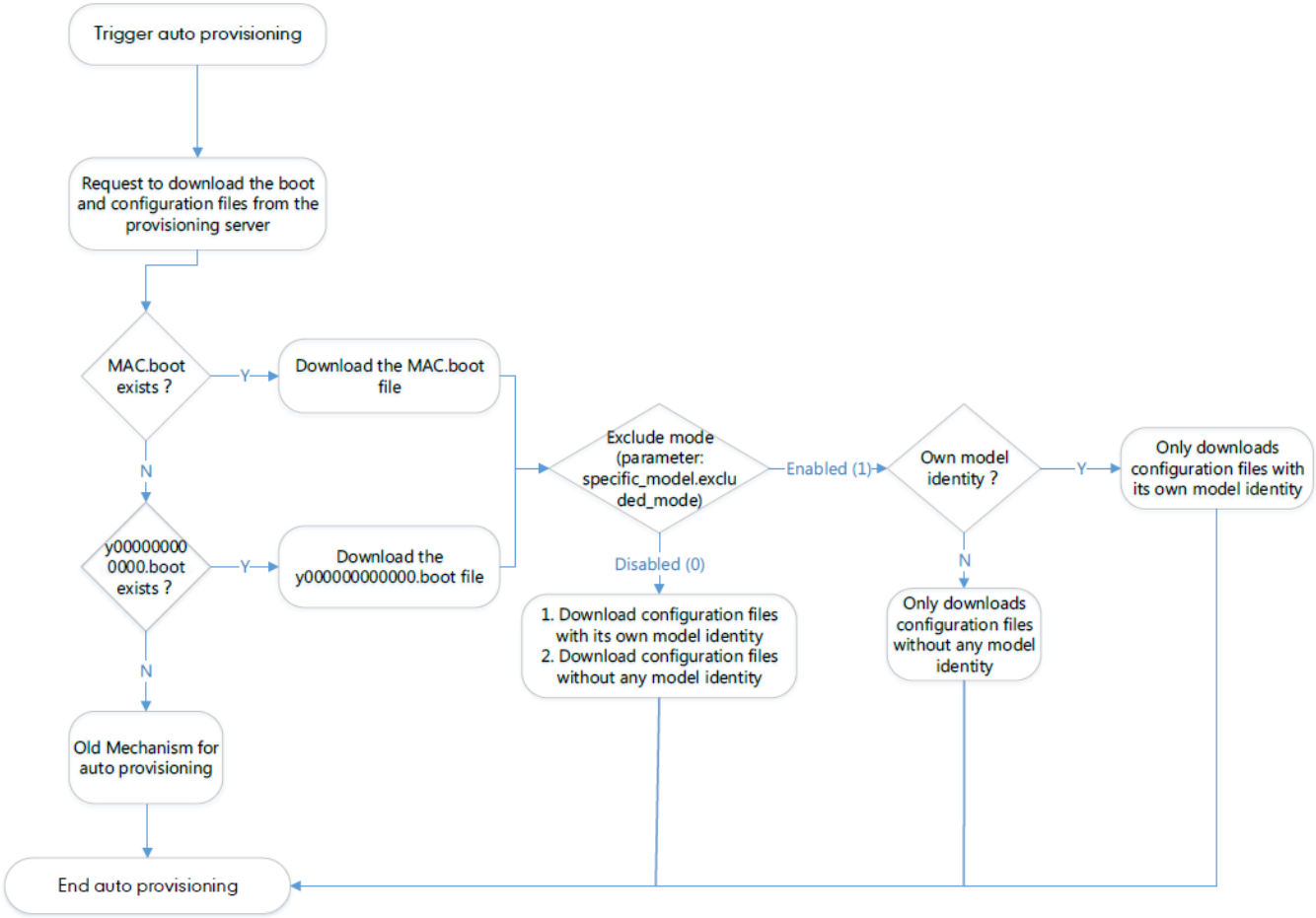
The following figure shows auto provisioning flowcharts for Yealink IP phones when using boot files:

Scenario A – Do Not Support Exclude Mode



Scenario B – Support Exclude Mode

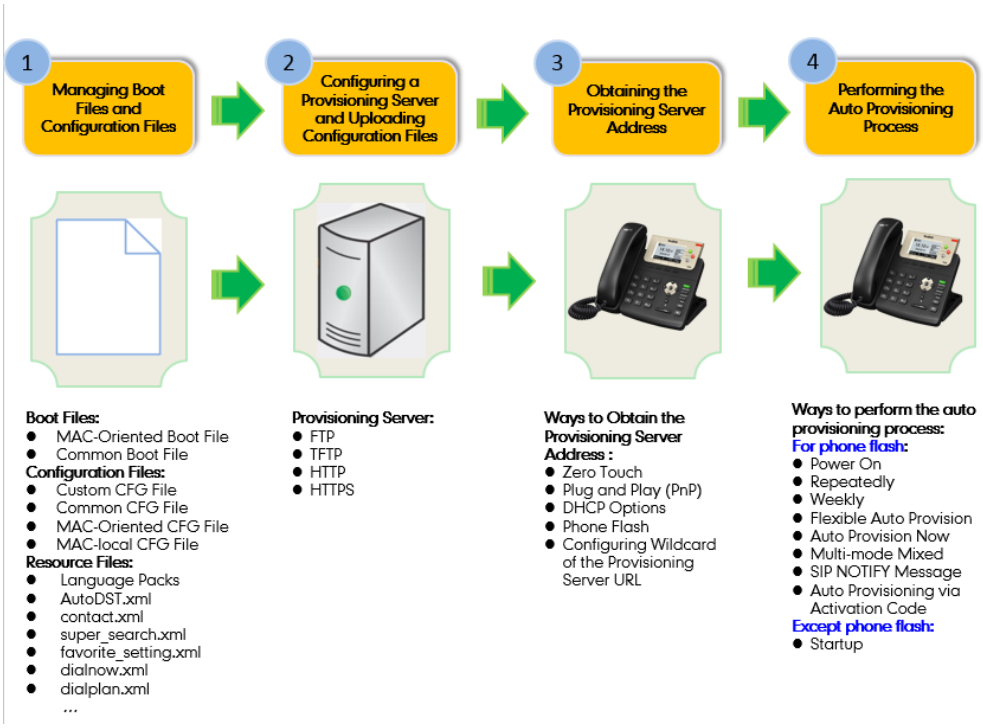
This scenario is only applicable to IP phones (except W53P/W56P IP phones) running firmware version 83 or later.



Major Tasks for Auto Provisioning

You need to complete four major tasks to provision Yealink IP phones.

The following figure shows an overview of four major provisioning tasks:



For more information on how to manage boot files, refer to Managing Boot Files.

For more information on how to manage configuration files, refer to Managing Configuration Files.

For more information on how to manage resource files, refer to Managing Resource Files.

For more information on how to configure a provisioning server, refer to Configuring a Provisioning Server.

For more information on how to obtain the provisioning server address, refer to Obtaining the Provisioning Server Address.

For more information on how to perform auto provisioning, refer to Triggering the IP Phone to Perform Auto Provisioning.

If you are not familiar with auto provisioning on Yealink IP phones, you can refer to An Instance of Auto Provision Configuration.

An Instance of Auto Provision Configuration

This section shows an instance of auto provision configuration.

1. Manage boot files.

   Specify the desired URL (e.g., tftp://10.2.5.193/network.cfg) of the configuration files in the boot file (e.g., y000000000000.boot). For more information, refer to Managing Boot Files.

```
#!version:1.0.0.1
## The header above must appear as-is in the first line


##[$MODEL]include:config <xxx.cfg>
##[$MODEL,$MODEL]include:config "xxx.cfg"

[T46S]include:config <tftp://10.2.5.193/network.cfg>
[T48S,T46G]include:config <../sip.cfg>
include:config "features.cfg"

overwrite_mode = 1
specific_model.excluded_mode=0
```

2. Manage configuration files.

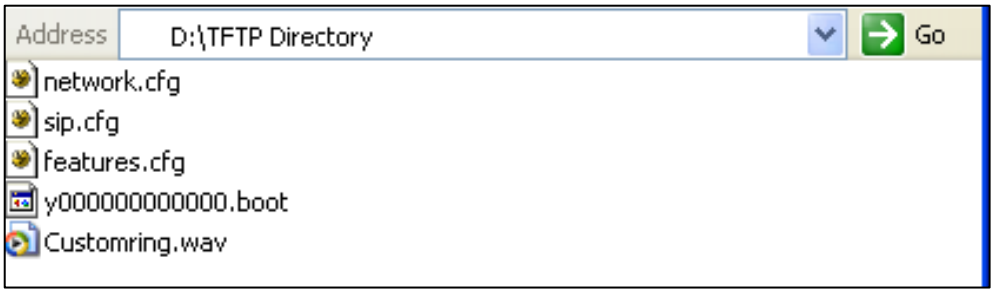
   Add/Edit the desired configuration parameters in the CFG file (e.g., features.cfg) you want the IP phone to download. For more information on how to manage configuration files, refer to Managing Configuration Files.

features.cfg

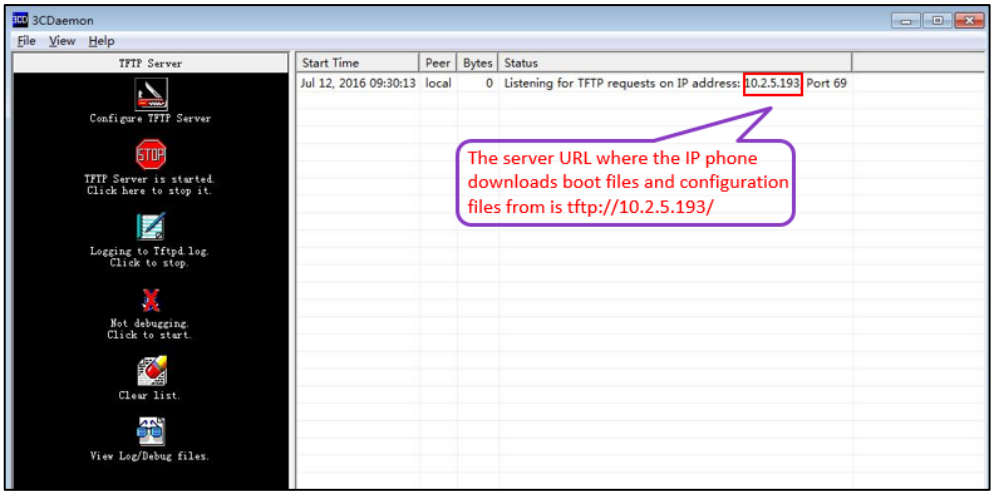
```
#!version:1.0.0.1
features.dnd_mode = 0
features.dnd.enable = 1
ringtone.url = tftp://10.2.5.193/Customring.wav
```

3. Configure the TFTP server.

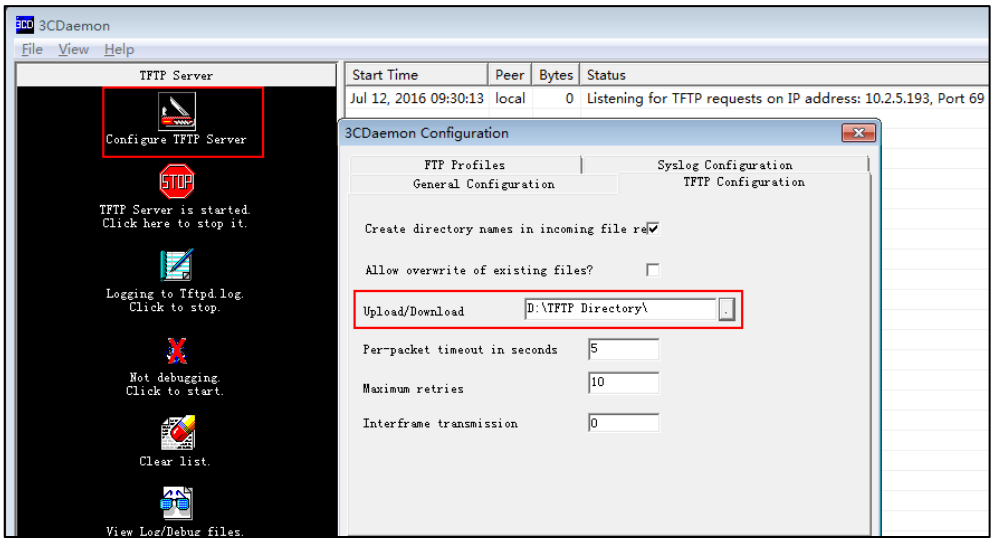
   1) Place boot files, configuration files and resource files to TFTP root directory (e.g., D:\TFTP Directory).



2) Start the TFTP sever. The IP address of the TFTP server is shown as below:

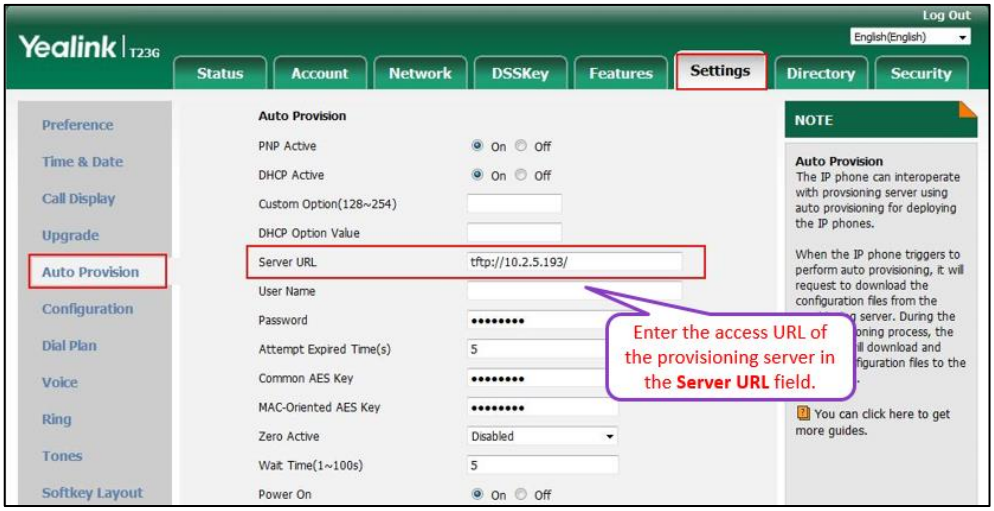


3) Select **Configure TFTP Server**. Click the ... button to locate the TFTP root directory in your local system.



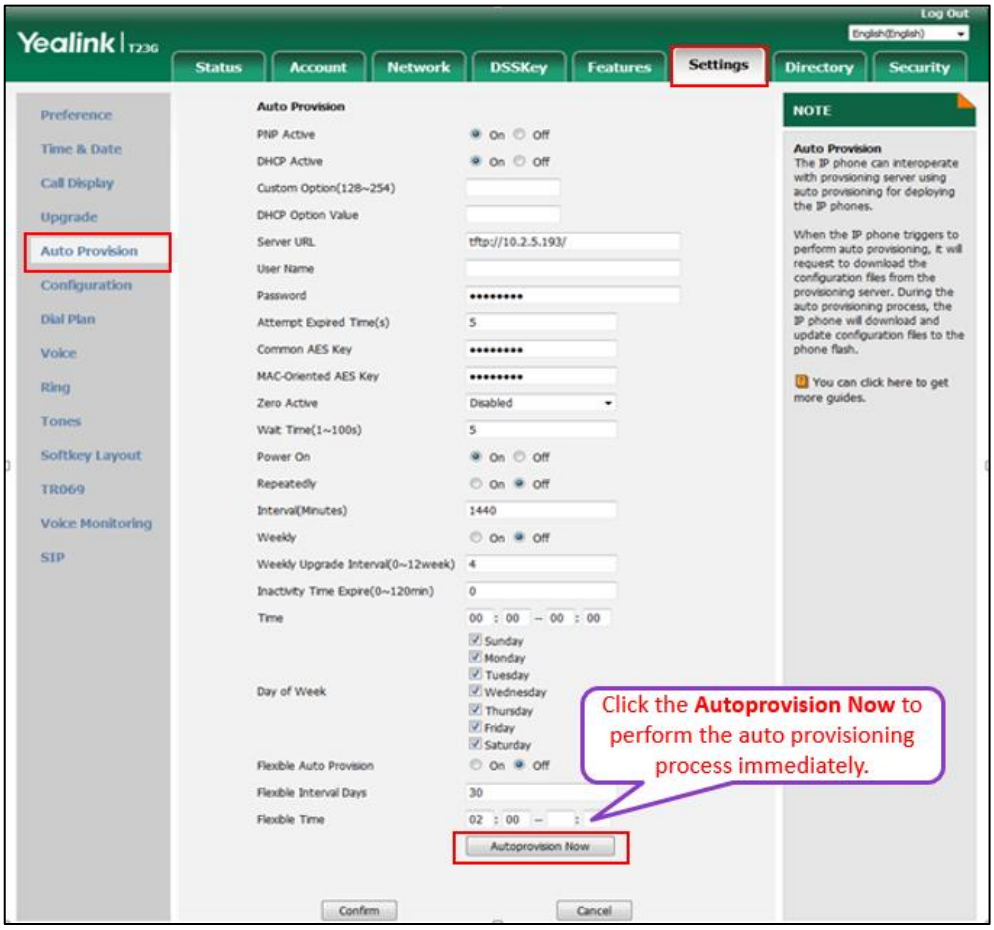
For more information on how to configure a provisioning server, refer to Configuring a Provisioning Server.

4. Configure the provisioning server address on the IP phone.



For more information on how to obtain the provisioning server address, refer to Obtaining the Provisioning Server Address.

5. Trigger the IP phone to perform auto provisioning.



For more information on how to trigger the phone to perform auto provisioning, refer to Triggering the IP Phone to Perform Auto Provisioning.

Managing Boot Files

Yealink IP phones can download CFG files referenced in the boot files. Before provisioning, you may need to edit and customize your boot files.

Yealink supports the following two types of boot files:

- MAC-Oriented boot file (e.g., 00156574b150.boot)
- Common boot file (y000000000000.boot)

You can edit the template boot file directly or create a new boot file as required. Open each boot file with a text editor such as Notepad++.

Editing Common Boot File

The common boot file is effective for all phones. It uses a fixed name "y000000000000.boot" as the file name.

The following figure shows the contents of the common boot file:

```
#!version:1.0.0.1
## The header above must appear as-is in the first line

include:config <xxx.cfg>
include:config "xxx.cfg"

overwrite_mode = 1
```

The following table lists guidelines you need to know when editing the boot file:

Item	Guidelines
#!version:1.0.0.1	It must be placed in the first line. Do not edit and delete.
## The header above must appear as-is in the first line	The line beginning with "#" is considered to be a comment. You can use "#" to make any comment in the boot file.
include:config <xxx.cfg> include:config "xxx.cfg"	**1)** Each "include" statement can specify a URL where a configuration file is stored. The configuration file format must be *.cfg. **2)** The URL in <> or "" supports the following two forms: • Relative URL (relative to the boot file): For example, sip.cfg, HTTP Directory/sip.cfg • Absolute URL: For example, http://10.2.5.258/HTTP Directory/sip.cfg

Item	Guidelines
	The URL must point to a specific CFG file. The CFG files are downloaded in the order listed (top to bottom). The parameters in the new downloaded configuration files will override the duplicate parameters in files downloaded earlier. **3)** The "include" statement can be repeated as many times as needed. **4)** The [$MODEL] can be added to specify settings for specific phone models. $MODEL represents the phone model name. The valid phone model names are: T58, T56A, CP960, T54S, T52S, T48S, T48G, T46S, T46G, T42S, T42G, T41P, T41S, T40P, T40G, T29G, T27G, T23P, T23G, T21P E2, T21 E2, T19P E2 and T19 E2. Multiple phone models are separated by commas. For example, [T46S, T23G]. It is only applicable to IP phones (except W53P/W56P) running firmware version 83 or later.
overwrite_mode	Enable or disable the overwrite mode. The overwrite mode is applied to the configuration files specified to download. Note that it only affects the parameters pre-provisioned via central provisioning. **1**-(Enabled) - If the value of a parameter in configuration files is left blank, or if a non-static parameter in configuration files is deleted or commented out, the factory default value takes effect. **0**-(Disabled) - If the value of a parameter in configuration files is left blank, deleted or commented out, the pre-configured value is kept. **Note**: This parameter can only be used in boot files. If a boot file is used but the value of the parameter "overwrite_mode" is not configured, the overwrite mode is enabled by default.
specific_model.excluded_mode	Enable or disable the exclude mode. The exclude mode applies to the configuration files specified in the boot file. **0**-Disabled (Append Mode), the phone downloads its own model-specific configuration files, and downloads other model-unspecified configuration files. **1**-Enabled (Exclude Mode), the phone attempts to download its own model-specific configuration files; if there is no own model-specific configuration files found on the server, it downloads model-unspecified configuration files. **Note**: Exclude mode can only be used in boot files. If a boot file is used but the value of the parameter "specific_model.excluded_mode" is not configured, the exclude mode is disabled by default. Exclude mode feature is only applicable to IP phones (except W53P/W56P) running firmware version 83 or later.

Creating MAC-Oriented Boot File

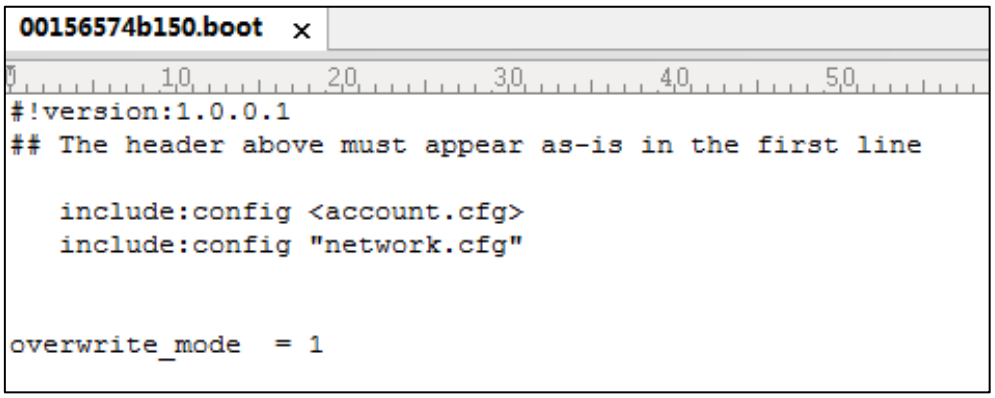
The MAC-Oriented boot file is only effective for the specific phone. It uses the 12-digit MAC address of the IP phone as the file name. For example, if the MAC address of the IP phone is 00156574B150, the MAC-Oriented boot file has to be named as 00156574b150.boot (case-sensitive) respectively.

If you want to create a MAC-Oriented boot file for your phone, follow these steps:

To create a MAC-Oriented boot file:

1. Create a boot file for your phone. Ensure the file complies with the guidelines that are listed in Editing Common Boot File.
2. Copy the contents from the common boot file and specify the configuration files to be downloaded.

   One or more configuration files can be referenced in the boot file. The following takes two configuration files for example:



3. Save the changes and close the MAC-Oriented boot file.

You can also make a copy of the common boot file, rename it and then edit it.

Managing Configuration Files

Auto provisioning enables Yealink IP phones to update themselves automatically via downloading Common CFG, MAC-Oriented CFG, custom CFG and MAC-local CFG files. Before provisioning, you may need to edit and customize your configuration files.

You can edit the template configuration files directly or create a new CFG file as required. Open each configuration file with a text editor such as Notepad++.

For more information on description of all configuration parameters in configuration files, refer to the latest Administrator Guide for your phone on Yealink Technical Support.

Editing Common CFG File

The Common CFG file is effective for all phones of the same model. It uses a fixed name "y0000000000XX.cfg" as the file name, where "XX" equals to the first two digits of the hardware version of the IP phone model.

The names of the common CFG file requirements for the phone are:

Product Name	Common CFG File
CP960	y000000000073.cfg
SIP-T58V/A	y000000000058.cfg
SIP-T56A	y000000000056.cfg
SIP-T54S	y000000000070.cfg
SIP-T52S	y000000000074.cfg
SIP-T48S	y000000000065.cfg
SIP-T46S	y000000000066.cfg
SIP-T42S	y000000000067.cfg
SIP-T41S	y000000000068.cfg
SIP-T48G	y000000000035.cfg
SIP-T46G	y000000000028.cfg
SIP-T42G	y000000000029.cfg
SIP-T41P	y000000000036.cfg
SIP-T40P	y000000000054.cfg
SIP-T40G	y000000000076.cfg
SIP-T29G	y000000000046.cfg

Product Name	Common CFG File
SIP-T27G	y000000000069.cfg
SIP-T23P/G	y000000000044.cfg
SIP-T21(P) E2	y000000000052.cfg
SIP-T19(P) E2	y000000000053.cfg
CP860	y000000000037.cfg
CP920	y000000000078.cfg
W53P/W60P	y000000000077.cfg
W52P/W56P	y000000000025.cfg

Common CFG file contains configuration parameters which apply to phones with the same model, such as language and volume.

The following figure shows a portion of the common CFG file:

```
#!version:1.0.0.1

##File header "#!version:1.0.0.1" can not be edited or deleted, and must be placed in the first line.##
##This template file is applicable to IP phones running firmware version 81 or later.##
##For more information on configuration parameters, refer to Description of Configuration Parameters in CFG Files.xlsx.##

##############################################################################
##                                Hostname                                  ##
##############################################################################
static.network.dhcp_host_name =


##############################################################################
##                            Network Advanced                              ##
##############################################################################
##It enables or disables the PC port.0-Disabled,1-Auto Negotiation.
##The default value is 1.It takes effect after a reboot.
static.network.pc_port.enable =

##It configures the transmission mode and speed of the Internet (WAN) port.
##0-Auto Negotiate
##1-Full Duplex 10Mbps
##2-Full Duplex 100Mbps
##3-Half Duplex 10Mbps
##4-Half Duplex 100Mbps
##5-Full Duplex 1000Mbps (only applicable to SIP-T48G/T46G/T46S/T42G/T29G/T23G/CP860 IP phones)
##The default value is 0.It takes effect after a reboot.
static.network.internet_port.speed_duplex =

##It configures the transmission mode and speed of the PC (LAN) port.
##0-Auto Negotiate
##1-Full Duplex 10Mbps
##2-Full Duplex 100Mbps
```

The following table lists guidelines you need to know when editing the common CFG file:

Item	Guidelines
#	The line beginning with "#" is considered to be a comment.
#!version:1.0.0.1	It must be placed in the first line. Do not edit and delete.
Filename	The filename complies with the requirements that are listed in the above table.
Line formats and Rules	Each line must use the following format and adhere to the following rules: *Configuration Parameter= Valid Value* • Separate each configuration parameter and value with an equal

Item	Guidelines
	sign. • Set only one configuration parameter per line. • Put the configuration parameter and value on the same line, and do not break the line. • The [$MODEL] can be added to the front of configuration parameter to specify the value for specific phone groups. $MODEL represents the phone model. The valid phone models are: T58, T56A, CP960, T54S, T52S, T48G, T48S, T46G, T46S, T42G, T42S, T41P, T41S, T40P, T40G, T29G, T27G, T23P, T23G, T21P E2, T21 E2, T19P E2 and T19 E2. Multiple phone models are separated by commas. For example, [T46S, T23G]. It is only applicable to IP phones (except W53P/W56P) running firmware version 83 or later. **Note**: The phone updates model-specific configurations and those model-unspecified configurations.

Editing MAC-Oriented CFG File

The MAC-Oriented CFG file is only effective for the specific phone. It uses the 12-digit MAC address of the IP phone as the file name. For example, if the MAC address of the IP phone is 00156574B150, the MAC-Oriented CFG file has to be named as 00156574b150.cfg (case-sensitive) respectively.

MAC-Oriented CFG file contains configuration parameters which are expected to be updated per phone, such as the registration information.

The following figure shows a portion of the MAC-Oriented CFG file:

```
1#!version:1.0.0.1

##File header "#!version:1.0.0.1" can not be edited or deleted, and must be placed in the first line.##
##This template file is applicable to IP phones running firmware version 81 or later.##
##For more information on configuration parameters, refer to Description of Configuration Parameters in CFG Files.xslx##

##############################################################################################
##                              Account1 Basic Settings                                     ##
##############################################################################################
account.1.enable =
account.1.label =
account.1.display_name =
account.1.auth_name =
account.1.user_name =
account.1.password =
account.1.outbound_proxy_enable =
account.1.outbound_host =
account.1.outbound_port =
account.1.dial_tone =



##It configures the transport type for account 1. 0-UDP,1-TCP,2-TLS,3-DNS-NAPTR
##The default value is 0.
account.1.sip_server.1.transport_type =
account.1.sip_server.2.transport_type =

##############################################################################################
##                                 Failback                                                 ##
##############################################################################################

account.1.naptr_build =
account.1.fallback.redundancy_type =
account.1.fallback.timeout =
account.1.sip_server.1.address =
```

The following table lists guidelines you need to know when editing the MAC-Oriented CFG file:

Item	Guidelines
#	The line beginning with "#" is considered to be a comment.
#!version:1.0.0.1	It must be placed in the first line. Do not edit and delete.
Filename	The filename matches the MAC address of your phone.
Line formats and Rules	Each line must use the following format and adhere to the following rules: *Configuration Parameter= Valid Value* • Separate each configuration parameter and value with an equal sign. • Set only one configuration parameter per line. • Put the configuration parameter and value on the same line, and do not break the line. • The [$MODEL] can be added to the front of configuration parameter to specify the value for specific phone groups. $MODEL represents the phone model. The valid phone models are: T58, T56A, CP960, T54S, T52S, T48G, T48S, T46G, T46S, T42G, T42S, T41P, T41S, T40P, T40G, T29G, T27G, T23P, T23G, T21P E2, T21 E2, T19P E2 and T19 E2. Multiple phone models are separated by commas. For example, [T46S, T23G]. It is only applicable to IP phones (except W53P/W56P) running firmware version 83 or later. **Note**: The phone updates model-specific configurations and those model-unspecified configurations.

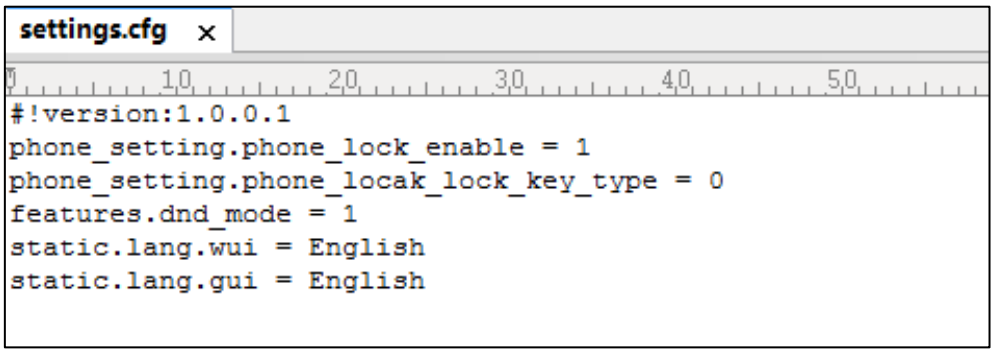
SIP-T58V/T58A/T56A/T54S/T48G/T48S/T46G/T46S/T29G IP phones support 16 accounts, SIP-T52S/T42G/T42S IP phones support 12 accounts, W53P/W60P DECT IP phones support 8 accounts. SIP-T41P/T41S/T27G IP phones support 6 accounts, W52P/W56P IP DECT phones support 5 accounts; SIP-T40P/T40G/T23P/T23G IP phones support 3 accounts, SIP-T21(P) E2 IP phones support 2 accounts, CP960/CP920/CP860/SIP-T19(P) E2 IP phones support only one account.

Creating a New CFG File

If you want to create a new CFG file for your phone, follow these steps:

To create a new CFG file:

1. Create a CFG file for your phone. Ensure the file complies with the guidelines that are listed in Editing Common CFG File or Editing MAC-Oriented CFG File.
2. Copy configuration parameters from the template configuration files and set the valid values for them.



3. (Optional.) Specify different parameter values for specific phone groups.

   For example:

   [T46S] features.dnd_mode = 1

   [T48G, T23G] features.dnd_mode = 0

4. Save the changes and close the CFG file.

You can also make a copy of the template configuration file, rename it and then edit it.

Managing MAC-local CFG File

By default, MAC-local CFG file automatically stores non-static settings modified via web user interface or phone user interface. This file is stored locally on the IP phone, but a copy can also be uploaded to the provisioning server (or a specified URL configured by "static.auto_provision.custom.sync.path"). This file enables the phone to keep user's personalization settings, even after auto provisioning. As with the MAC-Oriented CFG files, MAC-local CFG files are only effective for the specific phone. They use the 12-digit MAC address of the IP phone as the file name. For example, if the MAC address of the IP phone is 00156574B150, MAC-local CFG file has to be named as 00156574b150-local.cfg (case-sensitive).

If your IP phone with the current firmware version cannot generate a <MAC>-local.cfg file, the IP phone will automatically generate a MAC-local CFG file after it is upgraded to the latest firmware.

For more information on how to keep user's personalization settings, refer to the latest Administrator Guide for your phone on Yealink Technical Support.

We recommend you do not edit the MAC-local CFG file. If you really want to edit MAC-local CFG file, you can export and then edit it.
For more information on how to export CFG files, refer to the latest Administrator Guide for your phone on Yealink Technical Support.

Encrypting Configuration Files

To protect against unauthorized access and tampering of sensitive information (e.g., login password, registration information), you can encrypt configuration files using Yealink Configuration Encryption Tool. AES keys must be 16 characters and the supported characters contain: 0 ~ 9, A ~ Z, a ~ z and the following special characters are also supported: # $ % * + , - . : = ? @ [ ] ^ _ { } ~. For more information on how to encrypt configuration files, refer to *Yealink Configuration Encryption Tool User Guide*.

Managing Resource Files

Before provisioning, you may need to edit and customize your resource files.

You can edit the template resource files directly or create a new resource file as required. Open each resource file with a text editor such as Notepad++.

Customizing Resource Files

The resource files are effective for all phones of the same model or the specific phone. If the resource file is to be used for all IP phones of the same model, the access URL of resource file had better to be specified in the common CFG file. However, if you want to specify the desired phone to use the resource file, the access URL of resource file should be specified in the MAC-Oriented CFG file.

Refer to Resource Files to get support resource files:

For more information on how to customize these template resource files and an explanation of the configuration parameters that related to these features, refer to the latest Administrator Guide for your phone on Yealink Technical Support.

Configuring a Provisioning Server

Yealink IP phones support using FTP, TFTP, HTTP and HTTPS protocols to download boot files and configuration files. You can use one of these protocols for provisioning. The TFTP protocol is used by default. The following section provides instructions on how to configure a TFTP server.

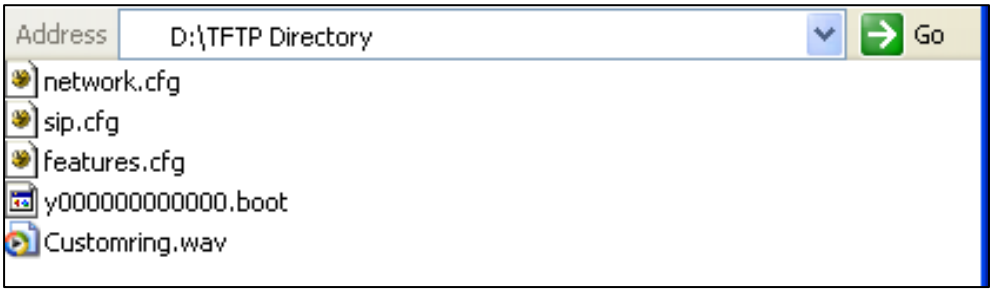
We recommend that you use 3CDaemon or TFTPD32 as a TFTP server. 3CDaemo and TFTPD32 are free applications for Windows. You can download 3CDaemon online: http://www.oldversion.com/3Com-Daemon.html and TFTPD32 online: http://tftpd32.jounin.net/.

For more information on how to configure FTP and HTTP servers, refer to Configuring an FTP Server and Configuring an HTTP Server.

Preparing a Root Directory

To prepare a root directory:

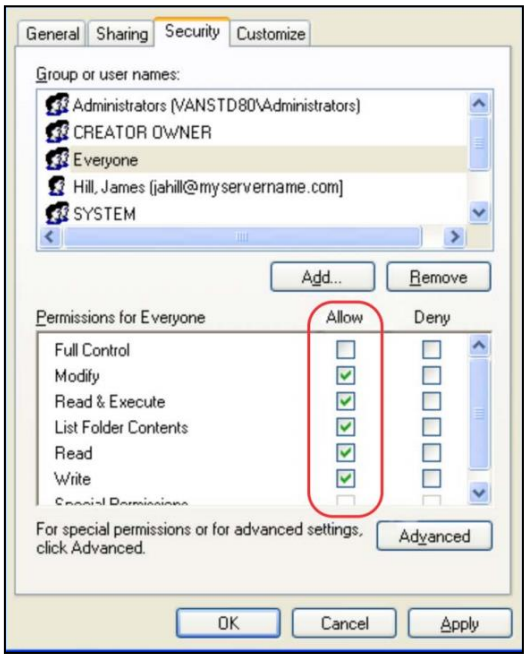
1. Create a TFTP root directory on the local system (e.g., D:\TFTP Directory).
2. Place the boot files, configuration files and resource files to this root directory.



3. (Optional.) Set security permissions for the TFTP directory folder.

   You need to define a user or a group name, and set the permissions: read, write or modify. Security permissions vary by organizations.

An example of configuration on the Windows platform is shown as below:



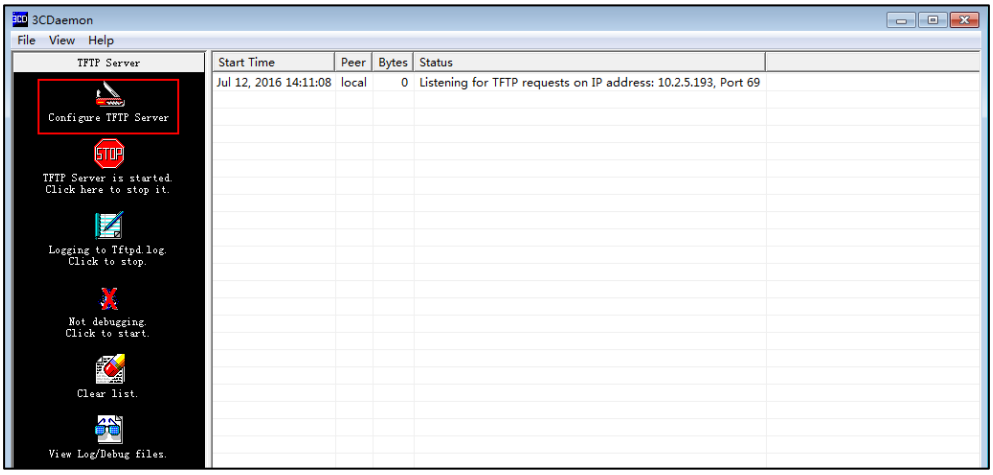
Configuring a TFTP Server

If you have a 3CDaemon application installed on your local system, use it directly. Otherwise, download and install it.

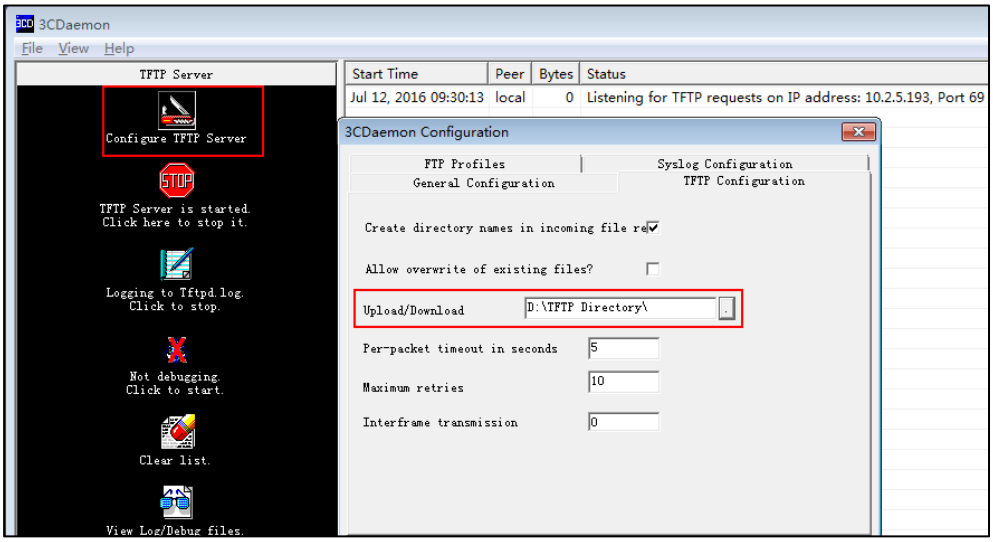
To configure a TFTP server:

1. Double click **3CDaemon.exe** to start the application.

   A configuration page is shown as below:



2. Select **Configure TFTP Server**. Click the ... button to locate the TFTP root directory from your local system:



3. Click the **Confirm** button to finish configuring the TFTP server.

The server URL "tftp://IP/" (Here "IP" means the IP address of the provisioning server, for example, "tftp://10.2.5.193/") is where the IP phone downloads configuration files from.

Obtaining the Provisioning Server Address

Yealink IP phones can obtain the provisioning server address in the following ways:

- Zero Touch
- Plug and Play (PnP) Server
- DHCP Options
- Phone Flash
- Configuring Wildcard of the Provisioning Server URL

The priority of obtaining the provisioning server address is as follows: Zero Touch>PnP Server>DHCP Options (for IPv4: IPv4 Custom option>option 66>option 43; for IPv6: IPv6 Custom option>option 59) >Phone Flash. The following sections detail the process of each way (take the SIP-T23G IP phone as an example).

IPv6 custom option is only applicable to SIP-T54S, SIP-T52S, SIP-T48G/S, SIP-T46G/S, SIP-T42G/S, SIP-T41P/S, SIP-T40P/G, SIP-T29G, SIP-T27G, SIP-T23P/G, SIP-T21(P) E2 and SIP-T19(P) E2 IP phones running firmware version 83 or later.

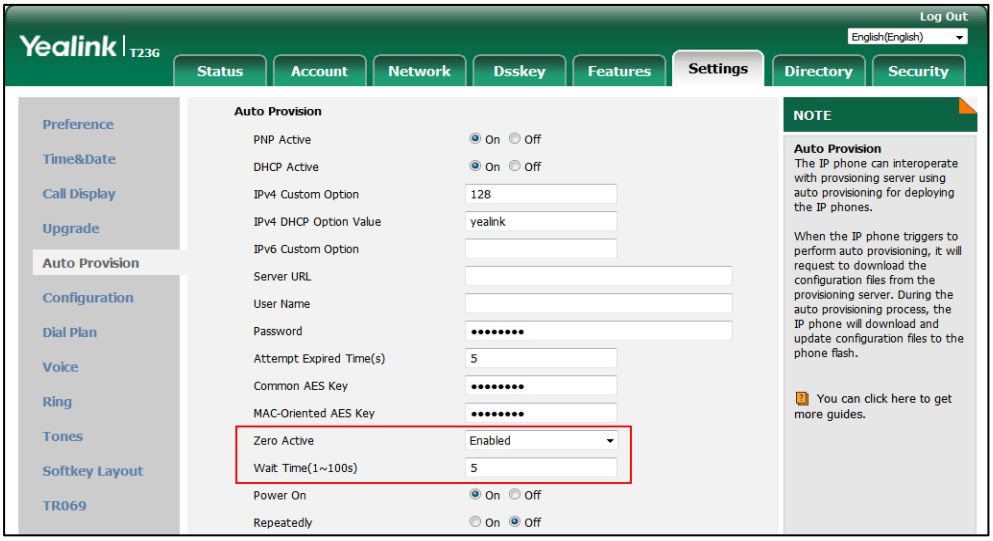
Zero Touch

Zero Touch allows you to configure the network parameters and provisioning server address via phone user interface during startup. This feature is helpful when there is a system failure on the IP phone. To use Zero Touch, make sure this feature is enabled. This feature is not applicable to W52P/W53P/W56P/W60P IP phones.

To configure zero touch via web user interface:

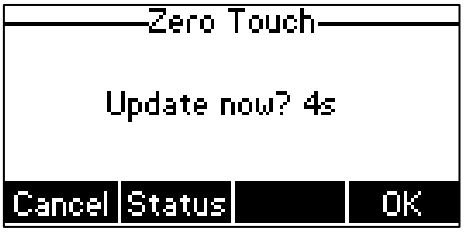
1. Click on **Settings**->**Auto Provision**.
2. Select **Enabled** from the pull-down list of **Zero Active**.
3. Enter the desired waiting time in the **Wait Time(1~100s)** field.

The default value is 5.



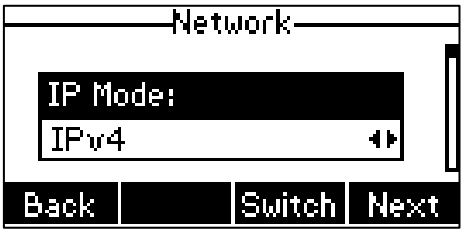
4. Click **Confirm** to accept the change.

When Zero Touch is enabled, there will be a configuration wizard during startup:



Press the **OK** soft key.

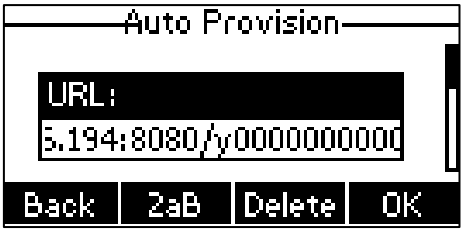
The network parameters are configurable via phone user interface:



Press the **Next** soft key after finishing network settings.

Configure the provisioning server address, authentication user name (optional) and password (optional) in the **Auto Provision** screen.

An example of screenshot is shown as below:



Press the **OK** soft key.

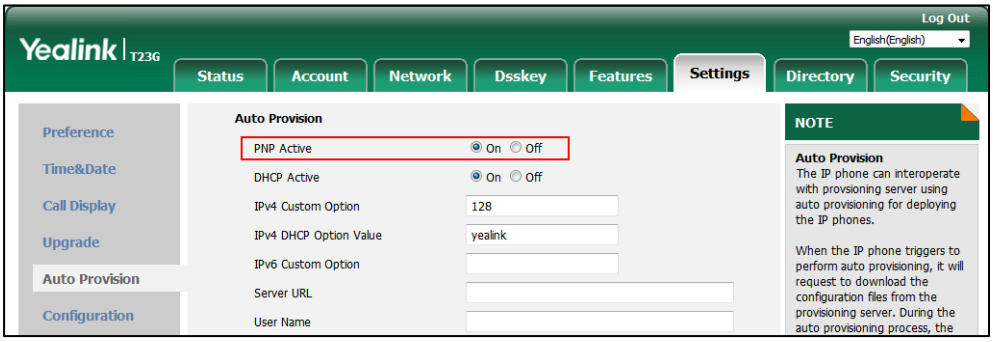
After the above configuration is completed, the IP phone will connect to the configured provisioning server and perform auto provisioning during startup.

Plug and Play (PnP) Server

Yealink IP phones support obtaining the provisioning server address from the PnP server. The IP phone broadcasts the PnP SUBSCRIBE message to obtain the provisioning server address during startup. To use Plug and Play, make sure this feature is enabled.

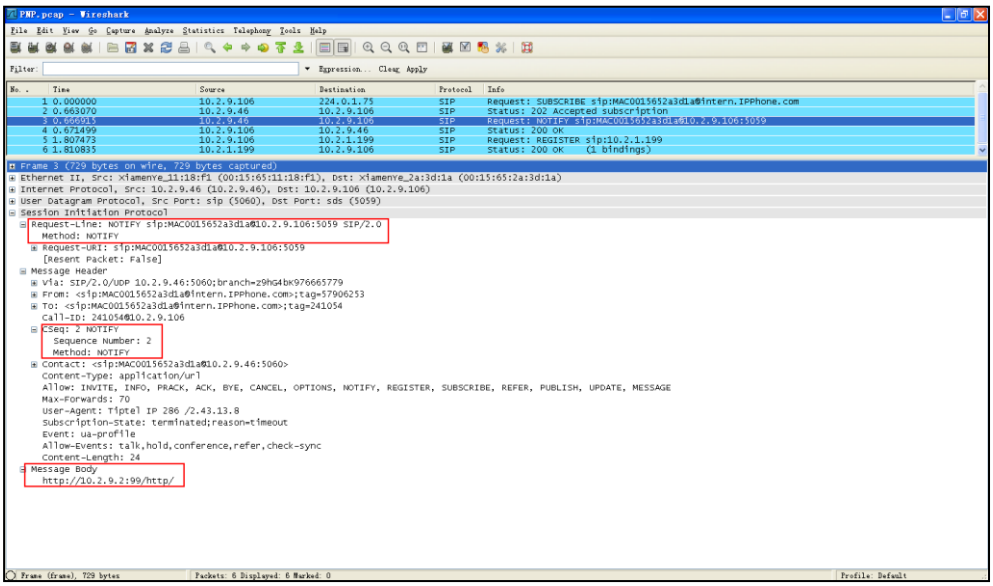
To configure PnP via web user interface:

1. Click on **Settings**->**Auto Provision**.
2. Mark the **On** radio box in the **PNP Active** field.



3. Click **Confirm** to accept the change.

   Any PnP server activated in the network responses with a **SIP NOTIFY** message, and an address of the provisioning server is contained in the message body.



After the IP phone obtains the provisioning server address from the PNP server, it will connect to the provisioning server and perform auto provisioning during startup.

DHCP Options

Yealink IP phones can obtain the provisioning server address by detecting DHCP options during startup.

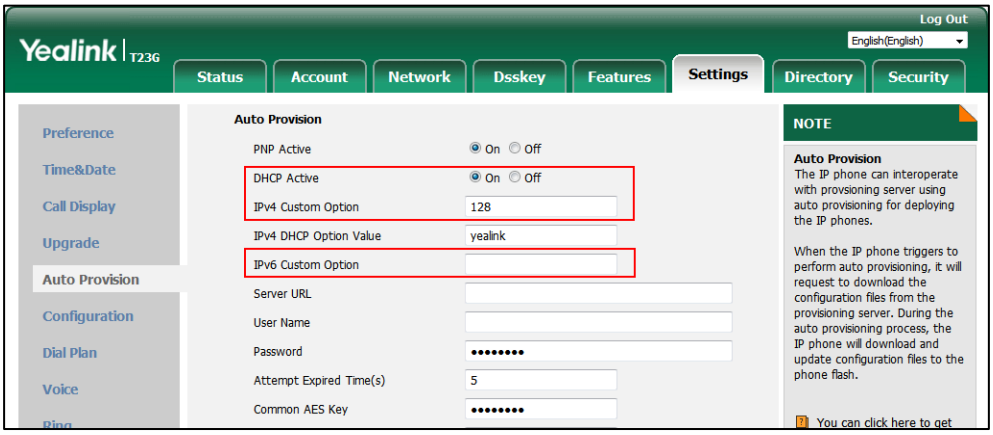
If you are using IPv4 network, the phone will automatically detect the option 66 and option 43 for obtaining the provisioning server address. DHCP option 66 is used to identify the TFTP server. DHCP option 43 is a vendor-specific option, which is used to transfer the vendor-specific information.

If you are using IPv6 network, the phone will automatically detect the option 59 for obtaining the provisioning server address. DHCP option 59 is used to specify a URL for the boot file to be downloaded by the client.

You can configure the phone to obtain the provisioning server address via a custom DHCP option. You can select to use IPv4 or IPv6 custom DHCP option according to your network environment. To obtain the provisioning server address via an IPv4 or IPv6 custom DHCP option, make sure the DHCP option is properly configured on the phone. The IPv4 or IPv6 custom DHCP option must be in accordance with the one defined in the DHCP server.

To configure the DHCP option via web user interface:

1. Click on **Settings**->**Auto Provision**.
2. Mark the **On** radio box in the **DHCP Active** field.
3. If you are using IPv4 network, enter the desired value in the **IPv4 Custom Option** field.
4. If you are using IPv6 network, enter the desired value in the **IPv6 Custom Option** field.



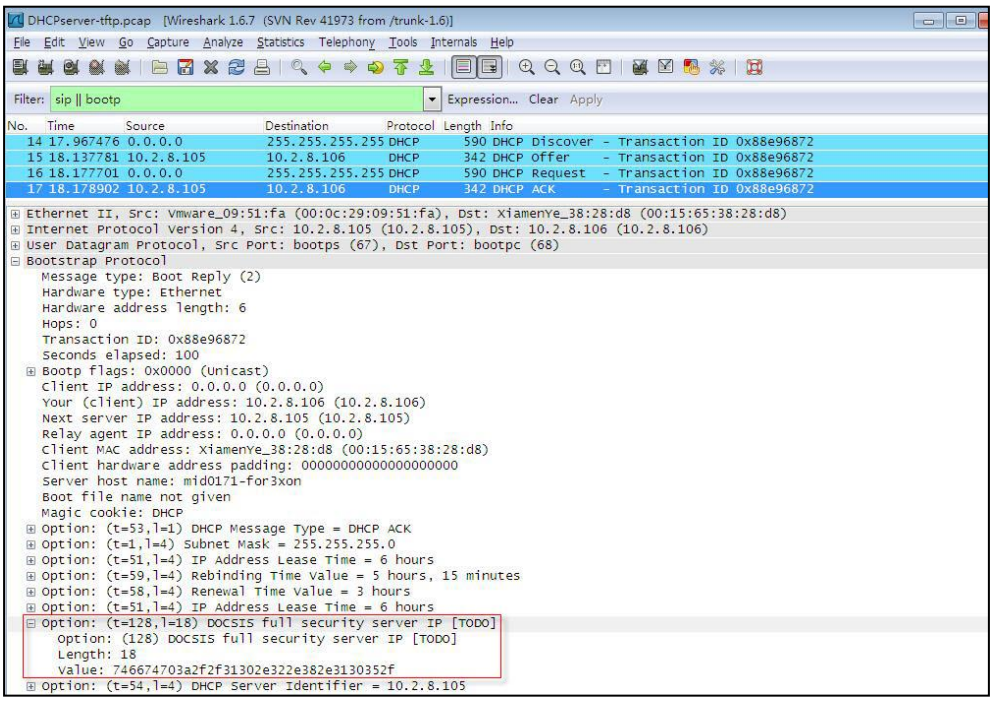
5. Click **Confirm** to accept the change.

During startup, the phone will broadcast DHCP request with DHCP options for obtaining the provisioning server address. The provisioning server address will be found in the received DHCP response message.

After the IP phone obtains the provisioning server address from the DHCP server, it will connect to the provisioning server and perform auto provisioning during startup.

For more information on the DHCP options, refer to the latest Administrator Guide for your phone on Yealink Technical Support.

The following figure shows the example messages of obtaining the TFTP server address from an IPv4 custom DHCP option:



Right click the root node of the custom option (e.g., option 128) shown on the above figure, and select **Copy**->**Bytes**->**Printable Text Only**. Paste the copied text in your favorite text editor to check the address, for example, tftp://192.168.1.100/.

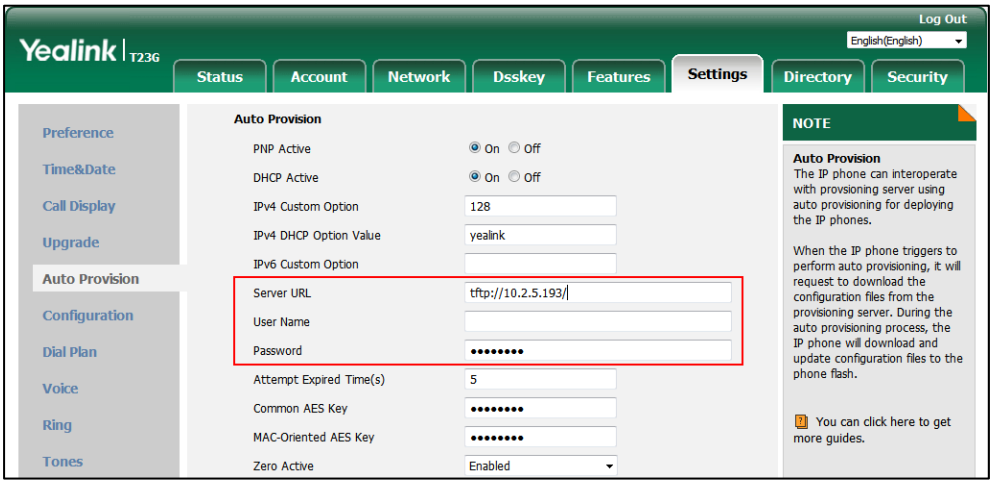
Phone Flash

Yealink IP phones can obtain the provisioning server address from the IP phone flash. To obtain the provisioning server address by reading the IP phone flash, make sure the configuration is set properly.

To configure the IP phone flash via web user interface:

1. Click on **Settings**->**Auto Provision**.

2. Enter the URL, user name and password of the provisioning server in the **Server URL**, **User Name** and **Password** field respectively (the user name and password are optional).



3. Click **Confirm** to accept the change.

After the above configuration is completed, the IP phone will connect to the configured provisioning server and perform auto provisioning by one of the following methods: Power On, Repeatedly, Weekly, Flexible Auto Provision, Auto Provision Now, SIP NOTIFY Message and Multi-mode Mixed. For more information on these methods, refer to Triggering the IP Phone to Perform Auto Provisioning.

Configuring Wildcard of the Provisioning Server URL

Normally, many phone models may be deployed in your environment. To deploy many phone models using a unified provisioning server, it is convenient for the administrator to configure a unified provisioning server URL for different phone models. On the provisioning server, many directories need to be configured for different phone models, each with a unique directory name. Yealink IP phones support the following wildcards in the provisioning server URL:

- **$PN**: it is used to identify the directory name of the provisioning server directory where the corresponding boot files and configuration files are located.
- **$MAC**: it is used to identify the MAC address of the IP phone.

The parameter "static.auto_provision.url_wildcard.pn" is used to configure the directory name where the boot files and configuration files located. For more information on the parameter, refer to the latest IP Phones Description of Configuration Parameters in CFG Files or Administrator Guide for your phone on Yealink Technical Support.

When the IP phone obtains a provisioning server URL containing the wildcard $PN, it automatically replaces the character $PN with the value of the parameter "static.auto_provision.url_wildcard.pn" configured on the IP phone. When the IP phone is triggered to perform auto provisioning, it will request to download the boot files and configuration files from the identified directory on the provisioning server.

The value of the parameter "static.auto_provision.url_wildcard.pn" must be configured in accordance with the directory name of the provisioning server directory where the boot files and configuration files of the IP phones are located.

The following example assists in explaining the wildcard feature:

You want to deploy SIP-T42G and SIP-T46G IP phones simultaneously in your environment. IP phones are configured to obtain the provisioning server URL via DHCP option 66. The following details how to deploy the SIP-T42G and SIP-T46G IP phones using wildcard feature.

1. Create two directories on the root directory of provisioning server.
2. Configure the directory names of these two directories to be "T42G" and "T46G".
3. Place the associated boot files and configuration files to the directory created above.
4. Configure the value of DHCP option 66 on the DHCP server as: tftp://192.168.1.100/$PN.
5. Configure the value of the parameter "static.auto_provision.url_wildcard.pn".

   The default value of the parameter "static.auto_provision.url_wildcard.pn" is ''T42G'' for the SIP-T42G IP phones and ''T46G'' for the SIP-T46G IP phones. If the default value is different from the directory name, you need to configure the value of this parameter to be the directory name on the IP phones in advance.

During startup, IP phones obtain the provisioning server URL "tftp://192.168.1.100/$PN" via DHCP option 66, and then replace the character "$PN" in the URL with "T42G" for the SIP-T42G IP phones and "T46G" for the SIP-T46G IP phones. When performing auto provisioning, the SIP-T42G IP phones and the SIP-T46G IP phones first request to download the MAC-Oriented boot files and configuration files referenced in MAC-Oriented boot files from the provisioning server address "tftp://192.168.1.100/T42G" and "tftp://192.168.1.100/T46G" respectively. If no matched MAC-Oriented boot files are found on the server, the SIP-T42G IP phones and the SIP-T46G IP phones request to download the common boot files and configuration files referenced in common boot files from the provisioning server address "tftp://192.168.1.100/T42G" and "tftp://192.168.1.100/T46G" respectively.

If the URL is configured as "tftp://192.168.1.100/$PN/$MAC.boot" on the DHCP server, the SIP-T42G IP phones and the SIP-T46G IP phones will replace the characters "$PN" with "T42G" and "T46G" respectively, and replace the characters "$MAC" with their MAC addresses. For example, the MAC address of one SIP-T42G IP phone is 00156543EC97. When performing auto provisioning, the IP phone will only request to download the 00156543ec97.boot file and configuration files referenced in the 00156543ec97.boot file from the provisioning server address "tftp://192.168.1.100/T42G".

For more information on boot files, refer to Managing Boot Files.

Triggering the IP Phone to Perform Auto Provisioning

This chapter introduces the following methods to trigger the IP phone to perform auto provisioning:

- Power On
- Repeatedly
- Weekly
- Flexible Auto Provision
- Auto Provision Now
- Multi-mode Mixed
- SIP NOTIFY Message
- Auto Provisioning via Activation Code

When there is an active call on the IP phone during auto provisioning, the IP phone will detect the call status every 30 seconds. If the call is released within 2 hours, the auto provisioning will be performed as usual. Otherwise, the process will be ended due to timeout.

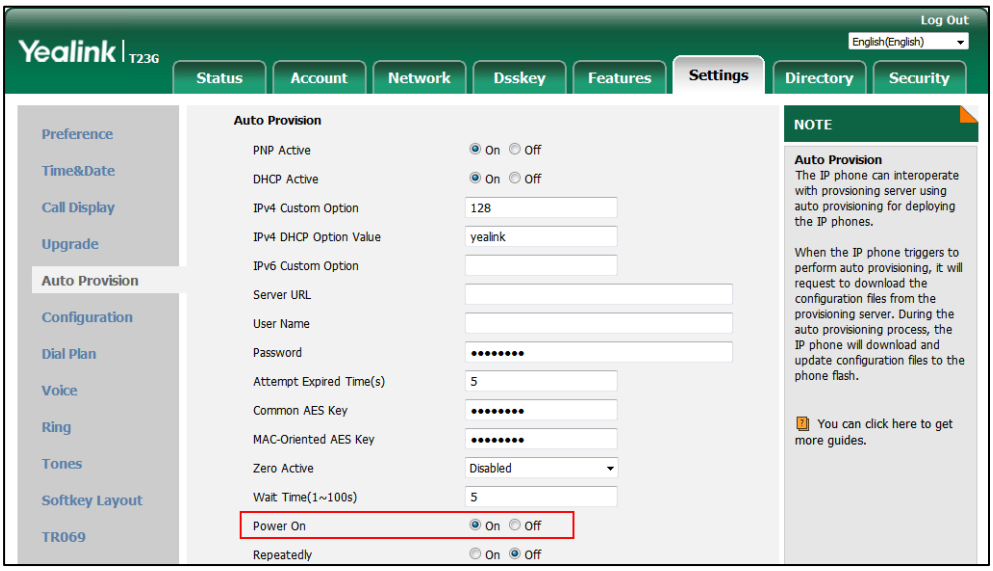
Power On

The IP phone performs the auto provisioning when the IP phone is powered on.

To activate the power on mode via a web user interface:

1. Click on **Settings**->**Auto Provision**.

2. Mark the **On** radio box in the **Power On** field.



3. Click **Confirm** to accept the change.

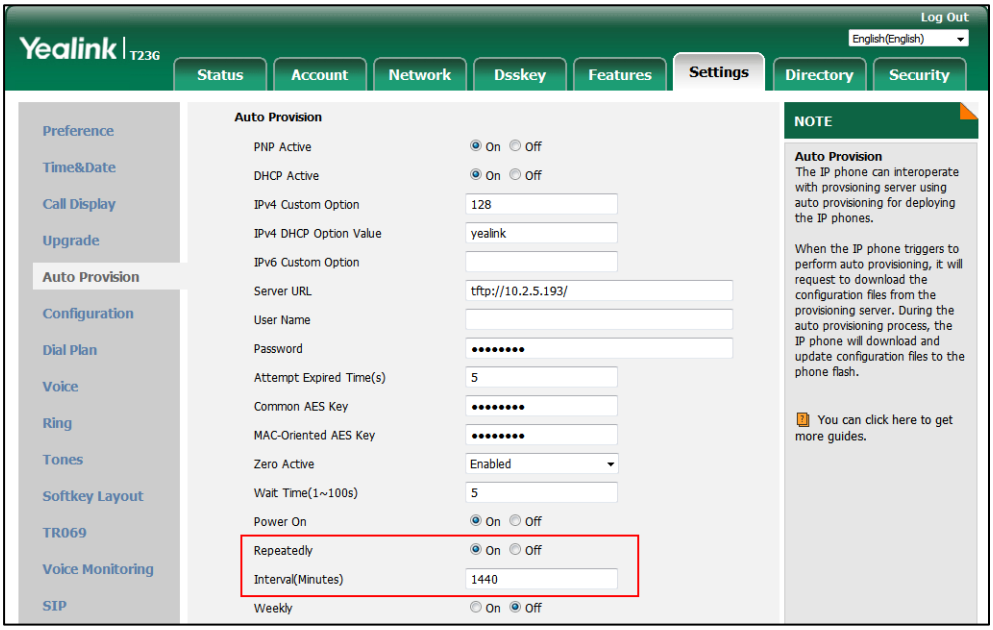
Repeatedly

The IP phone performs the auto provisioning at regular intervals. You can configure the interval for the repeatedly mode. The default interval is 1440 minutes.

To activate the repeatedly mode via web user interface:

1. Click on **Settings**->**Auto Provision**.
2. Mark the **On** radio box in the **Repeatedly** field.

3. Enter the desired interval time (in minutes) in the **Interval(Minutes)** field.



4. Click **Confirm** to accept the change.

Weekly

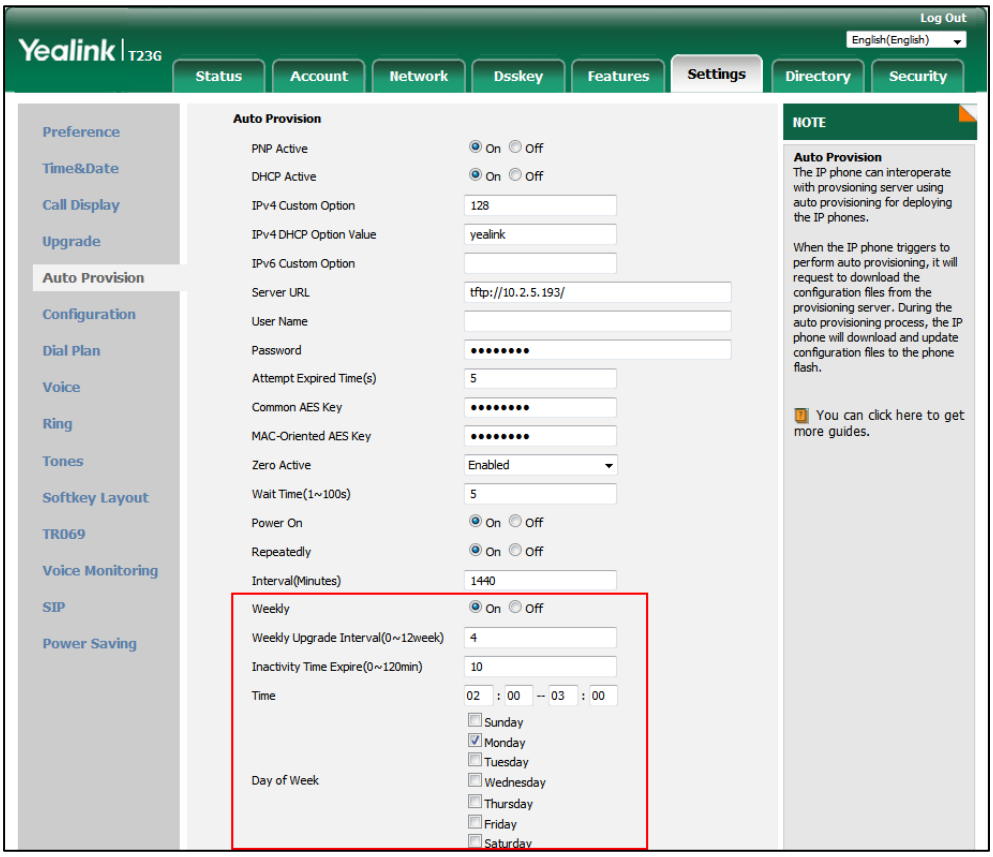
The IP phone performs auto provisioning at a random time every week/month/quarter. You can configure what time of the day and which day of the week to trigger the IP phone to perform auto provisioning. You can also configure a regular week interval to trigger the IP phone to perform auto provisioning. You can specify the delay time to perform auto provisioning when the IP phone is inactive at regular week. For example, you can configure the IP phone to check and update new configuration only when the IP phone has been inactivated for 10 minutes between 2 to 3 o'clock in the morning every Monday at a 4-week interval.

If you configure two or more days in a week, the auto provisioning only occurs at a random day.

To activate the weekly mode via web user interface:

1. Click on **Settings**->**Auto Provision**.
2. Mark the **On** radio box in the **Weekly** field.
3. Enter the desired upgrade interval in the **Weekly Upgrade Interval(0~12week)** field.
4. Enter the desired value in the **Inactivity Time Expire(0~120min)** field.
5. Enter the desired time in the **Time** field.

6. Check one or more checkboxes in the **Day of Week** field.



7. Click **Confirm** to accept the change.

Flexible Auto Provision

The IP phone performs auto provisioning at a random time on a random day within a specific period of time. The random day is calculated on the basis of the phone's MAC address. You can specify an interval and configure what time of the day to trigger the IP phone to perform auto provisioning.

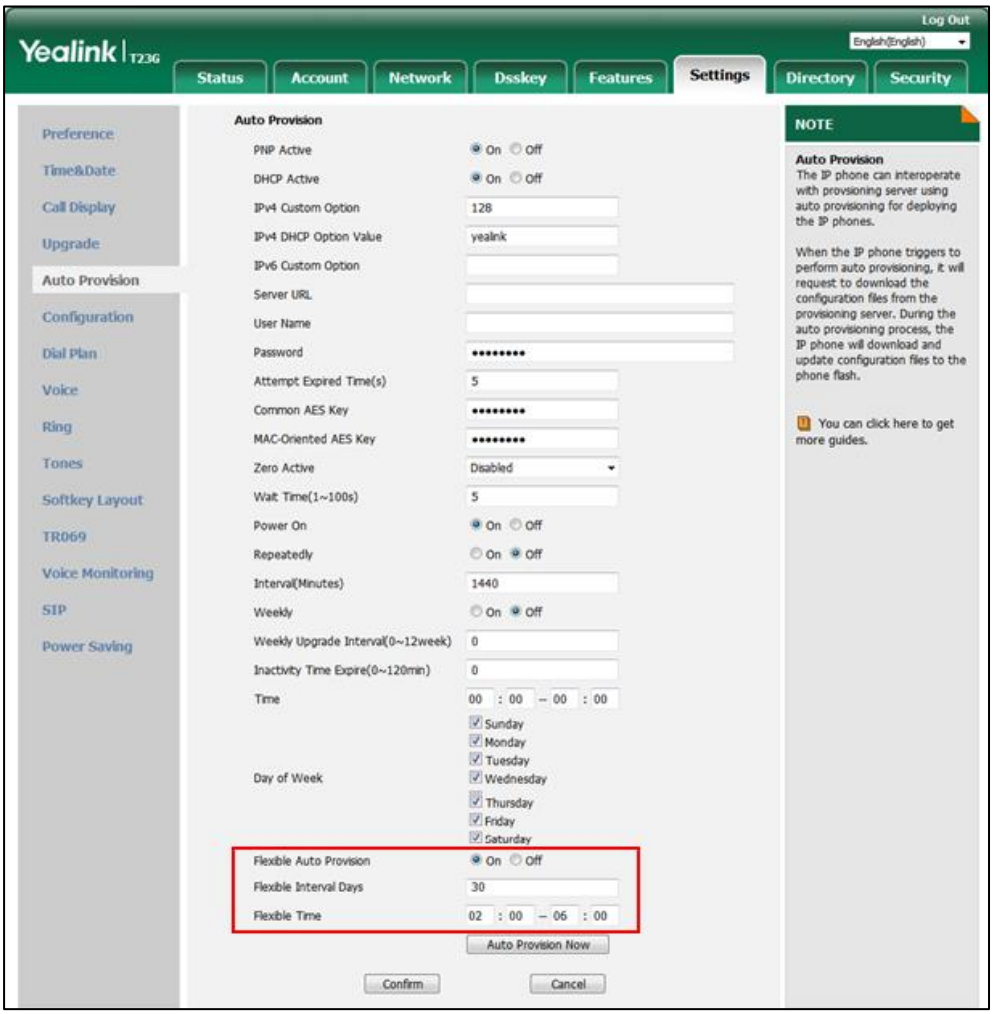
For example, you can configure the IP phone to check and update new configuration between 1 and 6 o'clock in the morning at a 30-day interval. The IP phone will perform auto provisioning at a random time (e.g., 03:47) on a random day (e.g., 18) based on the phone's MAC address.

Note that the update time will be recalculated if auto provisioning occurs (e.g., Auto Provision Now) during this specific period of time.

To activate the flexible auto provision mode via web user interface:

1. Click on **Settings**->**Auto Provision**.
2. Mark the **On** radio box in the **Flexible Auto Provision** field.
3. Enter the desired value in the **Flexible Interval Days** field.

4. Enter the desired start time and end time in the **Flexible Time** field.



5. Click **Confirm** to accept the change.

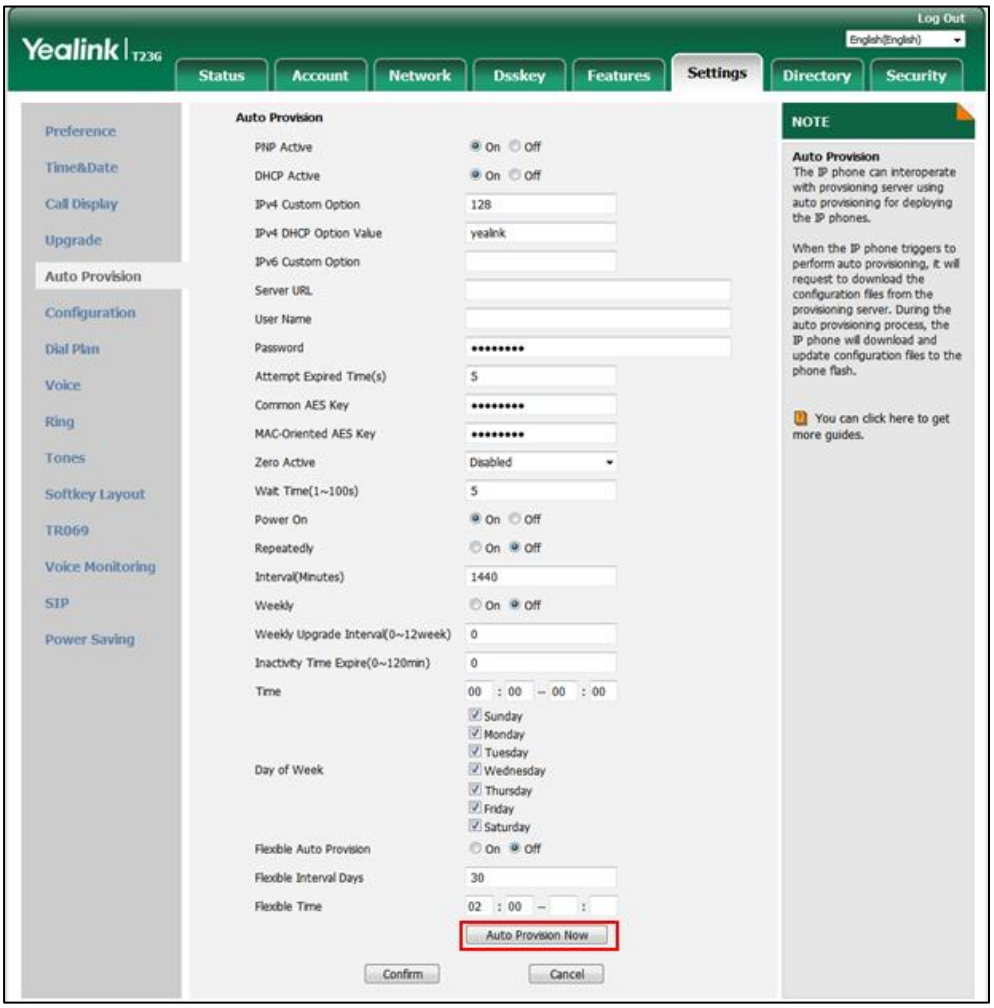
Auto Provision Now

You can use auto provision now mode to manually trigger the IP phone to perform auto provisioning immediately.

To use the auto provision now mode via web user interface:

1. Click on **Settings**->**Auto Provision**.

2. Click **Auto Provision Now**.



The IP phone will perform auto provisioning immediately.

Multi-mode Mixed

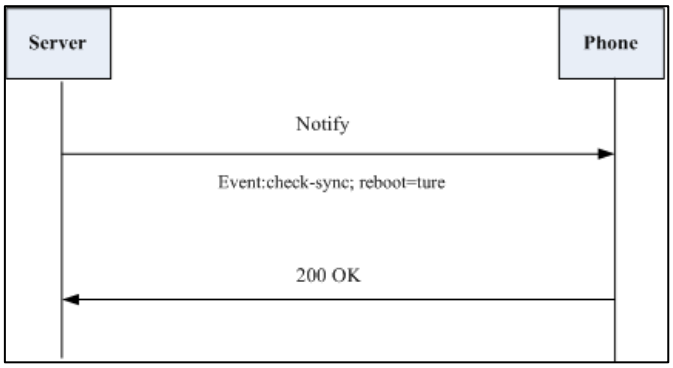
You can activate more than one method for auto provisioning. For example, you can activate the "Power On" and "Repeatedly" modes simultaneously. The IP phone will perform auto provisioning when it is powered on and at a specified interval.

SIP NOTIFY Message

The IP phone will perform auto provisioning when receiving a SIP NOTIFY message which contains the header "Event: check-sync". Whether the IP phone reboots or not depends on the value of the parameter "sip.notify_reboot_enable". If the value is set to 1, or the value is set to 0 and the header of the SIP NOTIFY message contains an additional string "reboot=true", the IP phone will reboot immediately. For more information on the parameter "sip.notify_reboot_enable", refer to the latest IP Phones Description of Configuration Parameters

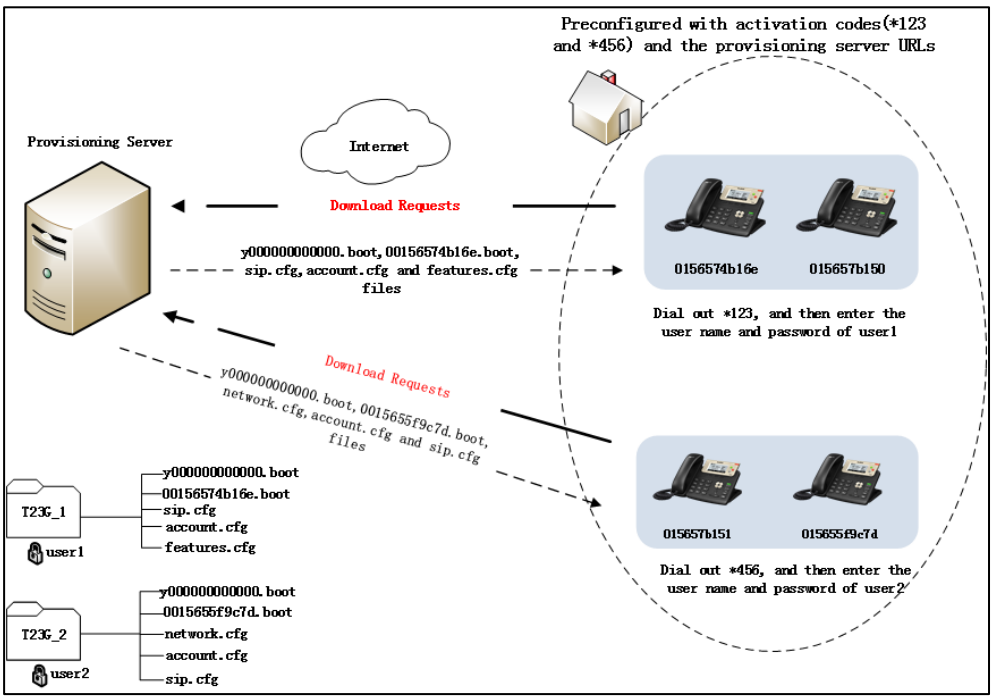
in CFG Files or Administrator Guide for your phone on Yealink Technical Support. This method requires server support.

The following figure shows the message flow:



Auto Provisioning via Activation Code

In addition to the updating modes introduced above, users can trigger IP phones to perform auto provisioning by dialing an activation code. To use this method, the activation code and the provisioning server URL need to be pre-configured on the IP phones. This method works only if there is no registered account on the IP phone. It is usually used for IP phones distributed by retail sales. It has the advantage that the IP phones do not need to be handled (e.g., registering account) before sending them to end-users.



The following lists the processes for triggering auto provisioning via activation code:

1. Create multiple directories (e.g., T23G_1 and T23G_2) on the provisioning server.
2. Store boot files and configuration files to each directory on the provisioning server.
3. Configure a user name and password for each directory on the provisioning server.

   The user name and password provides a means of conveniently partitioning the boot files and configuration files for different IP phones. To access the specified directory, you need to provide the correct user name and password configured for the directory.
4. Configure unique activation codes and the provisioning server URLs on IP phones.

   The activation code can be numeric characters, special characters "#", "*" or a combination of them within 32 characters.

   The following are example configurations in the configuration file for IP phones:

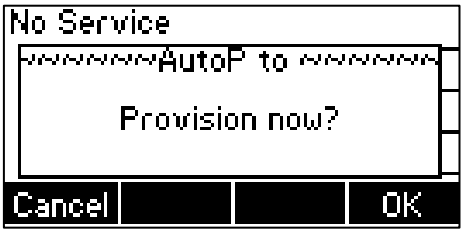
   static.autoprovision.1.code = *123

   static.autoprovision.1.url = http://192.168.1.30/T23G_1/

   static.autoprovision.2.code = *456

   static.autoprovision.2.url = http://192.168.1.30/T23G_2/
5. Send the specified activation code, associated user name and password to each end-user.
6. The user can set up the IP phone, and then input the activation code (e.g., *123) after the phone startup.

   The LCD screen will prompt the following dialog box:



7. Press the **OK** soft key to trigger the IP phone to perform auto provisioning.

   The LCD screen will prompt the following input box:

8. Enter the user name and password in the **User Name** and **Password** field respectively.

   The entered user name and password must correspond to the directory where the boot files and configuration files of the IP phone are located. If you enter invalid user name or password, the LCD screen will prompt the message "Wrong user name or password!". The prompt message will disappear in two seconds, and the LCD screen will return to the idle screen. You need to input the activation code again to trigger auto provisioning.

The IP phone downloads the specified configuration files in sequence in boot files from the provisioning server to complete phone configurations. For more information on boot files and configuration files, refer to Managing Boot Files and Managing Configuration Files.

The entered user name and password will be saved to the IP phone for next auto provisioning.

The LCD screen will not prompt for user name and password if the provisioning server does not require authentication, or the user name and password are already saved on the IP phone.

The following parameters are used to configure the auto provisioning via activation code method (X ranges from 1 to 50):

#(Optional.) Configure the code name for triggering auto provisioning.

static.autoprovision.X.name

#Configure the activation code.

static.autoprovision.X.code

#Configure the URL of the provisioning server.

static.autoprovision.X.url

#Configure the username and password for downloading boot files and configuration files. If configured, the LCD screen will not prompt for user name and password.

static.autoprovision.X.user

static.autoprovision.X.password

Downloading and Verifying Configurations

Downloading Boot, Configuration and Resource Files

After obtaining the provisioning server address in one of the ways introduced above, the phone will request to download the boot files and configuration files from the provisioning server when it is triggered to perform auto provisioning.

The IP phone will try to download the MAC-Oriented boot file firstly and then download the configuration files referenced in the MAC-Oriented boot file from the provisioning server during the auto provisioning. If no MAC-Oriented boot file is found, the IP phone will try to download the common boot file and then download the configuration files referenced in the common boot file. If no common boot file is found, the IP phone will try to download the Common CFG file firstly, and then try to download the MAC-Oriented CFG file from the provisioning server - that is, the old mechanism for auto provisioning.

For more information about auto provisioning, refer to Auto Provisioning Process.

If the access URLs of the resource files have been specified in the configuration files, the phone will try to download the resource files.

Resolving and Updating Configurations

After downloading, the phone resolves the configuration files and resource files (if specified in the configuration files), and then updates the configurations and resource files to the phone flash. Generally, updated configurations will automatically take effect after auto provisioning is completed. For update of some specific configurations which require a reboot before taking effect, for example, network configurations, the IP phone will reboot to make the configurations effective after auto provisioning is completed.

The IP phone calculates the MD5 values of the downloaded files before updating them. If the MD5 values of the Common and MAC-Oriented configuration files are the same as those of the last downloaded configuration files, this means these two configuration files on the provisioning server are not changed. The IP phone will complete the auto provisioning without repeated update. This is used to avoid unnecessary restart and the impact of phone use. On the contrary, the IP phone will update configurations.

The latest values to be applied to the IP phone are the values that take effect.

The phone only reboots when there is at least a specific configuration requiring a reboot after auto provisioning. If you want to force the IP phone to perform a reboot after auto provisioning, you can configure "static.auto_provision.reboot_force.enable = 1" in the configuration file.
For more information on the specific configurations which require a reboot during auto provisioning and the parameter "static.auto_provision.reboot_force.enable", refer to the latest IP Phones Description of Configuration Parameters in CFG Files for your phone on Yealink Technical Support.

If configuration files have been AES encrypted, the IP phone will use the Common AES key to decrypt the Common CFG file and the MAC-Oriented AES key to decrypt the <MAC>.cfg file after downloading the configuration files. For more information on how the IP phone decrypts configuration files, refer to *Yealink Configuration Encryption Tool User Guide*.

Using MAC-local CFG File

Uploading and downloading the <MAC>-local.cfg file

You can configure whether the IP phone uploads the <MAC>-local.cfg file to the provisioning server (or a specified URL configured by "static.auto_provision.custom.sync.path") once the file changes for backing up this file, and downloads the <MAC>-local.cfg file from the provisioning server (or a specified URL configured by "static.auto_provision.custom.sync.path") during auto provisioning to override the one stored on the phone. This process is controlled by the value of the parameter "static.auto_provision.custom.sync".

Updating configurations in the <MAC>-local.cfg file

You can configure whether the IP phone updates configurations in the <MAC>-local.cfg file during auto provisioning. This process is controlled by the value of the parameter "static.auto_provision.custom.protect". If the IP phone is configured to keep user's personalized settings (by setting the value of the parameter "static.auto_provision.custom.protect" to 1), it will update configurations in the <MAC>-local.cfg file. If the value of the parameter "overwrite_mode" is set to 1 in the boot file, the value of the parameter "static.auto_provision.custom.protect" will be forced to set to 1.

The IP phone updates configuration files during auto provisioning in sequence: CFG files referenced in the boot file>MAC-local CFG file (if no boot file is found, Common CFG file>MAC-Oriented CFG file>MAC-local CFG file). The configurations in the <MAC>-local.cfg file take precedence over the ones in other downloaded configuration files. As a result, the personalized settings of the phone configured via the phone or web user interface can be kept after auto provisioning.

Note that if the personalized settings are static settings, they cannot be kept after auto provisioning because the static settings will never be saved in the <MAC>-local.cfg file.

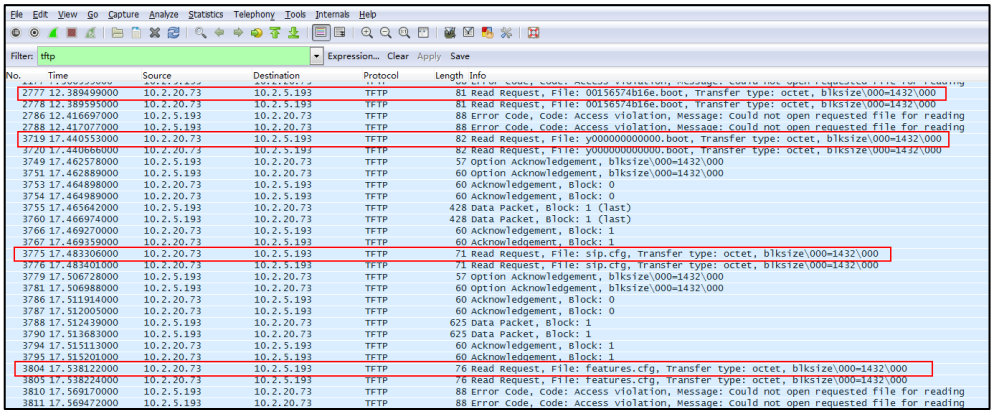
For more information, refer to the latest Administrator Guide for your phone on Yealink Technical Support.

Verifying Configurations

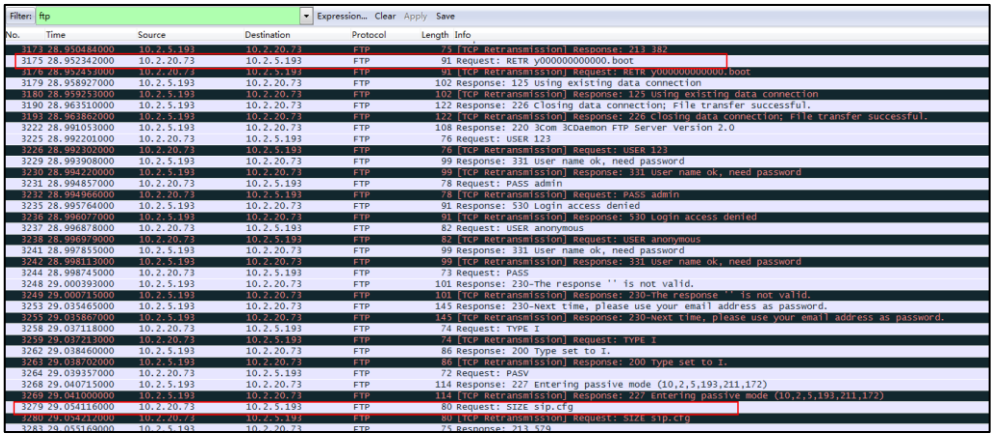
After auto provisioning, you can then verify the update via phone user interface or web user interface of the phone. For more information, refer to *Yealink phone-specific user guide*.

During auto provisioning, you can monitor the downloading requests and response messages by a WinPcap tool. The following shows some examples.

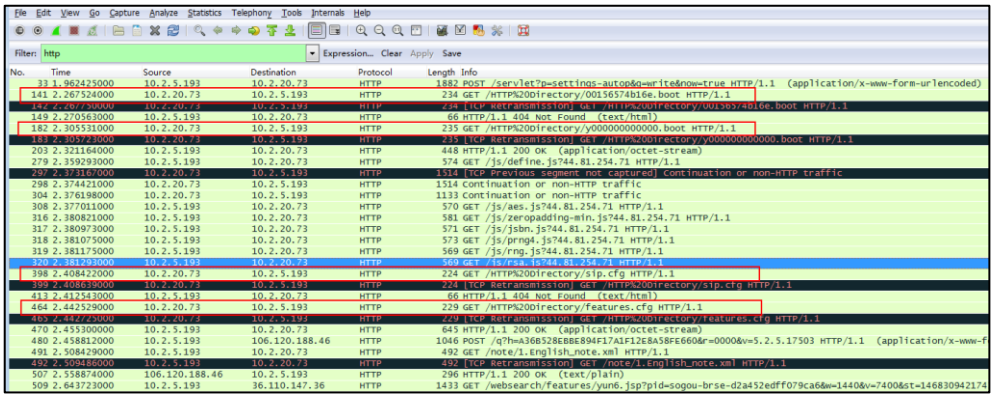
Example1: Yealink SIP-T23G IP phone downloads the boot file and configuration files from the TFTP server.



Example 2: Yealink SIP-T23G IP phone downloads the boot file and configuration files from the FTP server.



Example 3: Yealink SIP-T23G IP phone downloads boot file and configuration files from the HTTP server.



Troubleshooting

This chapter provides general troubleshooting information to help you solve problems you might encounter when deploying phones.

If you require additional information or assistance with the deployment, contact your system administrator.

Why does the IP phone fail to download configuration files?

- Ensure that auto provisioning feature is configured properly.
- Ensure that the provisioning server and network are reachable.
- Ensure that authentication credentials configured on the IP phone are correct.
- Ensure that configuration files exist on the provisioning server.
- Ensure that MAC-Oriented boot file and common boot file don't exist simultaneously on the provisioning server. If both exist, the IP phone only downloads MAC-Oriented boot file and the configuration files referenced in the MAC-Oriented boot file.

Why does the IP phone fail to authenticate the provisioning server during auto provisioning?

- Ensure that the certificate for the provisioning server has been uploaded to the phone's trusted certificates list. If not, do one of the following:
  - Import the certificate for the provisioning server to the phone's trusted certificates list (at phone's web path **Security**->**Trusted Certificates**->**Import Trusted Certificates**).
  - Disable the IP phone to only trust the server certificates in the trusted certificates list (at phone's web path **Security**->**Trusted Certificates**->**Only Accept Trusted Certificates**).

Why does the provisioning server return HTTP 404?

- Ensure that the provisioning server is properly set up.
- Ensure that the access URL is correct.
- Ensure that the requested files exist on the provisioning server.

Why does the IP phone display "Network unavailable"?

- Ensure that the Ethernet cable is plugged into the Internet port on the IP phone and the Ethernet cable is not loose.
- Ensure that the switch or hub in your network is operational.
- Ensure that the configurations of network are properly set in the configuration files.

Why is the permission denied when uploading files to the root directory of the FTP server?

- Ensure that the complete path to the root directory of the FTP server is authorized.
- Check security permissions on the root directory of the FTP server, if necessary, change the permissions.

Why doesn't the IP phone obtain the IP address from the DHCP server?

- Ensure that settings are correct on the DHCP server.
- Ensure that the IP phone is configured to obtain the IP address from the DHCP server.

Why doesn't the IP phone download the ring tone?

- Ensure that the file format of the ring tone is *.wav.
- Ensure that the size of the ring tone file is not larger than that the IP phone supports.
- Ensure that the properties of the ring tone for the IP phone are correct.
- Ensure that the network is available and the root directory is right for downloading.
- Ensure that the ring tone file exists on the provisioning server.

Why doesn't the IP phone update configurations?

- Ensure that the configuration files are different from the last ones.
- Ensure that the IP phone has downloaded the configuration files.
- Ensure that the parameters are correctly set in the configuration files.
- Ensure that the value of the parameter "static.auto_provision.custom.protect" is set to 0. If it is set to 1, the provisioning priority is shown as follows: phone/web user interface >central provisioning >factory defaults. A setting you make using a lower-priority method does not apply to or override a duplicate setting made using a higher-priority method.

  For more information, refer to the latest Administrator Guide for your phone on Yealink Technical Support.

Glossary

MAC Address: A Media Access Control address (MAC address) is a unique identifier assigned to network interfaces for communications on the physical network segment.

MD5: The MD5 Message-Digest Algorithm is a widely used as cryptographic hash function that produces a 128-bit (16-byte) hash value.

DHCP: Dynamic Host Configuration Protocol (DHCP) is a network configuration protocol for hosts on Internet Protocol (IP) networks. Computers that are connected to IP networks must be configured before they can communicate with other hosts.

FTP: File Transfer Protocol (FTP) is a standard network protocol used to transfer files from one host to another host over a TCP-based network, such as the Internet. It is often used to upload web pages and other documents from a private development machine to a public web-hosting server.

HTTP: The Hypertext Transfer Protocol (HTTP) is an application protocol for distributed, collaborative, hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web.

HTTPS: Hypertext Transfer Protocol Secure (HTTPS) is a combination of Hypertext Transfer Protocol (HTTP) with SSL/TLS protocol. It provides encrypted communication and secure identification of a network web server.

TFTP: Trivial File Transfer Protocol (TFTP) is a simple protocol to transfer files. It has been implemented on top of the User Datagram Protocol (UDP) using port number 69.

AES: Advanced Encryption Standard (AES) is a specification for the encryption of electronic data.

URL: A uniform resource locator or universal resource locator (URL) is a specific character string that constitutes a reference to an Internet resource.

XML: Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

Appendix

Configuring an FTP Server

Wftpd and FileZilla are free FTP application software for Windows. This section mainly provides instructions on how to configure an FTP server using wftpd for Windows. You can download wftpd online: http://www.wftpd.com/products/products.html or FileZilla online: https://filezilla-project.org.

We recommend that you use vsftpd as an FTP server for Linux platform if required.

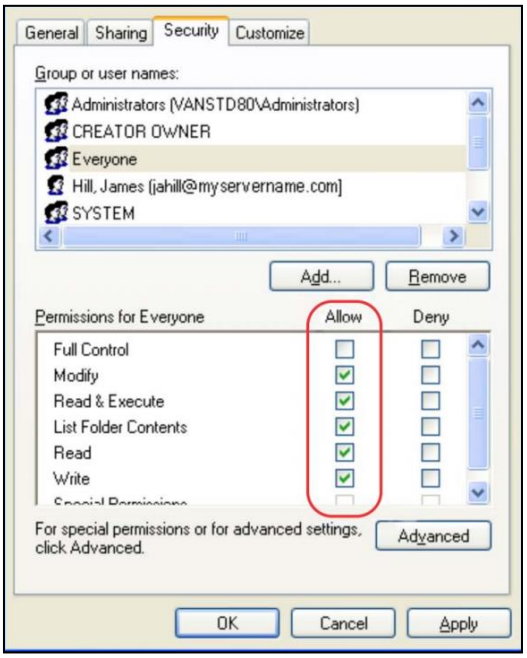
Preparing a Root Directory

To prepare a root directory:

1. Create an FTP root directory on the local system (e.g., D:\FTP Directory).
2. Place the boot files and configuration files to this root directory.
3. Set the security permissions for the FTP directory folder.

   You need to define a user or group name, and set the permissions: read, write, and modify. Security permissions vary by organizations.

   An example of configuration on the Windows platform is shown as below:

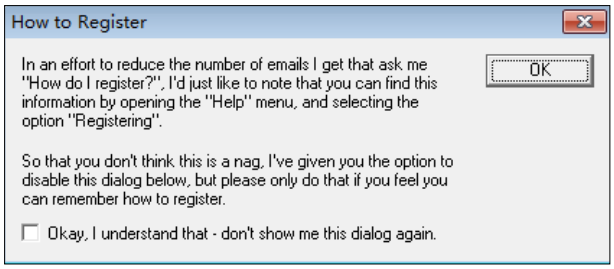


Configuring an FTP Server

To configure a wftpd server:

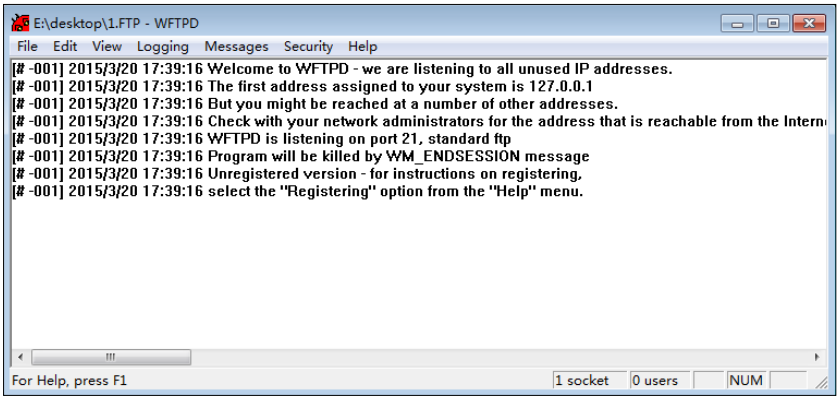
1. Download the compressed file of the wftpd application to your local directory and extract it.
2. Double click the **Wftpd.exe**.

   The dialogue box of how to register is shown as below:

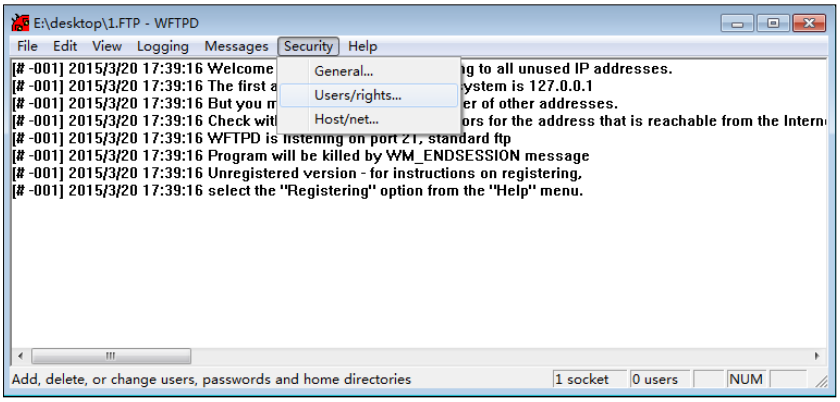


3. Check the check box and click **OK** in the pop-up box.

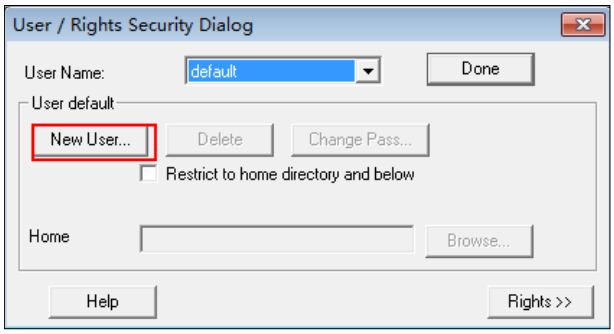
   The log file of the wftpd application is shown as below:



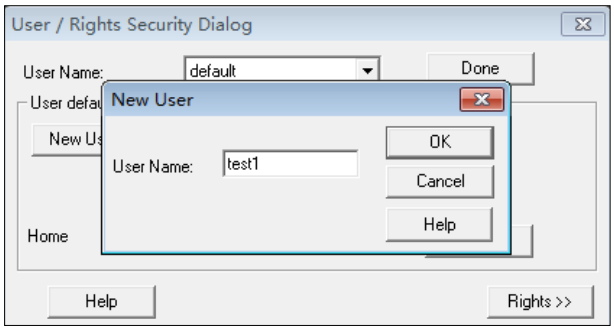
4. Click **Security**->**Users/rights**.



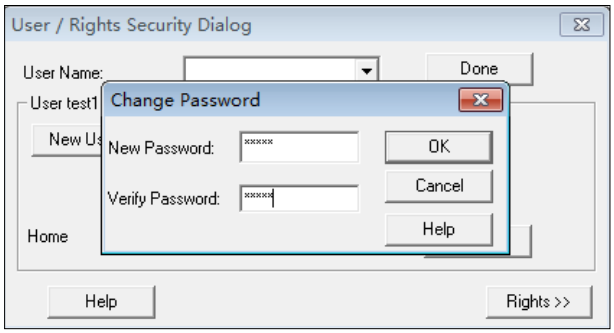
5. Click **New User**.



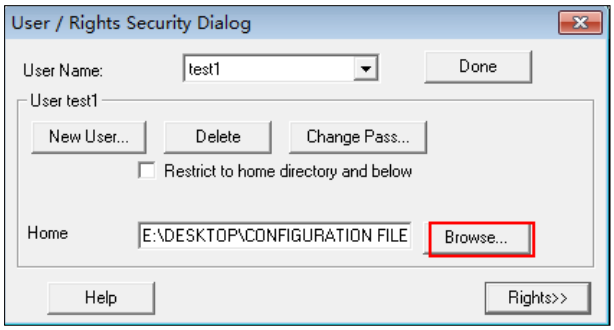
6. Enter a user name (e.g., test1) in the **User Name** field and then click **OK**.



7. Enter the password of the user (e.g., test1) created above in the **New Password** and **Verify Password** field respectively, and then click **OK**.

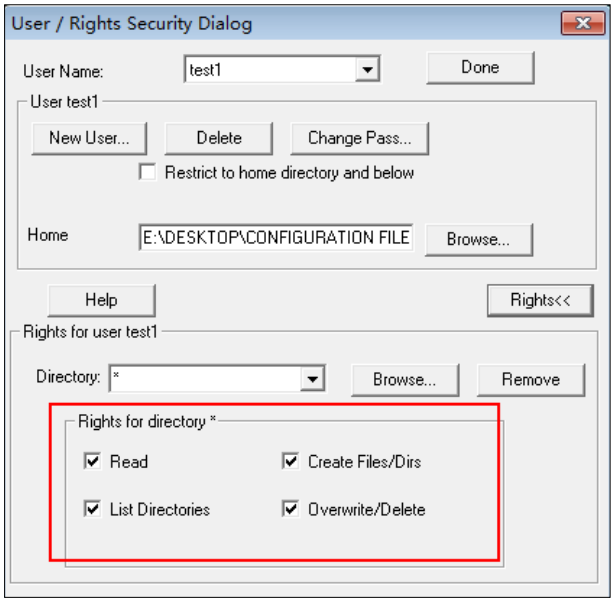


8. Click **Browse** to locate the FTP root directory in your local system.



9. Click **Rights>>** and assign the desired permission for the user (e.g., test1) created above.

10. Check the check boxes of **Read**, **Create Files/Dirs**, **List Directories** and **Overwrite/Delete** to make sure the FTP user has the read and write permission.



11. Click **Done** to save the settings and finish the configurations.

The server URL "ftp://username:password@IP/" (Here "IP" means the IP address of the provisioning server, "username" and "password" are the authentication for FTP download. For example, "ftp://test1:123456@10.3.6.234/") is where the IP phone downloads boot files and configuration files from.

Before configuring a wftpd server, ensure that no other FTP servers exist in your local system.

Configuring an HTTP Server

This section provides instructions on how to configure an HTTP server using HFS tool. You can download the HFS software online: http://www.snapfiles.com/get/hfs.html.

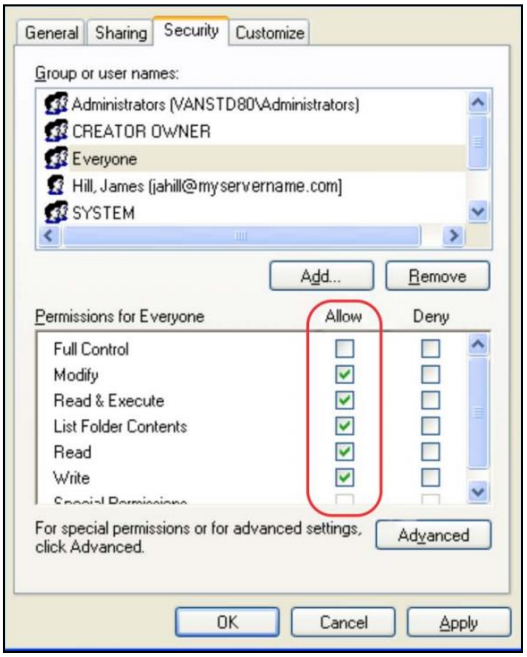
Preparing a Root Directory

To prepare a root directory:

1. Create an HTTP root directory on the local system (e.g., D:\HTTP Directory).
2. Place the boot files and configuration files to this root directory.
3. Set the security permissions for the HTTP directory folder.

You need to define a user or group name and set the permissions: read, write, and modify. Security permissions vary by organizations.

An example of configuration on the Windows platform is shown as below:



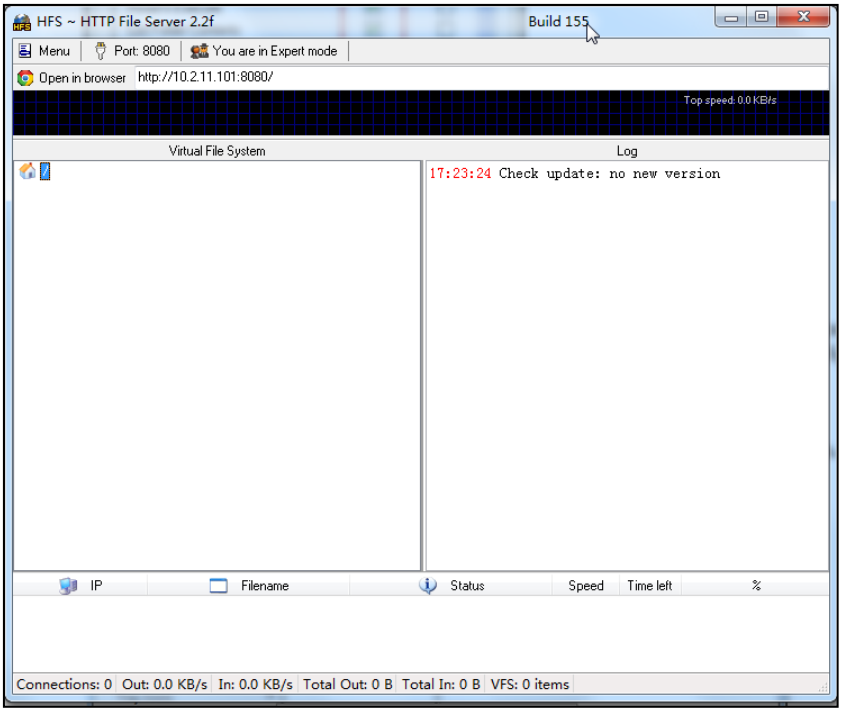
Configuring an HTTP Server

HFS tool is an executable application, so you don't need to install it.

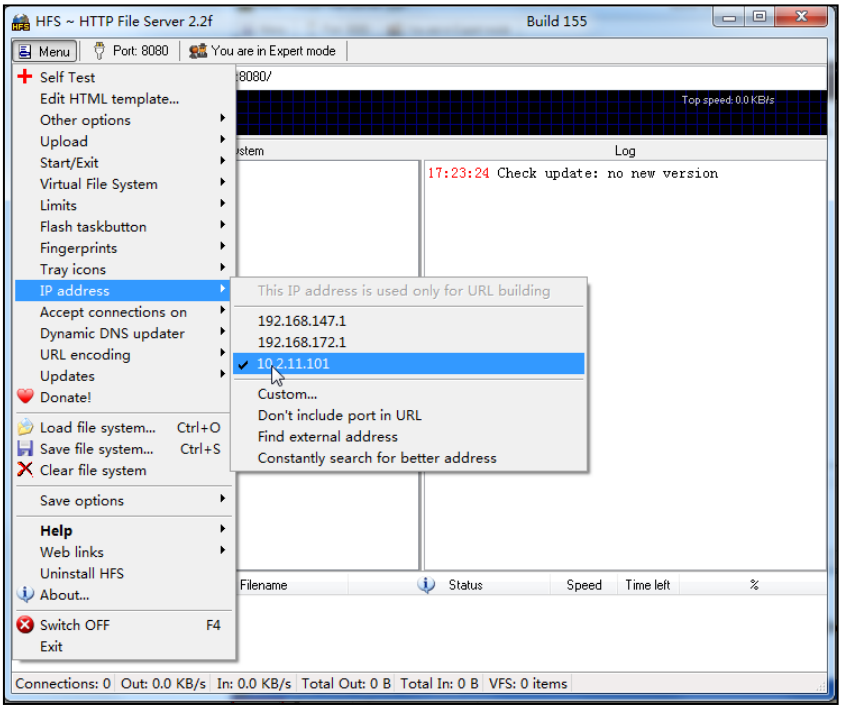
To configure an HTTP server:

1. Download the application file to your local directory, double click the **hfs.exe**.

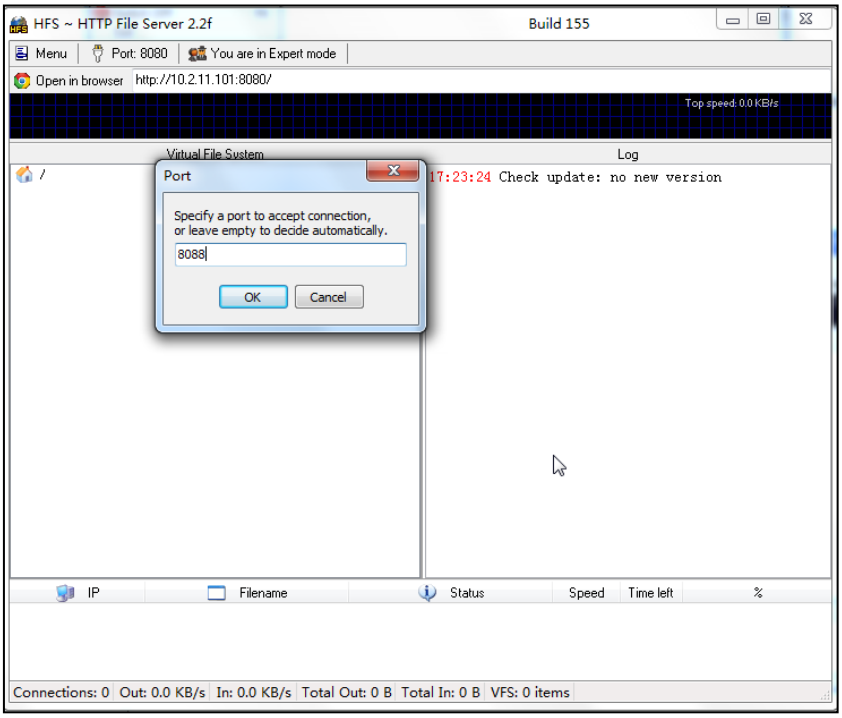
The main configuration page is shown as below:



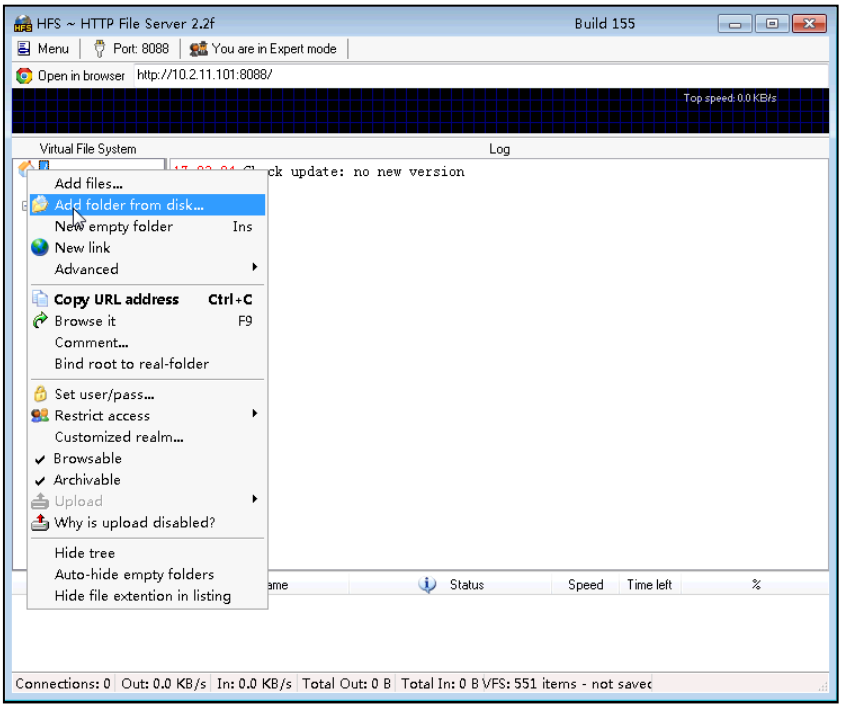
2. Click **Menu** in the main page and select the IP address of the PC from **IP address**.



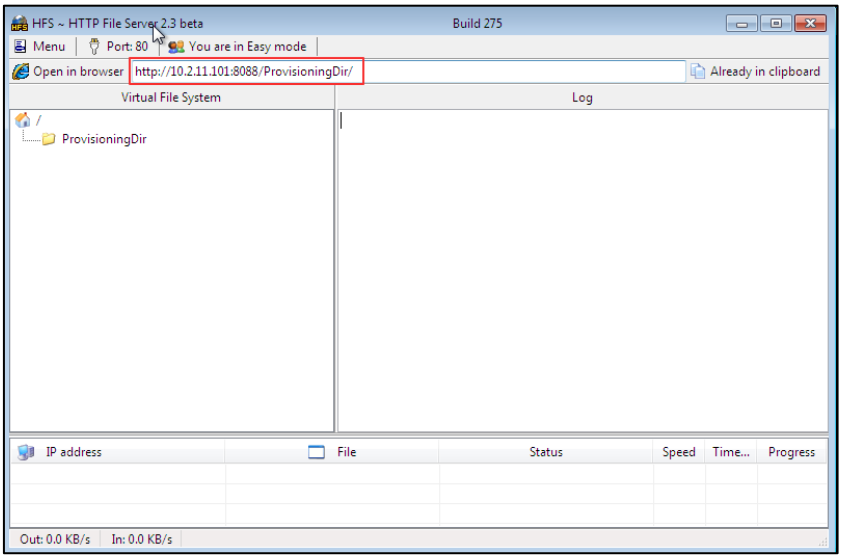
The default HTTP port is 8080. You can also reset the HTTP port (make sure there is no port conflict).



3. Right click the icon on the left of the main page, select **Add folder from disk** to add the HTTP Server root directory.



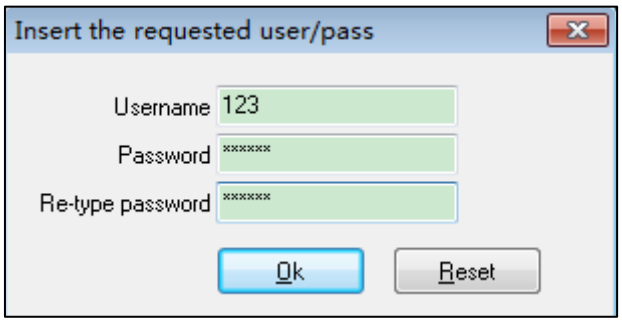
4. Locate the root directory from your local system.



5. Check the server URL (e.g., http://10.2.11.101:8088/ProvisioningDir) by clicking "**Open in browser**".

6. (Optional.) Right click the root directory name (e.g., ProvisioningDir), and then select **Set user/pass...**.

7. (Optional.) Enter the desired user name and password for the root directory in the corresponding fields and then click **OK**.



Yealink IP phones also support the Hypertext Transfer Protocol with SSL/TLS (HTTPS) protocol for auto provisioning. HTTPS protocol provides the encrypted communication and secure identification. For more information on installing and configuring an Apache HTTPS Server, refer to the network resource.