PTT Pro for Android

Version 3.3.10314

Workcloud Communication

Programmer Guide

Terms of Use

Proprietary Statement

This manual contains proprietary information of Zebra Technologies Corporation and its subsidiaries ("Zebra Technologies"). It is intended solely for the information and use of parties operating and maintaining the equipment described herein. Such proprietary information may not be used, reproduced, or disclosed to any other parties for any other purpose without the express, written permission of Zebra Technologies.

Product Improvements

Continuous improvement of products is a policy of Zebra Technologies. All specifications and designs are subject to change without notice.

Liability Disclaimer

Zebra Technologies takes steps to ensure that its published Engineering specifications and manuals are correct; however, errors do occur. Zebra Technologies reserves the right to correct any such errors and disclaims liability resulting therefrom.

Limitation of Liability

In no event shall Zebra Technologies or anyone else involved in the creation, production, or delivery of the accompanying product (including hardware and software) be liable for any damages whatsoever (including, without limitation, consequential damages including loss of business profits, business interruption, or loss of business information) arising out of the use of, the results of use of, or inability to use such product, even if Zebra Technologies has been advised of the possibility of such damages. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

Contents

Contents

About this Guide

This guide provides application programming interface (API) information for writing applications that use PTT Pro for Android. The guide assumes that the reader is familiar with PTT Pro for Android as well as Android intents.

The API consists of a series of intents that can be invoked through the following methods:

StageNow
An Android intent can be generated using StageNow as described at techdocs.zebra.com/mx/intent/. The intent can be sent immediately, scheduled, or triggered off by pressing a button or sensor.

MDM
An Android intent can be generated from many MDMs. Examples include:

> **Soti Mobicontrol**
> soti.net/mc/help/v14.4/en/scriptcmds/reference/sendintent.html

> **Airwatch/Workspace ONE**
> techzone.vmware.com/resource/workspace-one

Android application
An Android application can generate an intent to another application on the device. For more information, refer to developer.android.com/guide/components/intents-filters .

When PTT Pro for Android receives the intent, it takes the action indicated by the intent. The available intents are detailed in the following sections and include initiating PTT Pro calls, sending messages, signing in/out of the application, and configuring the client.

Chapter Descriptions

The chapters in this guide include:

- About this Guide provides an explanation about the document conventions and related documentation.

- Intents and Actions to Trigger Calls and Messages provides information about starting calls, ending calls, and sending messages.

- Intents and Actions to Sign in, Sign-out, and Manage Data Disclosure provides information about sign-in, sign-out, and manage the PTT Pro for Android application.

- Intents and Actions for Configuration provides information about configuring the PTT Pro for Android application.

- Intents and Actions for Status Information provides information about determining the status of PTT Pro for Android.

- [PTT Pro Sample Project](#) provides information about initiating sample PTT Pro calls and sending PTT Pro messages.

Document Conventions

Intents for PTT Pro for Android may include extras to pass data. Intents may also include a URI with query parameters.

Extras
An intent may have extras, a standard part of Android intents. These are indicated in the following tables by Extra 0, Extra 1, etc., followed by the type, name, and acceptable values.

URI Parameters
Several intents include a URI that may include query parameters. These are indicated in the following tables by `URI parameter 0`, `URI parameter 1`, etc., followed by the name and acceptable values

Notational Conventions

The following notational conventions make the content of this document easy to navigate.

- **Bold** text is used to highlight the following:
    - Dialog box, window, and screen names
    - Dropdown list and list box names
    - Checkbox and radio button names
    - Icons on a screen
    - Key names on a keypad
    - Button names on a screen
- Bullets (•) indicate:
    - Action items
    - List of alternatives
    - Lists of required steps that are not necessarily sequential
- Sequential lists (for example, those that describe step-by-step procedures) appear as numbered lists.

Service Information

If you have a problem with your equipment, contact Zebra Global Customer Support for your region. Contact information is available at: zebra.com/support.

When contacting support, please have the following information available:

- Serial number of the unit
- Model number or product name
- Software/firmware type and version number

Zebra responds to calls by email, telephone, or fax within the time limits set forth in support agreements.

If your problem cannot be solved by Zebra Customer Support, you may need to return your equipment for servicing and will be given specific directions. Zebra is not responsible for any damages incurred during

shipment if the approved shipping container is not used. Shipping the units improperly can possibly void the warranty.

If you purchased your Zebra business product from a Zebra business partner, contact that business partner for support.

Revision History

Revision	Date	Description
MN-004325-01EN Rev A	10/2021	Initial release.
MN-004325-02EN Rev A	9/2022	Removed Signout and added resertConfig parameter.
MN-004325-03EN Rev A	9/2022	Added IDP Solutions for PTT Pro.
MN-004325-04EN Rev A	7/2023	Added intents for status information.
MN-004325-05EN Rev A	11/2023	Added intent for setting DnD.
MN-004325-06EN Rev A	1/2024	Added note for configuring third-party devices with PVM and using an intent to initiate a one-to-one call in PTT Lite.
MN-004325-07EN Rev A	4/2024	Rebranded to Workcloud Communication.
MN-004325-08EN Rev A	5/2024	Added intent for call status.
MN-004325-09EN Rev A	8/2024	Added intent for an alert call.
MN-004325-10EN Rev A	12/2024	Added intent for Locate the User.
MN-004325-11EN Rev A	03/2025	Updated intents for Locate the User, initiate a PTT Call, and Retrieve LAS User List Using Intent.

Intents and Actions to Initiate Calls and Messages

You can use intents to initiate and end PTT calls, sending duress alerts, and sending messages.

Initiate a PTT Call

Use this intent to initiate a PTT call to a group or to one or more users. When multiple users are included, an ad hoc group call is initiated. You can also use this intent to initiate a one-to-one call in PTT Pro.

Prerequisites

- When initiating a call to a group, the group must be defined in the PTT Pro server and the user who initiates the PTT call (i.e., the target of the intent) must be a member of the group.
- When initiating a call to one or more users, the usernames(s) must be defined in the PTT Pro server.
- The minimum required PTT Pro Android version is 3.3.10134.

Intent Definition

User or group information with the intent is not mandatory, in such case one of the following-mentioned actions take place and supports from PTT Pro Android version is 3.3.10311 and later.

- Initiate a PTT Pro call to the selected user or group displayed in the PTT Pro UI if no specific user or group is mentioned in the PTT Pro Call intent.
- Initiate a call to the Default Callee when the device is in the background, the Default Callee is configured, and no specific user or group is designated in the PTT Pro Call intent.

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_PTT_PRO_CALL`
Intent Type	broadcast
Extra 0	This extra defines whether the call is targeted to a group or users. This is a mandatory parameter.
Type	Integer
Name	`recipientType`
Value	1 to target the call to a group. 0 to target the call to 1 or more users.
Extra 1	This extra defines the recipient(s) of the call. This is a mandatory parameter.

8

Name	Description
Type	String
Name	`recipientName`
Value	Use the group name or the user name(s) exactly as they appear in the PTT Pro Server.

ADB Examples

Call a group:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_CALL --ei
 recipientType 1 --es recipientName 'staff_admin'
```

Call a user:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_CALL --ei
 recipientType 0 --es recipientName 'george.shaw'
```

Call multiple users:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_CALL --ei
 recipientType 0 --es recipientName 'george.shaw,ira.gershwin,cole.porter'
```

End a PTT Call

This intent is used to end a PTT call.

Prerequisites

- The minimum required PTT Pro for Android version is 3.3.10141.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_PTT_PRO_CALL_END`
Intent Type	broadcast

ADB Examples

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_CALL_END
```

NOTE: This intent ends whatever call is currently in progress in the PTT Pro for Android.

Initiating and Ending a PTT Emergency call

This intent is used to initiate a PTT emergency call or to end an emergency call which is in progress. No extra parameters are supported with this intent. To initiate an emergency call, the emergency recipient

must be specified in the Workcloud Communication PTT Pro Management Portal by editing the user, selecting the Client Settings, and specifying the Emergency Group.

If this is intent is sent during an ongoing emergency call, it ends the emergency call.

Prerequisites

- When initiating an emergency call, the Client Settings for the initiating user must be set to the desired Emergency Group.
- The group must be defined in the Workcloud Communication PTT Pro Management Portal and the initiating PTT Pro user (i.e., the target of the intent) must be a member of the group.
- The minimum required PTT Pro Android version is 3.3.10150

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_PTT_PRO_EMERGENCY_CALL`
Intent Type	broadcast

ADB Example

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_EMERGENCY_CALL
```

NOTE:

- The emergency call is started or ended immediately upon receipt of the intent by PTT Pro for Android.
- If a PTT call is active when this intent is sent, PTT Pro for Android ends the active PTT call before initiating the emergency call.
- When ending the emergency call through this intent, the emergency call is ended, but the PTT call is not. The user needs to manually to end the PTT call to exit from the current PTT call session.

Worker Duress Alert

This intent is used to initiate a Worker Duress Alert, which sends an alert to the members of a PTT Pro group. The alert recipients can respond with a PTT call to the initiator to validate duress and obtain additional information.

PTT Pro for Android must be configured appropriately to enable the Worker Duress feature. The Workcloud Communication PTT Pro for Android User Guide at zebra.com/us/en/support-downloads/software/productivity-apps/push-to-talk-pro.html includes the configuration details for the PTT Pro client, the configuration of a button to send this intent, and the operation of the feature in the PTT Pro client.

Prerequisites

- The PTT Pro client must be configured as described in the PTT Pro User Guide. The intents for configuration in this guide can be used to communicate the configuration related information to the PTT Pro client.
- The recipient group which is configured in the PTT Pro client must be defined in the PTT Pro server.

- Once Worker Duress is enabled through configuration, the user is asked to allow access to the Android Do Not Disturb feature on Android 10 and later. The user must select **Allow**.

- The minimum required PTT Pro Android version is 3.3.10150.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.FREE_FALL_STATE_MANUALLY_TRIGGERED`
Intent Type	broadcast
Extra 0	Set this extra to 1/true.
Type	Boolean
Name	`FreeFallState`
Value	For Android 10 and below, set to "1" For Android 11 and above, set to "true"

Send a PTT Message

This intent is used to send a PTT message to a group or to one or more users.

Prerequisites

- When initiating a message to a group, the group must be defined in the PTT Pro Server, and the initiating PTT Pro user (i.e., the target of the intent) must be a member of the group.

- When initiating a call to one or more users, the usernames(s) must be defined in the PTT Pro Server.

- The minimum required PTT Pro for Android version is 3.3.10134.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_PTT_PRO_MESSAGE`
Intent Type	broadcast
Extra 0	This extra defines whether the call is targeted to a group or a user. This is a mandatory parameter
Type	Integer
Name	`recipientType`
Value	`1` to target the call to a group. `0` to target the call to 1 or more users.
Extra 1	
Type	String
Name	message

Name	Description
Value	Text string containing the message.

ADB Examples

Sending to a group:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_MESSAGE --ei
 recipientType 1 --es recipientName staff_admin' --es message 'Team meeting
 at 4pm'
```

Sending to a user:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_MESSAGE --
ei recipientType 0 --es recipientName 'george.shaw' --es message 'Review
 tomorrow is cancelled'
```

Sending to multiple users:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_MESSAGE --ei
 recipientType 0 --es recipientName 'george.shaw,ira.gershwin,cole.porter' --
es message 'Review tomorrow is cancelled'
```

Launch a PTT Pro Messaging Window

This intent is used to pop-up the PTT Pro for Android messaging dialog. The intent can optionally include a group or one or more users. If a group or users are not included, the dialog opens with no recipients, and the user must specify them.

Prerequisites

- When initiating a message to a group, the group must be defined in the PTT Pro Server, and the initiating PTT Pro user (i.e., the target of the intent) must be a member of the group.

- When initiating a call to one or more users, the usernames(s) must be defined in the PTT Pro Server.

- The minimum required PTT Pro for Android version is 3.2.10066.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_PTT_MESSAGE`
Intent Type	`broadcast`
Extra 0	This extra defines whether the call is targeted to a group or to users. This is an optional parameter, but it must be included if the `target` parameter is included.
Type	String
Name	`tab`

Name	Description
Value	g to target the call to a group c to target the call to 1 or more users
Extra 1	This extra defines the recipient(s) of the call. This is an optional parameter, but it must be included if the tab parameter is included.
Type	String
Name	`target`
Value	Use the Group name or Username(s) exactly as it appears in thePTT Pro Server.

NOTE: The following points are provided as notes on the target:

1. When sending to a group, only one group can be specified.

2. When sending to users, single or multiple users can be included. When including multiple user names, separate them with a comma.

3. Recipient names (user or group) are case-sensitive.

4. Two types of groups can be created by a PTT Pro user. When sending messages or initiating calls to these types of groups, the group name must be specified in the intent as a fully qualified group name. The two group types and the required formatting are defined below:

   • **Personal Groups**: Created by a user and only visible from the creator's Group List. Only the creator can initiate a call to a Personal Group. The fully qualified group name is specified as "<**group-name**>%<**creator-user-name**>", where **creator-user-name** is the PTT Pro user name of the creator.

   • **Member Groups**: Visible in all member's Group Lists. Any member of the group can initiate a call to the group. The fully qualified group name is specified as "<**group-name**>@<**creator-user-name**>", where **creator-user-name** is the PTT Pro user name of the creator.

ADB Examples

Sending to a group:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_MESSAGE --es "tab"
 "g" --es "target" "Hardware"
```

Sending to a user:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_MESSAGE –es "tab"
 "c" --es "target" "george.shaw"
```

Sending to multiple users:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_MESSAGE --ei "tab"
 "c" --es " target" "george.shaw,ira.gershwin,cole.porter"
```

Sending without specifying recipients:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_MESSAGE
```

Initiate a PTT Alert Call

Use this intent to trigger an alert call to an intended recipient.

Prerequisites

- The usernames must be defined in the PTT Pro server.

- The minimum required PTT Pro Android version is 3.3.10280.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_PTT_PRO_ALERT_CALL`
Intent Type	broadcast
Extra 0	Defines the call recipient. This is a mandatory parameter.
Type	String
Name	`recipientName`
Value	The user name as it appears in the PTT Pro Server (case sensitive).

ADB Examples

To trigger Alert Call

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_ALERT_CALL --
es recipientName 'john'
```

NOTE:

The call is not initiated if the intent has no `recipientName` and the following holds true:

- The app is not in the foreground

- The contacts tab is not selected

- The default call is not configured

Intents and Actions to Sign-in, Sign-out, and Manage Data Disclosure

You can use intents to sign in, sign out, and manage data disclosure for PTT Pro for Android.

Sign in to PTT Pro for Android

This intent is used to sign in to PTT Pro for Android. There are three different use cases for sign-in.

Sign in with ESN
The user is in a shared device environment, where devices are shared among different users. In a shared environment, the Electronic Serial Number (ESN) is automatically used as the activation code when the device is signed in to the PTT Pro Server.

Sign in with the Activation Code
If a user is not in a shared device environment, a user always uses the same device. In this case, each user is assigned a unique activation code. After the device is signed in with an activation code, the user and device are statically associated, unless reset at the portal.

Sign in with Username and Password
This intent is used to sign in to a user with a user name and password.

Prerequisites

- For a shared device environment:

    - The device's ESN must be configured in the PTT Pro Server.

    - The user must be configured in the PTT Pro Server.

    - OAuth authorization information must be configured in the PTT Pro Server.

- For a non-shared device environment:

    - The user must be configured in the PTT Pro server.

- The minimum required PTT Pro Android version is 3.0.10036.

Intent Definition

Name	Description
Action	`android.intent.action.VIEW with URI and package`
Intent Type	`startActivity`
URI	Shared device environment: pttpro://activatewithesn

Name	Description
	Non-shared device environment: pttpro://activate/?ac=
URI parameter 0 (optional)	This optional parameter can be used to prevent the display of **Data Consent** screen after installation.
Name	`showDisclosure`
Value	true or false
Package	`com.symbol.wfc.pttpro`

ADB Examples

Shared device environment:

```
adb shell am start -a android.intent.action.VIEW -d "pttpro://
activatewithesn/?showDisclosure=false" com.symbol.wfc.pttpro
```

Non-shared device environment:

```
adb shell am start -a android.intent.action.VIEW -d "pttpro://activate/?
ac=ABCDEFGHIJ\&showDisclosure=false" com.symbol.wfc.pttpro.
```

Sign in with a Username and Password

This intent is used to sign in a user to the PTT Pro for Android with a user name and password in a shared device environment.

There are two ways to sign in using this intent:

- Include `user_name` and `user_pwd` in the `login_info` JSON parameter or specify them as separate extra strings. This method of logging in requires a previous configuration of the key fields (For example; `key_user_name`, `key_user_pwd`) with provisioning configuration.

- Include `user_name` and `user_pwd` in the `login_info` JSON parameter.

Prerequisites

- The user must have signed out of the PTT Pro for Android.

- The minimum required PTT Pro for Android version is 3.3.10187.

- This intent is supported for Zebra devices using ESN OAuth login for a shared device environment.

  - The device's ESN must be configured in the PTT Pro Server.

  - The user must be configured in the PTT Pro Server.

  - OAuth authorization information must be configured in the PTT Pro Server.

- This intent is not applicable to PTT Pro for Android running in a Profile Manager environment.

- PTT Pro Data Consent page shall be approved or bypassed before issuing this intent. For more information, refer to the Bypass Data Consent.

- PTT Pro Client service must be already running.

- After signing out, it is recommended to wait 10 seconds or more before signing in again.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_LOGIN`
Intent Type	`broadcast`
Extra 0	This extra specifies the username. This is a mandatory parameter if the `login_info` parameter is not included.
Type	String
Name	`user_name`
Value	Username with domain (For example; `sample.user@domain`)
Extra 1	This extra specifies the user password. This is a mandatory parameter if the `login_info` parameter is not included.
Type	String
Name	`user_pwd`
Value	User Password
Extra 2	This extra specifies if the JSON login information is encrypted. This is an optional parameter.
Type	Boolean
Name	`json_encrypted`
Value	True indicates the login information is encrypted. False or absent indicates the information is not encrypted.
Extra 3	This extra `login_info` specifies the JSON login information. See below for the payload definition. This extra may or may not be encrypted.
Extra 3 - sent unencrypted	This extra specifies the JSON login information when not encrypted. This is an optional parameter.
Type	String
Name	`login_info`
Value	String in JSON format
Extra 3 – sent encrypted	This extra specifies the JSON login information when encrypted. This is an optional parameter.
Type	Byte array
Name	`login_info`
Value	See code snippet

NOTE: Extra values can be provided in any order.

ADB Examples

Sending user_name and user_pwd as separate extras:

```
adb shell am broadcast -a com.zebra.dfs.ACTION_LOGIN --es user_name --es
 user_pwd --es <other login parameters>
```

Sending login_info unencrypted:

```
adb shell am broadcast -a com.zebra.dfs.ACTION_LOGIN –es json_encrypted
 false--es login_info {user_name:sample.user@domain, user_pwd:<password>}
```

login_info payload

The `login_info` parameter contains JSON formatted data with the following fields:

```
{"user_name": "username",             //mandatory
"user_pwd": "userpwd",             //mandatory
"package": "<third_party_package_name>"      //optional;
}
```

PTT Pro for Android Configurations

The following optional parameters can be applied only once to the PTT Pro for Android. For more information about configuring the PTT Pro for Android, refer to the Load Configuration File.

Config String	Description	Type	Default
secret_key	<Optional> This value is used to decrypt the login_info when ACTION_LOGIN intent is sent with json_encrypted= true. key length must be 32 characters or UUID where PTT Pro for Android accepts 36 characters and remove the "-" character. For example: "5ecded3e-9562-11ea-bb37-0242ac130002" is converted as "5ecded3e956211eabb370242ac130002"	String	""
key_domain	<Optional> If this value is configured, the domain name is added to the "user_name". For example; testuser3 becomes testuser3@pttpro if key_domain is configured as "pttpro".	String	""
key_user_name	<Optional> Used with third-party launcher apps. This tag identifies the Username ID attribute and allows input on the current login web page. Sample HTML: """<input id=**userNameInput** name="UserName" type="email" class="text fullWidth" value="">"""	String	"userNameInput"
key_user_pwd	<Optional> Used with third-party launcher apps. This tag identifies the Password ID attribute and provides input on the current login web page. Sample HTML:"""<input id=**passwordInput** name="Password" type="password" class="text fullWidth" placeholder="Password" autocomplete="off">"""	String	"submitButton"
key_submit	<Optional> Used with third-party launcher apps. This tag identifies the Submit ID attribute for acceptance of credentials on the current login web page. Sample.html: """ <button id=**submitButton**	String	"submitButton"

Config String	Description	Type	Default
	type="submit" disabled="" tabindex="0" width="1" kind="primary">"""		

ADB Examples

There are two ways to log in to the PTT Pro for Android through intent using either ADB commands or an external application.

If "key_domain" is not configured.:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_LOGIN --es user_name
 'testuser3@pttpro' --es user_pwd 'pwdFor2022!' com.symbol.wfc.pttpro
```

If "key_domain" is configured with "pttpro":

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_LOGIN --es user_name
 'testuser3' --es user_pwd 'pwdFor2022!' com.symbol.wfc.pttpro
```

Sending login_info unencrypted:

If "key_domain" is not configured.

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_LOGIN --ez
 json_encrypted false --es login_info  '{"user_name":"testuser3@pttpro",
 "user_pwd":"pwdFor2022!"}' com.symbol.wfc.pttpro
```

If "key_domain" is configured with "pttpro"

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_LOGIN --ez
 json_encrypted false --es login_info  '{"user_name":"testuser3",
 "user_pwd":"pwdFor2022!"}' com.symbol.wfc.pttpro
```

Encyrption/Decryption

Encryption and decryption of the `login_info` parameter is accomplished using the Google Tink Library at //github.com/google/tink. Following is a sample code to accomplish encryption and decryption.

Code Snippet:

```
String aad = "";
String secret_key = "5ecded3e-9562-11ea-bb37-0242ac130002"; //Example: UUID

// Encryption
JSONObject data = new JSONObject();
data.put("user_name", "testuser3@pttpro");
data.put("user_pwd", "pwdFor2022!");
data.put("package", "com.example.launcherapp");
```

```
AesGcmJce agjEncryption = new AesGcmJce(secret_key.getBytes());
 byte[] encrypted = agjEncryption.encrypt(data.toString().getBytes(),
 aad.getBytes());

//send encryped data
Intent intent = new Intent("com.symbol.wfc.pttpro.ACTION_LOGIN");
intent.putExtra("login_info", encrypted);
intent.putExtra("json_encrypted", true);
intent.setPackage("com.symbol.wfc.pttpro");
sendBroadcast(intent);
```

The `secret_key` must be configured to encrypt/decrypt the `login_info`. If the PTT Pro Client does not have the `secret_key` or `json_encrypted=false`, it does not decrypt the information.

A sample source code project showcasing some of the APIs and configuration methods described in this document is available on request.

Bypass Data Consent

This intent is used to bypass the data consent screen.

Prerequisites

- The user must be signed in.

- The minimum required PTT Pro for Android version is 3.3.10092.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro/.MainActivity`
Intent Type	`startActivity`
Extra 0	This parameter can be used to disable or enable the display of the Data Consent Screen after installation. This parameter is optional.
Type	Boolean
Name	`showDisclosure`
Value	`True` Do not bypass the Data Consent screen. `False` Bypass the Data Consent screen.

ADB Example

```
adb shell am start -n com.symbol.wfc.pttpro/.MainActivity --ez
 "showDisclosure" false
```

NOTE: For more information on Data Consent, see General Rules Related to Data Consent.

General Rules Related to Data Consent

The Data Consent screen appears when PTT Pro for Android is first activated. The app will not function unless the consent is accepted or the screen is bypassed.

- After the Data Consent Screen is seen and accepted, it is not seen again unless the client is uninstalled and re-installed or the user data is cleared.

- When PTT Pro for Android is used in combination with the Profile Client, the Data Consent screen does not appear. The screen will not be seen unless the client is uninstalled and re-installed or the user data is cleared.

- If the Data Consent Screen is bypassed by the intents in this document, it will not be seen unless the client is uninstalled and re-installed or the user data is cleared.

- The PTT Pro for Android cannot be configured before accepting the Data Consent Screen unless the Data Consent Screen is bypassed.

Sign Out

These intents are used to sign out the user. Two intents are available.

- The `com.symbol.wfc.pttpro.ACTION_SIGN_OUT` action signs out of the application. The user can sign in again as described in the Workcloud Communication PTT Pro Installation Guide. If an activation code was previously used to sign in, that same activation code is already populated in the activation code field.

- The `com.symbol.wfc.pttpro.RESET_DEVICE_CONFIG` action signs out of the application and also clears the configuration. Unlike the `com.symbol.wfc.pttpro.ACTION_SIGN_OUT` action, the ESN or activation code configuration is also cleared. The Public key needs to be cleared for the device in the PTT Pro Management Portal after this command is applied to the client.

Prerequisites

- The user must be signed in.

- The minimum required PTT Pro for Android version is 3.3.10103.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_SIGN_OUT` to sign out `com.symbol.wfc.pttpro.RESET_DEVICE_CONFIG` to sign out and clear the configuration
Intent Type	broadcast
Extra 0	The pass key parameter provides an additional level of security to protect against misuse of these intents.
Type	String
Name	`pass_key`
Value	Contact Zebra support for your passkey1

Name	Description
Extra 1 (optional)	This extra specifies the restart of the PTT service. To allow the client to sign in again, this parameter needs to be set to true. The default value is false. The minimum required PTT Pro for Android version is 3.3.10186
Type	Boolean
Name	`restart_app`
Value	true or false

ADB Example

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_SIGN_OUT  --es
 "pass_key" "key-phrase"
```

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_SIGN_OUT  --es
 "pass_key" "pass_key" --ez "restart_app" true
```

NOTE:

- With the extra string `restart_app` is set to true, the PTT Pro for Android signs out, and the PTT Pro service is still running.

- Without the `restart_app` string, or if `restart_app` is set to false, the PTT Pro for Android signs out, and the service stops running and does not respond to commands. The only way to restart the PTT Pro service is to re-launch the application.

Intents and Actions for Configuration

Use these intents to configure PTT Pro for Android using a token from Workcloud Communication Provisioning Manager or by loading the configuration file.

Update with Token

This intent is used to pass a token to the Workcloud Communication Provisioning Manager and receive configuration information associated with the token in return. After the configuration is successfully applied, the application restarts. Refer to the Workcloud Communication PTT Pro for Android Configuration Guide for information on specific configuration parameters.

Prerequisites

- The Workcloud Communication Provisioning Manager must be set up with the token used in the intent. Refer to the Workcloud Communication Provisioning Manager Customer Administrator Guide for more information on tokens and associating configuration information with them. Refer to the Workcloud Communication PTT Pro for Android Configuration Guide for information on configuration parameters.

- The minimum required PTT Pro Android version is 3.3.10103.

Intent Definition

Name	Description
Action	`android.intent.action.VIEW` with URI and package
Intent Type	`startActivity`
URI	wfctp://<Token>
URI parameter 0	This parameter is optional and indicates if the existing configuration should be reset before applying the new configuration.
Name	`resetConfig`
Value	true or false
Package	`com.symbol.wfc.pttpro`

NOTE: PTT Pro for Android running on a third-party device cannot be configured with a Workcloud Communication Provisioning Manager token when used in a Workcloud Communication Profile Manager environment.

ADB Examples

- ```
  adb shell am start -a android.intent.action.VIEW -d "wfctp://<token>"
    com.symbol.wfc.pttpro
  ```

- ```
  adb shell am start -a android.intent.action.VIEW -d "wfctp://<token>?
  resetConfig=true" com.symbol.wfc.pttpro
  ```

NOTE:

Upon a fresh installation of the application, PTT Pro for Android connects to the Workcloud Communication Provisioning Manager, using its device ID as the token, and receives any configuration information associated with its device ID. This is done automatically after the data consent is approved or bypassed.

If PTT Pro for Android is in a call when the command is sent, the app waits until the call ends to apply the new configuration.

Load Configuration File

This intent is used to load a configuration file to PTT Pro for Android. After the configuration is successfully applied, the application restarts and the file is deleted. Refer to the Workcloud Communication PTT Pro for Android Configuration Guide guide for information on specific configuration parameters.

Prerequisites

- The minimum required PTT Pro for Android version is 3.3.10122.

- The configuration file must be loaded to the device.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_DEFAULT_CONFIG`
Intent Type	`startActivity`
Extra 0	This extra specifies the path to the JSON configuration that is loaded on the device. This is a mandatory parameter.
Type	String
Name	`configpath`
Value	Path to the JSON configuration file loaded in the device
Extra 1	This extra specifies whether the existing configuration should be reset or not before loading the new configuration. This parameter is optional; if not included, it defaults to false and the configuration is reset.
Type	Boolean
Name	`resetConfig`
Value	`true` Reset the current configuration before applying the new configuration.

Name	Description
	`false` Do not reset the current configuration before applying the new configuration.

ADB Example

```
adb shell am start -a com.symbol.wfc.pttpro.ACTION_DEFAULT_CONFIG --es
 "configpath" /enterprise/WFCPTTProDefault.json --ez "resetconfig" true
```

Enable or Disable Do not Disturb (DnD)

This intent is used to control the DnD configuration for PTT Pro for Android.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_DND`
Intent Type	broadcast
Extra	Sets the DnD status of the client.
Type	Boolean
Name	`enable`
Value	true—enables the DnD mode in PTT Pro for Android false—disables the DnD mode in PTT Pro for Android

ADB Examples

Enable the DnD mode for PTT Pro for Android:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_DND  --ez "enable"
 true
```

Disable the DnD mode for PTT Pro for Android:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_DND  --ez "disable"
 false
```

Retrieve LAS User List Using Intent

Intent Definition

The PTT Pro Client sends an INTENT that provides the list of users in a specific area (LAS Friendly Name) within a particular site.

The PTT Pro Client can now locate users within the same site or department via intent.

The minimum required PTT Pro for Android version is 3.3.10312 and later.

Name	Definition
Action	`com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE_LAS_FRIENDLY_NAME`
Intent Type	broadcast
Extra 0	Returns the user's LAS Information.
Type	String
Name	friendlyName
Value	• If the extra name is friendlyName • It is mandatory to provide a friendly name. • The friendly name is case-sensitive. • Use the name exactly as it is created on the LAS server.

NOTE:

• If the extra name is friendlyName and Users are configured with the specified friendlyName , the list of users belonging to that friendlyName will be shared.

• One must register the locate (com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE_LAS_FRIENDLY_NAME_STATUS) intent specified in the table below under the Register Intent section.

ADB Example:

```
adb shell am broadcast -a
 com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE_LAS_FRIENDLY_NAME --es
 friendlyName 'Reception'
```

Register Intent

The following information is about the Intent and the extra parameter that other applications need to register to get the result of the `ACTION_PTT_PRO_LOCATE_LAS_FRIENDLY_NAME` intent.

Name	Definition
Action	`com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE_LAS_FRIENDLY_NAME_STATUS`
Extra 0	Returns `friendly_name_status`
Type	Int
Name	friendly_name_status
Value	• 0 indicates a failure and checks friendly_name_error_reason extra for more information. • 1 Indicates response for a list of users belonging to that friendly name.
Extra 1	Returns `locate_result`
Type	String

Name	Definition
Name	friendly_name_result

Value	If `friendly_name_status` is 1: List of users belonging to that friendly name will be shared if `friendly_name_status` is returned as 1, as shown below
	
	This will return list of users belonging to that friendly name as an array of String.
	
	Example:
	
	``
	{
	[user1,user2,user3,user4,...]
	}
	``

Extra 2	Returns `friendly_name_error_reason`
Type	String
Name	friendly_name_error_reason
Value	If `friendly_name_status` is returned as 0, `friendly_name_error_reason` provides details about the failure. The reason can be one of the following: • Invalid friendly name • No users found under the specified friendly name. • Network is not available. • LAS is not configured. **NOTE:** The first two error messages are ended with the friendly name.
Extra 3	Returns `friendly_name`
Type	String
Name	friendly_name
Value	Returns the friendly LAS name passed as part of the `com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE_LAS_FRIENDLY_NAME` intent

Code Snippet

Send Broadcast

```
Intent intent = new Intent();
intent.setAction("com.symbol.wfc.pttpro.ACTION_PTT_
PRO_LOCATE_LAS_FRIENDLY_NAME");
   intent.putExtra("friendlyName","Reception");
   sendBroadcast(intent);
```

Register Intent

```
IntentFilter intent = new IntentFilter();
intent.addAction("com.symbol.wfc.pttpro.ACTION_PTT
_PRO_LOCATE_LAS_FRIENDLY_NAME_STATUS");
registerReceiver(mReceiver,intent);




public void onReceive(Context context, Intent intent) {
     String action= intent.getAction();
 if(action.equals("com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE
_LAS_FRIENDLY_NAME_STATUS")) {
             int status = intent.getIntExtra("friendly_name_status" , 0);
             String friendlyName = intent.getStringExtra("friendly_name");
              if (status == 0) {
              String error_reason =
 intent.getStringExtra("friendly_name_error_reason");
           Log.i(TAG, "friendlyName: "+friendlyName+"  error_reason : "+
 error_reason );
               }else if (status == 1){
String friendly_name_result =  intent.getStringExtra("friendly_name_result");
Log.i(TAG, "friendly_name_result : "+friendly_name_result +" friendlyName:
 "+friendly_name);
              }

     }
}
```

Intents and Actions for Status Information

The following intents provide status information for PTT Pro for Android.

Login Status

This intent is sent when a user logs into PTT Pro for Android.

Prerequisites

- The minimum required PTT Pro for Android version is 3.3.10223.

Intent Definition

Name	Definition
Action	`com.symbol.wfc.pttpro.LOGIN_STATUS`
Intent Type	broadcast
Extra	Returns information about the logged in user.
`userName`	String
`firstName`	String
`lastName`	String
`userId`	Integer
`departmentId`	Integer
`presenceStatusInteger`	Integer
`presenceStatusString`	String

If PTT Pro for Android receives a login status request, and the user is logged out, the client returns only `presenceStatusInteger` and `presenceStatusString`.

NOTE: When used with Profile Manager, PTT Pro for Android does not send Login status when the Profile Client is logged out or performing a role selection.

Code Example

```
val userName = intent.extras?.getString("userName")
val firstName = intent.extras?.getString("firstName")
val lastName = intent.extras?.getString("lastName")
```

```
val userId = intent.extras?.getInt("userId")
val departmentId = intent.extras?.getInt("departmentId")
val departmentName = intent.extras?.getString("departmentName")
val presentStr = intent.extras?.getString("presenceStatusString")
val presentInt = intent.extras?.getInt("presenceStatusInteger")
```

Logout Status

This intent is sent when a user logs out of PTT Pro for Android.

Prerequisites

- The minimum PTT Pro for Android version is 3.3.10223.

Intent Definition

Name	Definition
Action	`com.symbol.wfc.pttpro.LOGOUT_STATUS`
Intent Type	broadcast
Extra	Returns information about the logged out user.
userName	String
userID	Integer

Presence Status

This intent is sent when the presence status of a PTT Pro for Android user changes.

Prerequisites

- The minimum required PTT Pro for Android version is 3.3.10223.

Intent Definition

Name	Definition
Action	`wfc.pttpro.PRESENCE_STATUS`
Extra	
userName	String
presenceStatusInteger	Integer
presenceStatusString	String

Table 1   Presence State Values and Descriptions

presenceStatusInteger	presenceStatusString
0	UNAVAIALABLE
1	AVAILABLE
2	IN_CALL

Table 1   Presence State Values and Descriptions (Continued)

presenceStatusInteger	presenceStatusString
3	DND
4	NOT_RESPONDING
5	SILENT
6	URGET_COMM_ONLY
7	CIRCUIT_CALL
8	SIGNED_OUT

Conditions

- The client does not send an intent when Session State = Registering.

- If the client logs in and registers successfully, it sends AVAILABLE. Otherwise, the client sends UNAVAILABLE.

NOTE: When used with Profile Manager, PTT Pro for Android does not send Presence status when the Workcloud Communication Profile Client is logged out or performing a role selection.

Group List

PTT Pro for Android sends a broadcast with the following group details after the client successfully logs in.

Prerequisites

- The minimum required PTT Pro for Android version is 3.3.10223.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.GROUP_LIST`
Extra	
userName	String
status	Integer 0 = Success -1 = Error or NA
groupList	String Required to read this when status = 0.

There may be a delay of several seconds between Login Status and the Group List broadcast while PTT Pro for Android logs in and updates the group list. In addition, it may take up to four minutes for PTT Pro for Android to broadcast new group details when the user is added to a group on the PTT Pro Server .

Example

User is not logged in

```
INTENT_GROUPLIST userName:null status:-1 jsonObjString:null
groupListSize:0
```

User is logged in, but the user is not a member of a group

```
INTENT_GROUPLIST userName:Luke.Steber status:0 jsonObjString:
{"groupList":[]} groupListSize:0
```

User is logged in and is a member of two groups

```
INTENT_GROUPLIST userName:Luke.Steber status:0
jsonObjString:{"groupList":[{"gid":8434,"name":"Grocery"},
{"gid":8433,"name":"Hardware"}]} groupListSize:2
```

groupList JSON Example

```
{
    "groupList": [
    {
        "name": "Grocery",
        "gid": 174401
    },
    {
        "name": "Hardware",
        "gid": 184209
    },
    {
        "name": "All.Store",
        "gid": 8434
    },
    {
        "name": "Reception",
        "gid": 8433
    }
}
```

Request Login Status

A third-party client can request the login status of a user by sending this intent to the PTT Pro for Android. The client responds with the Login Status intent.

Prerequisites

- The minimum required PTT Pro for Android version is 3.3.10223.

Intent Definition

Name	Definition
Action	com.symbol.wfc.pttpro.ACTION_LOGIN_STATUS

ADB Example

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_LOGIN_STATUS
```

Request Group List

A third-party client can request a user's group list by sending this intent to PTT Pro for Android. The client responds with the Group List intent.

Prerequisites

• The minimum required PTT Pro for Android version is 3.3.10223.

Intent Definition

Name	Description
Action	`com.symbol.wfc.pttpro.ACTION_GROUP_LIST_REQUEST`

ADB Example

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_GROUP_LIST_REQUEST
```

Call Status

This intent alerts users about call status changes in PTT Pro for Android.

Prerequisites

• The minimum required PTT Pro for Android version is 3.3.10271.

Intent Definition

Name	Definition
Action	`com.symbol.wfc.pttpro.PTTPRO_CALL_STATUS`
Intent Type	broadcast
Extra	Returns information about call status
`PTT_PRO_CALL_STATUS`	String Possible values: • `CALL_STARTED_OUTGOING` • `CALL_STARTED_INCOMING` • `CALL_ENDED` • `CALL_FAILED`
`PTT_PRO_CALL_STATUS_CODE`	Int
`PTT_PRO_CALL_STATUS_REASON`	String
`PTT_PRO_CALL_STATUS_DETAILED_REASON`	String

Register for the broadcast as shown in the code example, listen with action `PTTPRO_CALL_STATUS`, and filter the result with extra types to get the status and other information from the broadcast.

Code Example

```
IntentFilter call_info = new IntentFilter();
call_status.addAction("com.symbol.wfc.pttpro.PTTPRO_CALL_STATUS");
registerReceiver(mReceiver,call_status);
```

Status Code

The PTT_PRO_CALL_STATUS_CODE has the following codes and reasons:

Status Code	Status Reason	Detailed Reason
0	UNKNOWN	The reason is unknown
1	NO_RESOURCES	There are no call resources available on the server.
2	USER_INITIATED	User-initiated call-clearing
3	UNAVAILABLE	The called party was unavailable
4	SIGNED_OUT	The called party was signed out of ESChat
5	NOT_RESPONDING	The called party was not responding.
6	BUSY	The called party was busy.
7	LIST_NOT_SYNCED	The group list was not in sync
8	GROUP_FULL	The group was full
9	NO_PRIVS	The user does not have high enough privilege to call this contact or group
10	DND	The user was in Do Not Disturb mode
11	CIRCUIT_CALL	The user was in or started a Circuit Switched voice call
12	CALL_DROPPED	The call was dropped
13	CRYPTO_ERROR	There was a security problem and the call was cleaned up
14	SILENT	The user device is in silent mode
15	USER_REJECT	The user rejected the incoming alert call
16	NO_ANSWER	There was no answer to the alert call
17	OVERRIDE	The user was removed from this call for a higher-priority call
18	DUPLICATE_GROUP	Received an invite to a group, but the call for that group is already in session

Status Code	Status Reason	Detailed Reason
19	REMOVED_FROM_GROUP	The User has been removed from a group while in a call with that group
20	NOT_SUPPORTED	The called party does not support the type or options of a call it's invited to
21	AMBIGUOUS_NAME	The name specified in a call does not have a single match so the call cannot proceed
22	EXTERNAL_NETWORK_UNAVAILABLE	The Called party can not be reached because the external network it is on was unavailable
23	LISTEN_ONLY	A listen Only Group Member sent an Invite to a group that is not in-call
24	TRY_AGAIN	The client should retry the call request again in 20 msec (or longer).
25	IDLE_TIMEOUT	The call ended with an idle timeout
26	TARGET_NOT_FOUND	The client attempts a call to an unknown group or user(s).
27	RECONFIGURED	Indicates a configuration change that forced the call to end
28	USER_INITIATED_WITH_BLOCK	User-initiated call clearing and the user will block the group

Status Code Usage Example

```
public void onReceive(Context context, Intent intent) {
  String action= intent.getAction();
  if(action.equals("com.symbol.wfc.pttpro.PTTPRO_CALL_STATUS")) {
    String status ="";

   if (intent.hasExtra("PTT_PRO_CALL_STATUS") {
      status = intent.getStringExtra("PTT_PRO_CALL_STATUS");
      Log.d("TAG","Call status:- "+status);
  }

        if(status.equals("CALL_ENDED") || status.equals("CALL_FAILED")) {
     Log.d("TAG","PTT_PRO_CALL_STATUS_CODE:-
 "+intent.getIntExtra("PTT_PRO_CALL_STATUS_CODE",0));
     Log.d("TAG","PTT_PRO_CALL_STATUS_REASON:-
 "+intent.getStringExtra("PTT_PRO_CALL_STATUS"));
     Log.d("TAG","PTT_PRO_CALL_STATUS_DETAILED_REASON:-
 "+intent.getStringExtra("PTT_PRO_CALL_STATUS_DETAILED_REASON"));
  }

}
```

Locate User

This intent returns a user's location within the same site or department. This intent requires the Location Area Service (LAS) for location data.

Prerequisites

- The minimum required PTT Pro for Android version is 3.3.10290.

- The ACTION_PTT _PRO_LOCATE intent must be registered to return the user location.

Intent Definition

Name	Definition
Action	`com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE`
Intent Type	broadcast
Extra 0	Returns the user's LAS Information.
Type	String
Name	user
Value	- If the extra name is user - Only one user can be configured. - The user name is mandatory. - The user name is case-sensitive. - Use the user name must be exactly as it is created in the PTT Pro server. - The user is located only if the user belongs to the same Site ID.

ADB Example:

```
adb shell am broadcast -a com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE --es
 user 'Gabriel'
```

Register Intent

Name	Definition
Action	`com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE_STATUS`
Extra 0	Returns `locate_status`
Type	Int
Name	locate_status
Value	0 indicates a failure and that LAS is not configured or LAS information is unavailable. 1 indicates a response for locating the user.
Extra 1	Returns `locate_result`
Type	String

Name	Definition
Name	locate_result

Value	If `locate_status` is 1, the user's LAS Information will be shared if `locate_status` returns 1. The user's friendly name is returned in JSON format.
	
	Example:
	
	``
	{
	"las_friendly_name": "Fishing Department"
	}
	``

Extra 2	Returns `locate_error_reason`
Type	String
Name	locate_error_reason
Value	If `locate_status` is returned as 0, the `locate_error_reason` provides the details about the failure. The reason can be one of the following: • Error while fetching LAS information. • The user is not registered. • The user is not logged in. • The user is not available. • LAS is not configured.
Extra 3	Returns `las_user_name`
Type	String
Name	las_user_name
Value	Returns the username that is passed as part of the `com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE` intent.

Code Snippet

Send Broadcast

```
Intent intent = new Intent();
intent.setAction("com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE");
    intent.putExtra("user", userName);
    sendBroadcast(intent);
```

Register Intent

```
IntentFilter locate_intent= new IntentFilter();
locate_intent.addAction("com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE_STATUS");
registerReceiver(mReceiver,locate_intent);


public void onReceive(Context context, Intent intent) {
String action= intent.getAction();
if(action.equals("com.symbol.wfc.pttpro.ACTION_PTT_PRO_LOCATE_STATUS")) {
```

```
    int locate_status = intent.getIntExtra("locate_status" , 0);
    String locate_result = intent.getStringExtra("locate_result");
  Log.i(TAG, "locate_status : "+locate_status+" locate_result :
 "+locate_result);
 }
}
}
```

PTT Pro for Android Sample Project

The wfc-pttpro-java-sample project is a sample Android app that allows a user to initiate PTT calls and sends PTT messages remotely by sending intents to the PTT Pro for Android on the same device.

The wfc-pttpro-java-sample project is available at github.com/ZebraDevs/wfc-pttpro-java-sample.