Yealink Configuration Encryption Tool User Guide

SOFTWARE LICENSE AGREEMENT

SOFTWARE LICENSE AGREEMENT FOR YEALINK CONFIGURATION CONVERSION TOOL IS IMPORTANT. PLEASE READ THIS LICENSE AGREEMENT CAREFULLY BEFORE CONTINUING WITH THIS PROGRAM: YEALINK NETWORK TECHNOLOGY CO., LTD Software License Agreement (SLA) is a legal agreement between you (either an individual or a single entity) and Yealink Network Technology CO., LTD. For the Yealink software product(s) identified above which may include associated software components, media, printed materials, and “online” or electronic documentation ("SOFTWARE PRODUCT"). By installing, copying, or otherwise using the SOFTWARE PRODUCT, you agree to be bound by the terms of this SLA. This license agreement represents the entire agreement concerning the program between you and Yealink Network Technology CO., LTD., (referred to as "licenser"), and it supersedes any prior proposal, representation, or understanding between the parties. If you do not agree to the terms of this SLA, do not install or use the SOFTWARE PRODUCT.
The SOFTWARE PRODUCT is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The SOFTWARE PRODUCT is licensed, not sold.

COPYRIGHT

WARRANTIES

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS GUIDE ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS GUIDE ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF PRODUCTS.
YEALINK NETWORK TECHNOLOGY CO., LTD. MAKES NO WARRANTY OF ANY KIND WITH

REGARD TO THIS GUIDE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Yealink Network Technology CO., LTD. shall not be liable for errors contained herein nor for incidental or consequential damages in connection with the furnishing, performance, or use of this guide.

Introduction

Configuration files contain sensitive information such as user accounts, login passwords or registration information. To protect sensitive information from tampering, you must encrypt configuration files. Yealink provides tools for encrypting configuration files on Windows and Linux platform respectively. You can ask the distributor or the Yealink Field Application Engineer for these tools, or you can download them from website at: http://www.yealink.com/DocumentDownload.aspx?CateId=142&flag=142.

This document provides detailed information on how to encrypt configuration files using Yealink-supplied encryption tools, and how to deploy Yealink IP phones using these encrypted configuration files. The information applies to Yealink IP phones running firmware version 71 or later.

The encryption tool encrypts plaintext <y0000000000xx>.cfg and <MAC>.cfg files (one by one or in batch) using 16-character symmetric keys (the same or different keys for configuration files) and generates encrypted configuration files with the same file name as before. This tool also encrypts the plaintext 16-character symmetric keys using a fixed key, which is the same as the one built in the IP phone, and generates new files named as <xx_Security>.enc (xx indicates the name of the configuration file, for example, y000000000000_Security.enc for y000000000000.cfg file). This tool generates another new file named as Aeskey.txt storing the plaintext 16-character symmetric keys for each configuration file.

For security, administrator should upload encrypted configuration files, <y0000000000xx_Security>.enc and/or <MAC_Security>.enc files to the root directory of the provisioning server. During auto provisioning, the IP phone requests to download <y0000000000xx>.cfg file first. If the downloaded configuration file is encrypted, the IP phone will request to download <y0000000000xx_Security>.enc file (if enabled) and decrypt <y0000000000xx>.cfg file into the plaintext key (e.g., key2) using the built-in key (e.g., key1). Then the IP phone decrypts <y0000000000xx>.cfg file using key2. After decryption, the IP phone resolves configuration files and updates configuration settings onto the IP phone system. The way the IP phone processes the <MAC>.cfg file is the same as the <y0000000000xx>.cfg file.

Configuration Encryption Tool on Windows Platform

This tool supports Microsoft Windows XP and Windows 7 (both 32-bit and 64-bit).

To encrypt configuration files:

1. Double click "Config_Encrypt_Tool.exe" to start the application tool.

   The screenshot of the main page is shown as below:

2. Click **Browse** to locate configuration file(s) (e.g., y000000000000.cfg) from your local system in the **Select File(s)** field.

   To select multiple configuration files, you can select the first file and then press and hold the **Ctrl** key and select the next files.

3. (Optional.) Click **Browse** to locate a target directory from your local system in the **Target Directory** field.

4. (Optional.) Mark the desired radio box in the **AES Model** field.

   If you mark the **Manual** radio box, you can enter an AES key in the **AES KEY** field or click **Re-Generate** to generate an AES key in the **AES KEY** field. The configuration file(s) will be encrypted using the AES key in the **AES KEY** field.

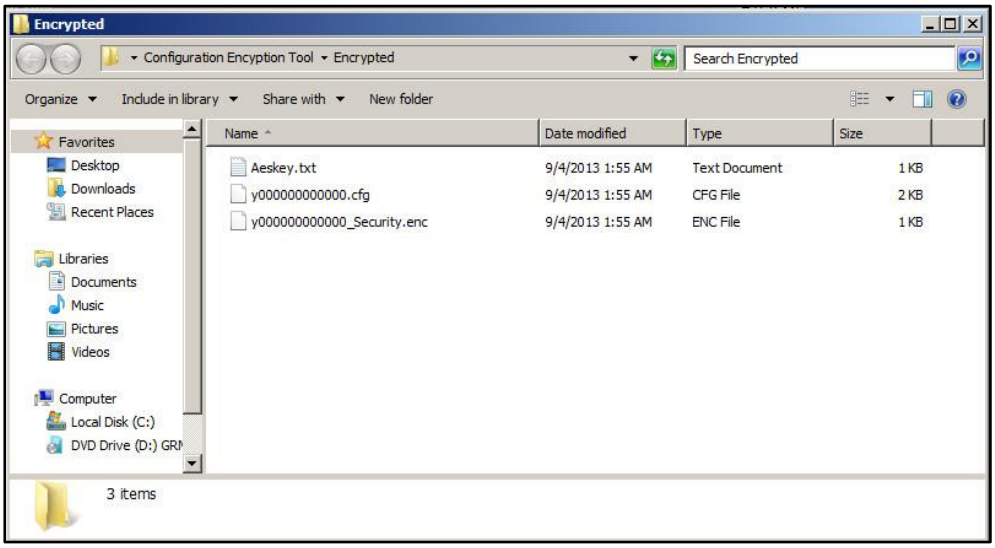
   If you mark the **Auto Generate** radio box, the configuration file(s) will be encrypted using random AES key. The AES keys of configuration files are different.

Note AES keys must be 16 characters and the supported characters contain: 0 ~ 9, A ~ Z, a ~ z.

5. Click **Encrypt** to encrypt the configuration file(s).

6. Click **OK**.

   The target directory will be automatically opened. You can find the encrypted CFG file(s), encrypted key file(s) and an Aeskey.txt file storing plaintext AES key(s).



Configuration Encryption Tool on Linux Platform

To encrypt the configuration files:

1. Place the encryption tool "yealinkencrypt" and configuration files in the same directory.
2. Open a terminal window.
3. Execute the **cd** command to locate the directory where the encryption tool is stored. For example, execute **cd /tmp** to locate the **/tmp** directory.

4. Execute one of the following commands according to your requirements:

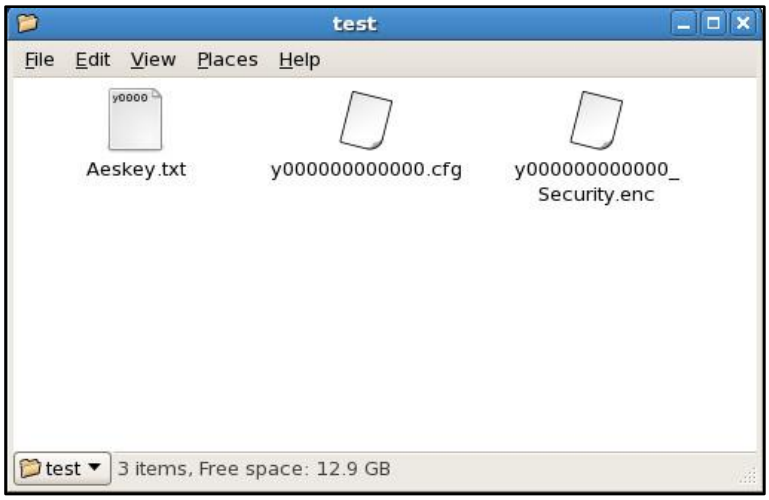
   - If you want to encrypt one or multiple specified configuration files, you need to execute the following command:

     ./yealinkencrypt -f file1.cfg [file2.cfg ...] [-p DESTPATH(Default as 'Encrypted')] [-k AESKEY(Default as random)]

     Example:

     ```
     [root@localhost tmp]#./yealinkencrypt -f y000000000000.cfg -p
     /home/test -k 0123456789123456
        AES Key: 0123456789123456
        Generate Security Key File...
        Generate Encrypt Config File...
        Write file to /home/test/Aeskey.txt!
        Write file to /home/test/y000000000000_Security.enc!
        Read file y000000000000.cfg!
        Write file to /home/test/y000000000000.cfg!
     ```

     This tool will encrypt the y000000000000.cfg file using the AES key 0123456789123456. You can find the encrypted y000000000000.cfg file, y000000000000_Security.enc file and an Aeskey.txt file storing the plaintext AES key 0123456789123456 in the specified directory.



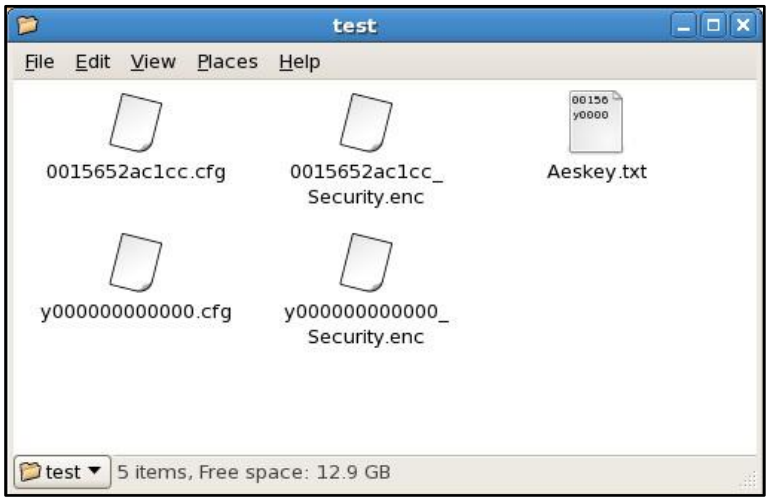
- If you want to encrypt configuration files in batch using a random AES key, you need to execute the following command:

  ./yealinkencrypt –f *.cfg [-p DESTPATH(Default as 'Encrypted')] -m

  Example:

```
[root@localhost tmp]#./yealinkencrypt -f *.cfg -p /home/test -m
   Generate AES Key...
   Write file to /home/test/Aeskey.txt!
   Write file to /home/test/0015652ac1cc_Security.enc!
   Read file 0015652ac1cc.cfg!
   Write file to /home/test/0015652ac1cc.cfg!
   Write file to /home/test/Aeskey.txt!
   Write file to /home/test/y000000000000_Security.enc!
   Read file y000000000000.cfg!
   Write file to /home/test/y000000000000.cfg!
```

  This tool will encrypt all CFG files using random AES keys (each CFG file corresponds to a random AES key). You can find the encrypted CFG files, encrypted key files and an Aeskey.txt file storing the plaintext AES keys in the specified directory.



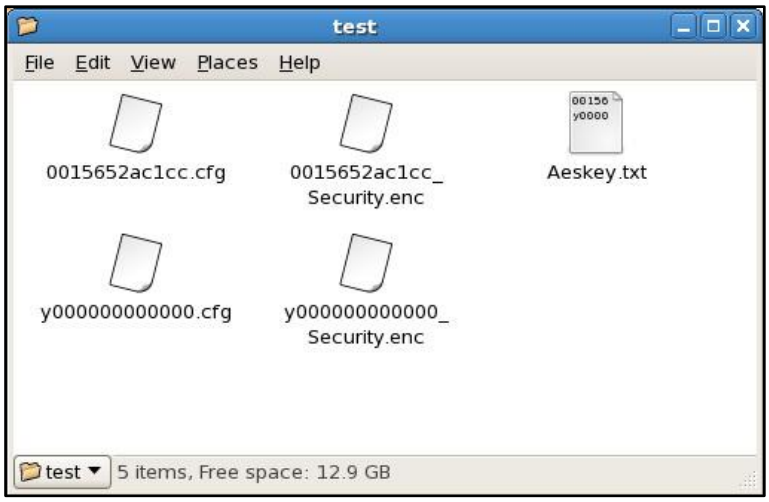
- If you want to encrypt configuration files in batch using a specified AES key, you need to execute the following command:

  ./yealinkencrypt –f *.cfg [-p DESTPATH(Default as 'Encrypted')] `-k 0123456789123456`

  Example:

```
[root@localhost tmp]#./yealinkencrypt -f *.cfg -p /home/test -k
0123456789123456

   AES Key: 0123456789123456

   Generate Security Key File...

   Generate Encrypt Config File...

   Write file to /home/test/Aeskey.txt!

   Write file to /home/test/0015652ac1cc_Security.enc!

   Read file 0015652ac1cc.cfg!

   Write file to /home/test/0015652ac1cc.cfg!

   Write file to /home/test/Aeskey.txt!

   Write file to /home/test/y000000000000_Security.enc!

   Read file y000000000000.cfg!

   Write file to /home/test/y000000000000.cfg!
```

  This tool will encrypt all CFG files using a specified AES key. You can find the encrypted CFG files, encrypted key files and an Aeskey.txt file storing the plaintext AES key in the specified directory.



Note AES keys must be 16 characters and the supported characters contain: 0 ~ 9, A ~ Z, a ~ z.

Configuring Yealink IP Phones

To ensure no plaintext configurations and keys are transmitted across the network, you need to configure the following parameters using the configuration file first.

Parameter	Description	Valid Value	Default Value
auto_provision.aes_key_in_file	Enable or disable the IP phone to download the encrypted key files (e.g., <y0000000000xx_Security>.enc, <MAC_Security>.enc) during auto provisioning. **0**-Disabled **1**-Enabled	Boolean	0
auto_provision.update_file_mode	Enable or disable the IP phone to update encrypted configuration settings only during auto provisioning. **0**-Disabled **1**-Enabled	Boolean	0

Deploying Yealink IP Phones Using Encrypted Configuration Files and AES keys

This section shows a scenario on how to deploy Yealink IP phones using encrypted configuration files and AES keys

Scenario: Encrypt configuration files and ensure no plaintext configurations and keys are transmitted across the network

Scenario Conditions:

- The administrator wants to encrypt configuration files to protect sensitive information in configuration files from tampering.
- SIP-T28 IP phone MAC: 0015651137F6.
- auto_provision.aes_key_in_file =1 (Enable the IP phone to download y000000000000_Security.enc and 0015651137F6_Security.enc files during auto provisioning)
- auto_provision.update_file_mode =1 (Enable the IP phone to update encrypted configuration settings only during auto provisioning)

- auto_provision.aes_key_16.com = 1234 (The parameter value can be set to an arbitrary value, but cannot be blank)
- auto_provision.aes_key_16.mac = 1234 (The parameter value can be set to an arbitrary value, but cannot be blank)

Scenario Operations:

1. The administrator encrypts y000000000000.cfg and 0015651137F6.cfg files and then uploads y000000000000_Security.enc, 0015651137F6_Security.enc, y000000000000.cfg (encrypted) and 0015651137F6.cfg (encrypted) files to the root directory of the provisioning server.

   For more information on encrypting configuration files, refer to Configuration Encryption Tool on Windows Platform on page 2 or Configuration Encryption Tool on Linux Platform on page 3.

2. Reboot the IP phone to trigger auto provisioning process. For more information, refer to Yealink IP Phones Auto Provisioning Guide.

During auto provisioning, the IP phone requests to download y000000000000.cfg file first. Because the downloaded configuration file is encrypted, the IP phone requests to download y000000000000_Security.enc file and then decrypts the file into the plaintext key (e.g., key2) using the built-in key (e.g., key1). The IP phone then decrypts the configuration file using the key2. After decryption, the IP phone resolves configuration files and updates configuration settings onto the IP phone system. If the downloaded configuration file is not encrypted, the IP phone will not request to download y000000000000_Security.enc file and update configuration settings in the configuration file onto the IP phone system.

The way the IP phones process the <MAC>.cfg file is the same as the <y000000000000>.cfg file.

For more information, refer to Appendix Auto Provisioning Flowchart on page 9.

Appendix Auto Provisioning Flowchart

The following shows auto provisioning flowchart for Yealink IP phones. The way the IP phones process the MAC-Oriented CFG file is the same as the Common CFG file.