Deploying and Managing the Applications on Yealink Phones

Introduction

Yealink T58A and VP59 phones offer access to a large amount of third-party Android applications. The third-party applications can be installed or uninstalled either one by one or in batch.

There are two versions of the application deployment and management function: special version and standard version. This guide provides detailed information on how to deploy and manage the applications in both two versions.

Special Version

In special version, the IP phone runs with highest privileges, you can deploy the applications directly without granting the appropriate permissions to the application. We recommend that you to use the special version with caution, because in this version the IP phone has all the permissions and there may be a security risk.

To deploy the third-party applications in special version using configuration files:

1. Add/Edit corresponding parameters in configuration files (for example, app.cfg).

    The following table lists the information of parameters:

Parameters	Permitted Values	Default
pm.version	**standard, special**	**standard**
Description: Configures the version of the application deployment and management function.		
app.install_url	**URL within 511 characters**	**Blank**
Description: Configures the access URL of the third-party application. **Example:** app.install_url = http://192.168.10.25/QQ.apk During the auto provisioning process, the IP phone connects to the HTTP provisioning server "192.168.10.25", and downloads the application "QQ.apk".		

Parameters	Permitted Values	Default
If you want to install/update multiple applications to the phone simultaneously, you can configure as following: app.install_url = http://192.168.10.25/QQ.apk app.install_url = http://192.168.10.25/Kugou.apk		
app.uninstall	**app name/app name.apk/package name**	**Blank**
Description: Uninstalls the specified third-party application. **Example:** To uninstall QQ, do one of the following: app.uninstall = QQ app.uninstall = QQ.apk app.uninstall = com.tencent.mobileqq If you want to uninstall multiple applications on the phone simultaneously, you can configure as following: app.uninstall = QQ app.uninstall = Kugou		

The following shows an example of QQ installation in configuration files:

pm.version = special

app.install_url = http://192.168.10.25/QQ.apk

The following shows an example of uninstalling QQ in configuration files:

pm.version = special

app.uninstall = QQ

2. Reference the configuration file in the boot file (for example, y000000000000.boot).

    Example:

    include:config "http://192.168.10.25/app.cfg"

3. Upload the boot file and configuration file to the root directory of the provisioning server.

4. Trigger IP phones to perform an auto provisioning for configuration update.

    For more information on auto provisioning, refer to the latest Auto Provisioning Guide on Yealink Technical Support.

Standard Version

In standard version, the IP phone does not have all the permissions, only the administrator can deploy and manage applications using configuration files. There are a series of parameters for administrator

to control the applications and grant permissions to applications.

After installing the application, you may need to grant the appropriate permissions to the application to use the application.

Deploying the Third-party Applications

To deploy the third-party applications in standard version using configuration files:

1. Add/Edit corresponding parameters in configuration files (for example, app.cfg).

   The following table lists the information of parameters:

Parameters	Permitted Values	Default
pm.version	**standard, special**	**standard**
Description: Configures the version of the application deployment and management function.		
app.install_url	**URL within 511 characters**	**Blank**
Description: Configures the access URL of the third-party application. **Example:** app.install_url = http://192.168.10.25/QQ.apk During the auto provisioning process, the IP phone connects to the HTTP provisioning server "192.168.10.25", and downloads the application "QQ.apk". If you want to install/update multiple applications to the phone simultaneously, you can configure as following: app.install_url = http://192.168.10.25/QQ.apk app.install_url = http://192.168.10.25/Kugou.apk		
app.uninstall	**app name/app name.apk/package name**	**Blank**
Description: Uninstalls the specified third-party application. **Example:** To uninstall QQ, do one of the following: app.uninstall = QQ app.uninstall = QQ.apk		

Parameters	Permitted Values	Default
app.uninstall = com.tencent.mobileqq If you want to uninstall multiple applications on the phone simultaneously, you can configure as following: app.uninstall = http://192.168.10.25/QQ.apk app.uninstall = http://192.168.10.25/Kugou.apk		

The following shows an example of QQ installation in configuration files:

pm.version = standard

app.install_url = http://192.168.10.25/QQ.apk

The following shows an example of uninstalling QQ in configuration files:

pm.version = standard

app.uninstall = QQ

2. Reference the configuration file in the boot file (for example, y000000000000.boot).

Example:

include:config "http://192.168.10.25/app.cfg"

3. Upload the boot file and configuration file to the root directory of the provisioning server.

4. Trigger IP phones to perform an auto provisioning for configuration update.

For more information on auto provisioning, refer to the latest Auto Provisioning Guide on Yealink Technical Support.

You can also control the application to run after startup, run in the background, run during the call and so on. For more information, refer to Application Control on page 4.

Managing the Applications

Application Control

To control the applications using configuration files:

1. Add/Edit corresponding parameters in configuration files (for example, app.cfg).

The following table lists the information of parameters:

Parameters	Permitted Values	Default
app.unavailable	**app name/app name.apk/package name**	**none**
Description:		

Parameters	Permitted Values	Default
Hides the system applications on the IP phone. **Example**: app.unavailable = Settings If you want to hide multiple applications on the phone simultaneously, you can configure as following (between two applications are separated by "/"): app.unavailable = Settings/FileManager/Message If you want to cancel the operations for applications on the phone, you can configure as following: app.unavailable = none **Note:** It works only if the value of the parameter "pm.version" is set to standard. It is only applicable to the system applications.		
app.autorun	**app name/app name.apk/package name**	**none**
Description: Allows the application to run automatically after the phone starts up. For more information on how to configure it, refer to the parameter "app.unavailable". **Note:** It works only if the value of the parameter "pm.version" is set to standard.		
app.prohibit_bgrun	**app name/app name.apk/package name**	**none**
Description: Prohibits the application to run in the background. For more information on how to configure it, refer to the parameter "app.unavailable". **Note**: It works only if the value of the parameter "pm.version" is set to standard.		
app.prohibit_callrun	**app name/app name.apk/package name**	**none**
Description: Prohibits the application to run during the call. For more information on how to configure it, refer to the parameter "app.unavailable". **Note**: It works only if the value of the parameter "pm.version" is set to standard.		
app.prohibit_uninstall	**app name/app name.apk/package name**	**none**

Parameters	Permitted Values	Default
Description: Prohibits the users to uninstall the third-party application via phone user interface. For more information on how to configure it, refer to the parameter "app.unavailable". **Note**: It works only if the value of the parameter "pm.version" is set to standard. It is only applicable to the third-party applications.		

The following shows an example of QQ installation in configuration files:

app.unavailable = Settings

app.autorun = QQ

app.prohibit_bgrun = Recorder

app.prohibit_callrun = Browser

app.prohibit_uninstall = QQ

2. Reference the configuration file in the boot file (for example, y000000000000.boot).

Example:

include:config "http://192.168.10.25/app.cfg"

3. Upload the boot file and configuration file to the root directory of the provisioning server.

4. Trigger IP phones to perform an auto provisioning for configuration update.

For more information on auto provisioning, refer to the latest Auto Provisioning Guide on Yealink Technical Support.