

Bootstrap Application for Android™

1.06 Release

Overview.....	1
1. Current Release Summary.....	1
2. Hardware Platform Support and Hardware Notes	1
3. New/Major Features or Fixes	2
4. Software Installation and Upgrade Instructions	2
6. Usage Notes.....	2
7. Technical Publications.....	40

Overview

This document comprises the release notes associated with the BLE Bootstrap Application for Android.

These notes are intended as a highlight of the major updates or changes to the code since the prior release and include some additional release history. This document is not a list of software-based features, an archive of every code change, a software test report, or qualification report. This document is provided for information purposes only and is subject to change without notice.

1. Current Release Summary

The following are the main details of the current release:

Release Type	Bootstrap Application for Android
FW Number:	Release 1.06
Release Date:	August 2022

Highlighted Code Updates: Initial product release

2. Hardware Platform Support and Hardware Notes

This application is compatible with Android versions 8 or later:

- The Android device must have the ability to serve as a WiFi hotspot.

- The Android device need not have internet access.
- All applications for access to data and device functions must be accepted.
- This bootstrap application is intended only for use with Zebra BLE IoT bridges.
- This application is provided as-is and may not be compatible with each hardware and software combination. We only test a sample of hardware devices and suspect not all hardware platforms will be compatible with this application.
- We performed testing with Android 8.1, 9, 10, and 12.

3. New/Major Features or Fixes

This is the initial release and posting of the application.

4. Software Installation and Upgrade Instructions

Zebra support is available for customers with valid support entitlements going back as far as the prior and one before that production release code (“R versions”).

Be sure to read and understand the entire Readme.txt file before attempting to install or use the application

Important Note

Downloading and installing this software application is your explicit acceptance of the software as-is and your agreement that Zebra is not liable for any damages of any type resulting from installation and / or attempted usage of this application.

6. Usage Notes

The following pictures show how and when to use the application. Please simply use this software version in place of the one listed in the presentation:



Zebra's BLE IoT Bridge Bootstrap Process (Beta Mobile Android Application)





IoT Bridge Configuration

The bootstrap process

- “IoT Bridges” are Zebra’s MB5xxx and MB6xxx BLE-to-WiFi devices
- What is IOT Bridge bootstrapping and why do we need it?
- What information does IOT Bridge gets bootstrapped with?
- How to bootstrap IOT Bridges?
- What happens during IOT Bridge bootstrap process?
- What happens after IOT Bridge is successfully bootstrapped?
- This is work the first time...don't wait to do this until you need it!





IoT Bridge Configuration

The bootstrap process

- The IoT Bridges come from factory “un-configured” (i.e. programmed with the same default config.)

Question: How do you configure the device with two buttons and two LEDs (no keyboard or screen)...?

Answer: You “bootstrap” the device

- Out-of-the-box bridges have no predefined WiFi network configuration
 - IoT Bridges do not know how to connect to the WiFi network
 - IoT Bridges do not know the SLE / ZLA / MWE / Gateway URL providing beacon filter configuration
 - IoT Bridges do not have the credentials to pull beacon filter configuration
- Bootstrapping is a process where factory defaults are used to configure the IoT Bridges
 - The process is the same for both the fixed (MB5xxx) and mobile (MB6xxx) IoT bridges





Bootstrapping

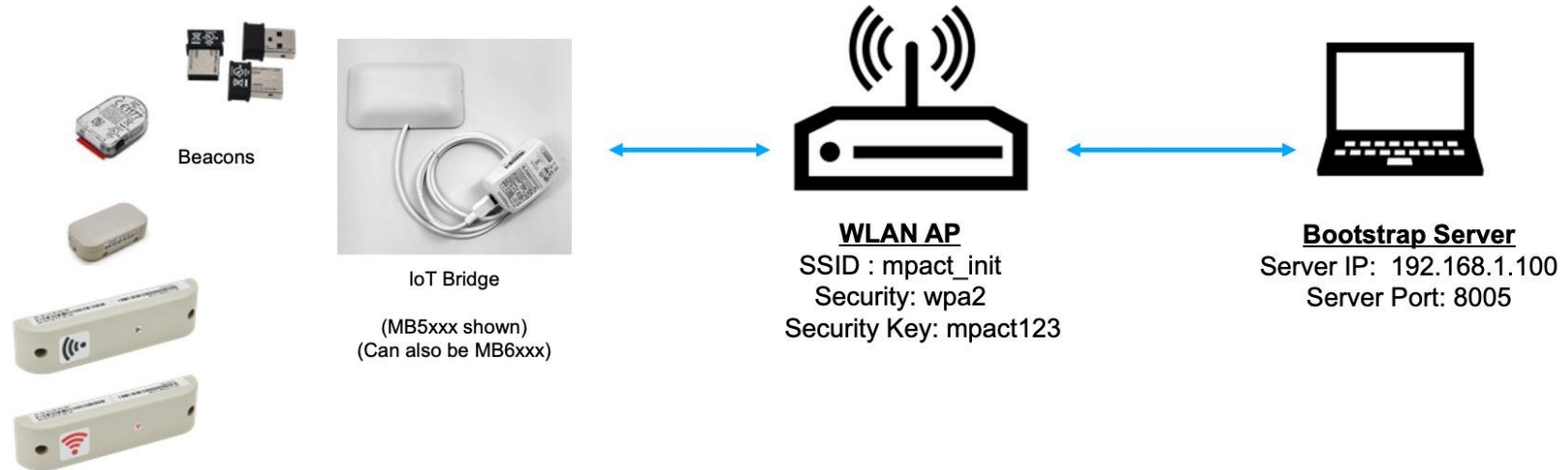
IoT Bridge Bootstrapping Process

- IoT Bridges validate the bootstrap configuration (badge_config.json) file
- Store the bootstrap information
- Reboot
- Join the network using the information from badge_config.json
- Pull the IoT Bridge beacon filter configuration file from server
- Parse and validate the IoT Bridge beacon filter configuration file
- Store beacon filter configuration information
- Start scanning for beacons



IoT Bridge Bootstrap Process Overview

Works with all IoT bridge's released FW versions





IoT BRIDGE BOOTSTRAP PROCESS OVERVIEW

Bootstrap Configuration Information

- Bootstrap configuration information is provided in JavaScript Object Notation (JSON)
 - IoT Bridges expect data adhering to this specific schema
 - Take some time to learn this process BEFORE you need hardware working (like for a demo)
- IOT Bridges need the following important information
 - WiFi Network connectivity details (SSID, security key, security type, WPA2 Enterprise connection details)
 - Gateway/ZLA/MWE/SLE URL serving beacon filter configuration
 - Gateway/ZLA/MWE/SLE authentication details (basic authentication details)
 - Frequency to pull beacon filter configuration





IoT BRIDGE BOOTSTRAP PROCESS OVERVIEW

Bootstrapping Server

- New IoT Bridges come out of the box looking to join a WiFi network with following details
 - SSID : mpact_init
 - Security: wpa2
 - Security Key: mpact123
- New IoT Bridges look to connect to a bootstrapping server running with following details
 - Server IP: 192.168.1.100 or the gateway IP
 - Server Port: 8005
- The IoT Bridges pull bootstrap configuration file called *badge_config.json*





IoT BRIDGE BOOTSTRAP PROCESS OVERVIEW

Sample Bootstrap Configuration File (badge_config.json)

```
{  
  "wifiProfiles": [{  
    "ssid": "Test5AP7532",  
    "securityKey": "aaaaeff12",  
    "securityType": "wpa2",  
    "wpaEnterpriseUser": "",  
    "wpaEnterprisePassword": "",  
    "wpaEnterpriseOuterIdentity": "",  
    "eapType": "",  
    "enable": true  
  }],  
  "gatewayConfigs": [{  
    "user": "superuser",  
    "password": "mpact123",  
    "receiverConfigURL": "http://10.21.201.29:8005/ReceiverConfig.json",  
    "configPullFrequencyInMins": 3,  
    "enable": true  
  }]  
}
```

Subject to change without notice



MOBILE ANDROID BOOTSTRAP APPLICATION

For BLE IoT Bridge Initial Configuration

- The mobile application performs the initial out-of-box configuration
 - Provides the wireless local area network attributes
 - Provides the receiver_config.JSON URL
 - Can configure multiple devices at once
 - Can allocate unique username-password pairs to units based on MAC ID
 - Application is provided as-is by Zebra to assist partner deployments
 - Can operate as the access point or file server or both depending on Android device capabilities
- Mobile application versions
 - Standard version: Uses readable device passwords in the clear
 - Enhanced version: Uses encryption to mask the passwords (requires Python)



MOBILE ANDROID BOOTSTRAP APPLICATION

For BLE IoT Bridge Initial Configuration

- System Requirements
 - Android device
 - 2.4 GHz 802.11 client (minimum)
 - To act as file server only
 - 2.4 GHz 802.11 hotspot (enhanced capabilities)
 - To serve as both soft AP and file server
 - May require cellular SIM and active account depending on device/carrier
 - Bluetooth® low energy (BLE)
 - Must be an unlocked device and you must have access to add apps and files
 - Zebra IoT bridges running a firmware version:
 - Unit firmware version determine the associated application capabilities
 - Unit FW can be updated (check with Zebra for details)
 - Computer running MS Excel or compatible spreadsheet program (open, edit, and save)
 - Computer with Python 3.8.x installed and running (<https://www.python.org/downloads/>)
 - Python is only required to mask passwords using the Enhanced Version of the application



MOBILE ANDROID BOOTSTRAP APPLICATION

For BLE IoT Bridge Initial Configuration



App Description	Firmware Version	Comment
Standard	2.7.6.1-010R <u>or earlier</u>	Only supports Android app in client mode (no hotspot, uses fixed IP address—file server)
Standard	2.7.6.2-001R or later	Support both client and hotspot mode with unique username-password pairs based on MAC ID (in the clear)
Enhanced (Enterprise)	3.0.0.0-007R or later	Support both client and hotspot mode with unique username-password pairs based on MAC ID (encrypted)

The above FW versions apply for Android 8, 9, and 10
For Android 11 it must be FW version 4.x or later (in testing now, not yet released)



MOBILE ANDROID BOOTSTRAP APPLICATION

For BLE IoT Bridge Initial Configuration

- Installation and Application Operation
 - Install the application on the mobile Android device
 - Applications are provided as-is
 - Applications can be downloaded (BLE engineering team’s internal reference only)
<https://zebra-my.sharepoint.com/:f/p/a7033c/Epx27Sk0sXJAgcdT7U9n2-QBM7YOPCiY8iIM1JtCDrOOLQ?e=BcTwqy>
 - Standard Version: com.zebra.mpactbootstrap-Signed-1.0.4.apk 
 - Enhanced Version: com.zebra.mpactbootstrapenc-Signed-1.0.4.apk 
 - Download the sample csv file
 - Standard Version: EnterpriseUsers.csv
 - Enhanced Version: PlainPasswordInputFile.csv, stored in PasswordEncryption_1.0.zip
 - Fill out the file with MAC ID, Username and Password in the csv file. Don’t change the first row.(i.e column headers)
 - If you are using the enhanced version with masked passwords
 - First time only: Run the command window: pip.exe install pycryptodome
 - Command to encrypt: python.exe PasswordFunctions.py encrypt PlainPasswordInputFile.csv EnterpriseUsers.csv
 - Command to decrypt: python.exe PasswordFunctions.py decrypt EnterpriseUsers.csv Decrypted.csv
 - Upload the file EnterpriseUsers.csv to the Android device
 - Android internal storage: /Download





MOBILE ANDROID BOOTSTRAP APPLICATION

EnterpriseUsers.csv Data File

	A	B	C	D	E
1	MacId	WpaEnterpriseUser	WpaEnterprisePassword	WpaEnterpriseOuterIdentity	Description
2	40:83:DE:D9:B6:29	User1	Password1	OuterId1	Description 1
3	40:83:DE:D9:B6:30	User2	Password2	OuterId2	Description 2
4	40:83:DE:D9:B6:31	User3	Password3	OuterId3	Description 3
5	40:83:DE:D9:B6:32	User4	Password4	OuterId4	Description 4
6	40:83:DE:D9:B6:33	User5	Password5	OuterId5	Description 5
7	40:83:DE:D9:B6:34	User6	Password6	OuterId6	Description 6
8	40:83:de:fe:42:EC	User7	Password7	OuterId7	Description 7

Notes:

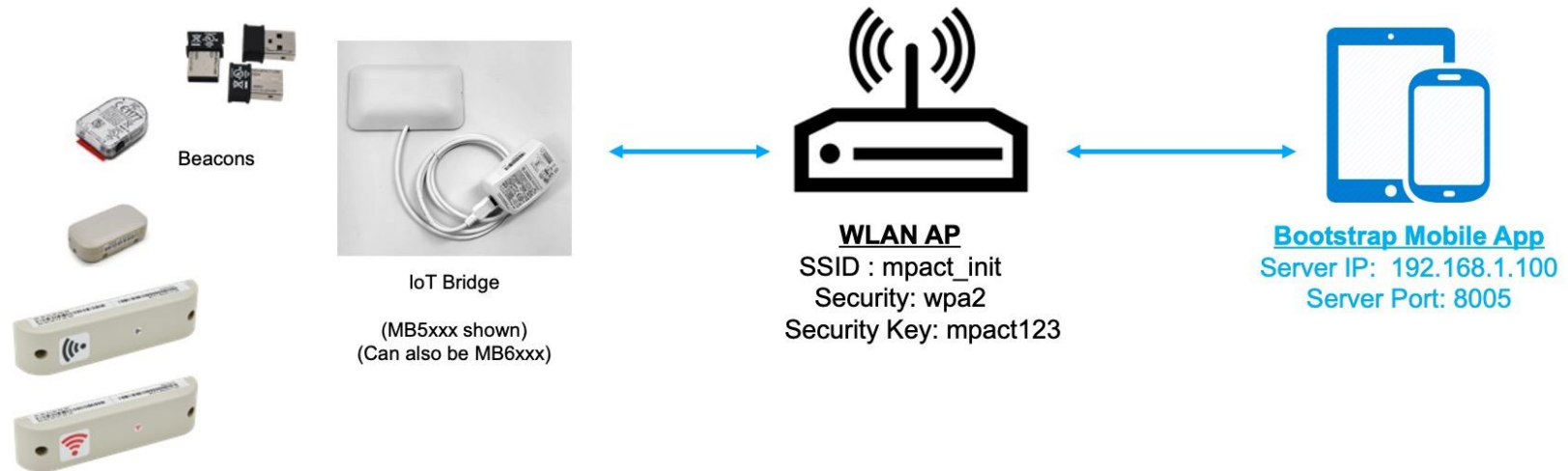
- Do not change the Row 1 column headers
- You can add as many additional rows as needed
- Do not change txt, font, or cell formatting
- Description column is for reference. It is not used during the bootstrap process





IoT BRIDGE BOOTSTRAP PROCESS OVERVIEW

Mobile Android Bootstrap App as the Bootstrap File Server





IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrap with the Mobile Android Application

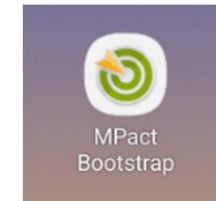
- Works with **ALL** IoT Bridge FW releases (MB5xxx and MB6xxx)
- New IoT Bridges come out of the box looking to join a WiFi network with following details
 - SSID : mpact_init
 - Security: wpa2
 - Security Key: mpact123
- New IoT Bridges look to connect to a bootstrapping mobile device running with following details
 - Server IP: 192.168.1.100
 - Server Port: 8005
- The IoT Bridges pull bootstrap configuration file called *badge_config.json*



IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrap with the Mobile Android Application

- The badge_config.json is hosted in the Android mobile App named “MPact Bootstrap”
- The wifi setting represents the wifiProfiles in badge_config.json
 - The unique username-password pairs based on MAC ID will be put into wifi setting
- The gateway setting represents the gatewayConfigs in badge_config.json
- The MPact Bootstrap app takes effect upon loading and runs either in the foreground or background until the application is stopped. Once opening the app it runs in foreground (showing the UI) or its then background. It runs until you manually stop the application. The application gives visible notifications in both foreground and background operation.





IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Mobile App

- Sample wifi settings in the App
- Same for both versions



Enhanced/Enterprise

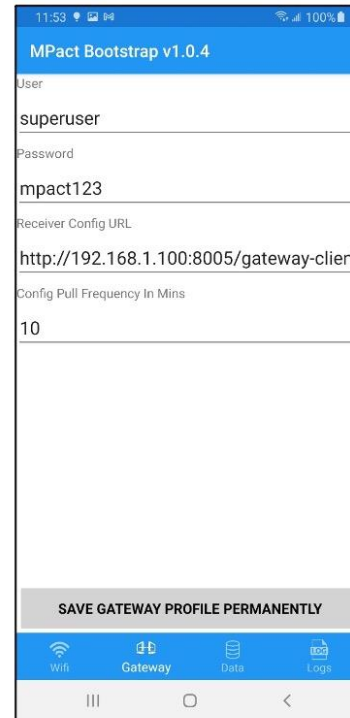




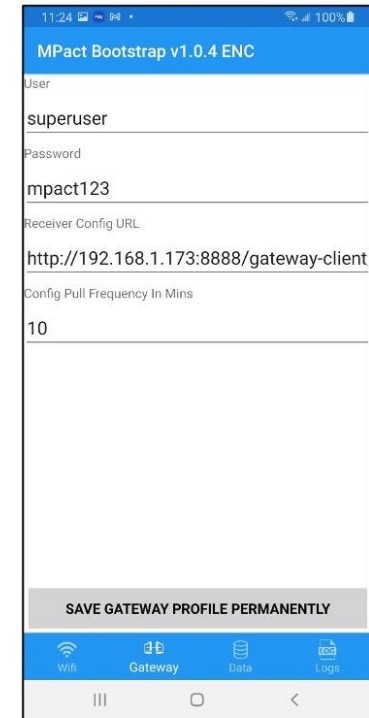
IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Mobile App

- Sample gateway settings in the App
- Same for both versions



Enhanced/Enterprise



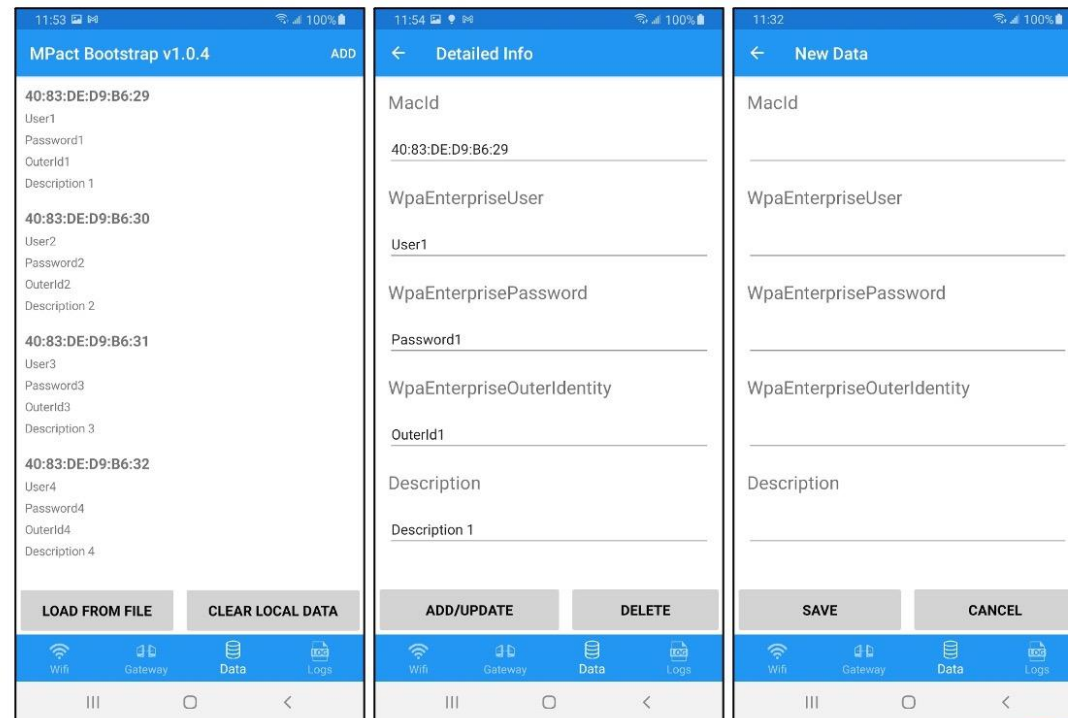


IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Mobile App

All Screens From the Standard Version

- Only effective for IoT Bridge with firmware **3.0.0.0-007R** or later
- Loading data from file:
 - \Download\EnterpriseUsers.csv
 - A.K.A. the “Data File” or “Local Data File”
- Standard version:
 - Password field is plain text
 - User can add/update/delete based on existing data
 - User can add new data
 - User can clear local data
 - The data file \Download\EnterpriseUsers.csv won't be deleted



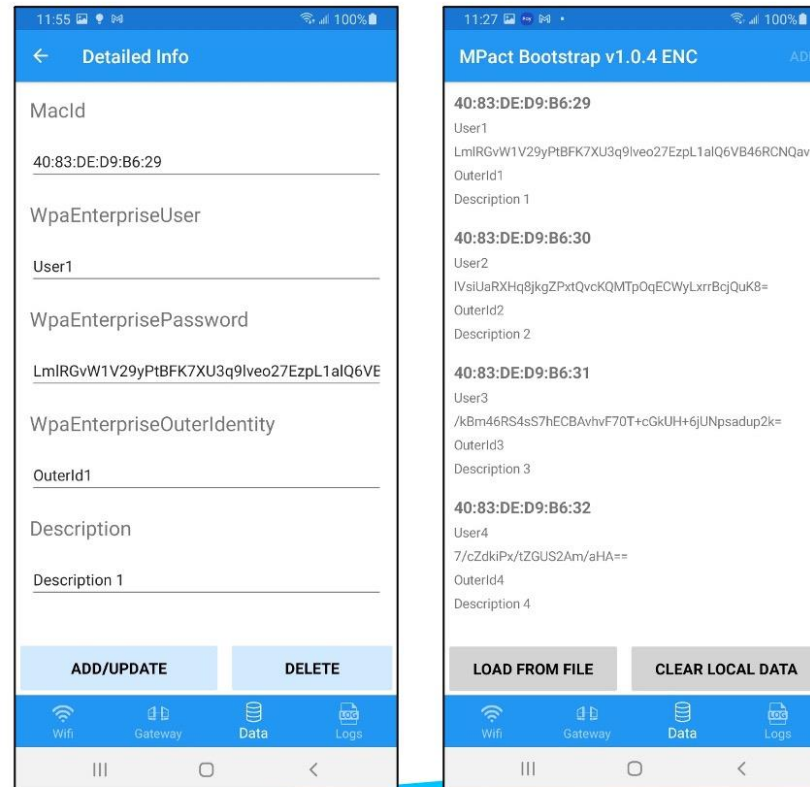


IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Mobile App

- Only effective for IoT Bridge with firmware **3.0.0.0-007R** or later
- Loading data from file:
 - \Download\EnterpriseUsers.csv
- Enhanced version:
 - Password field is encrypted
 - Data is read-only
 - User can clear local data
 - The data file \Download\EnterpriseUsers.csv won't be deleted

All Screens From the Enhanced/Enterprise Version





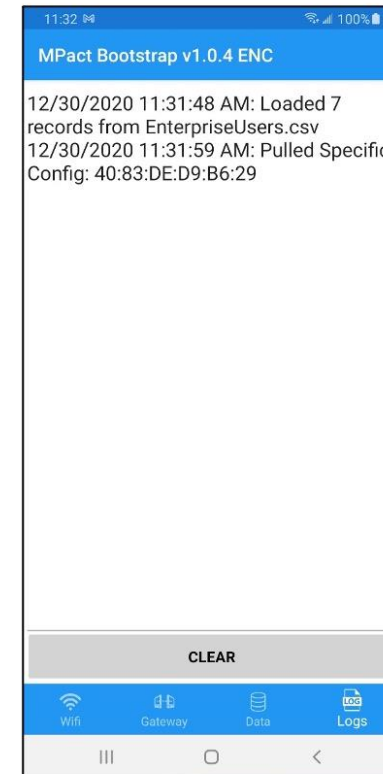
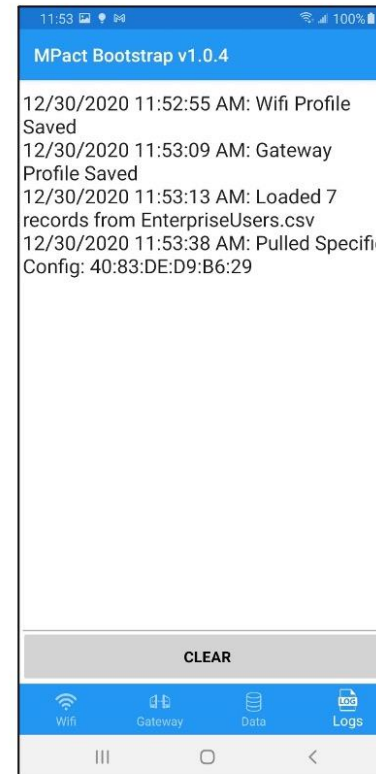
IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Mobile App

Activity Log File

- Captures a record of all major app activity and actions
- Can be used to verify which units were bootstrapped (i.e. pulled the bootstrapConfig file)
- User can copy out the permanent log from the Android device:
 - \Download\MPactBootstrap.log
- User can check logs in the app
- User can clear logs in the app
 - The permanent log file \Download\MPactBootstrap.log won't be deleted

Enhanced/Enterprise

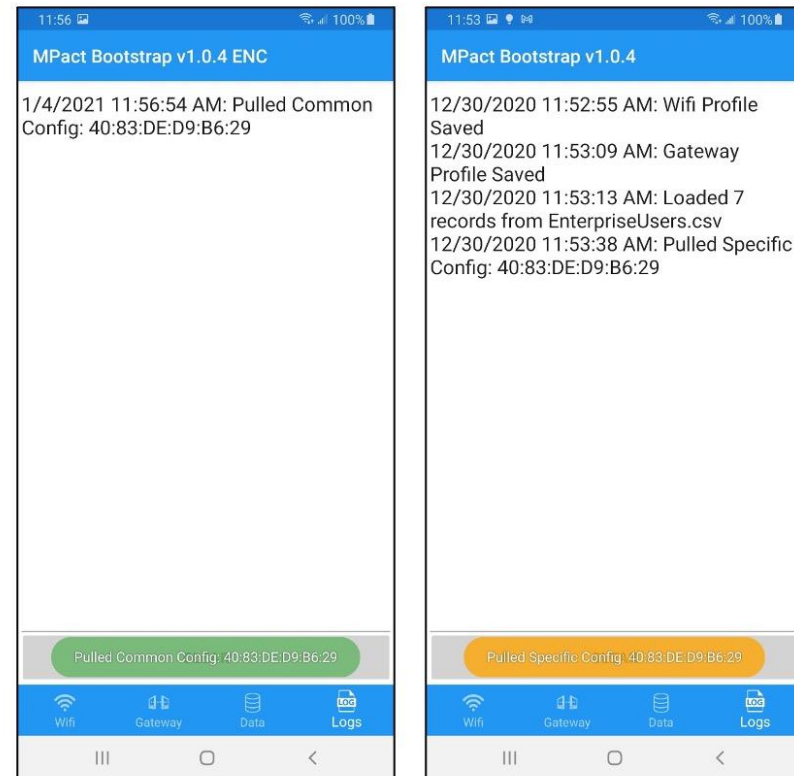




IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Mobile App

- The foreground app gives notifications when an IoT Bridge pulls config:
 - Common config: green
 - Special config based on MAC ID: orange
- Same behaviors for both app versions

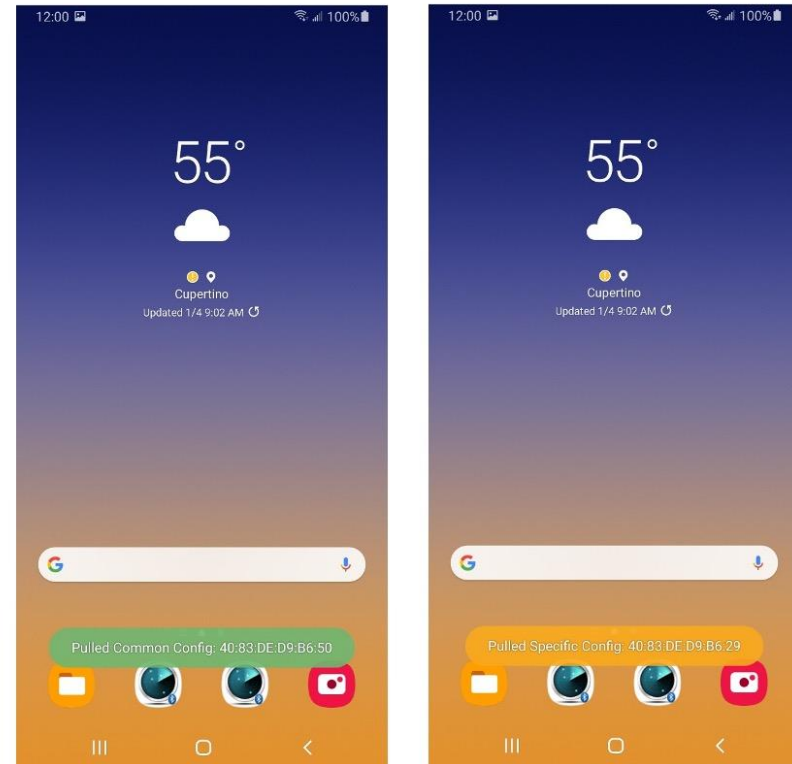




IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Mobile App

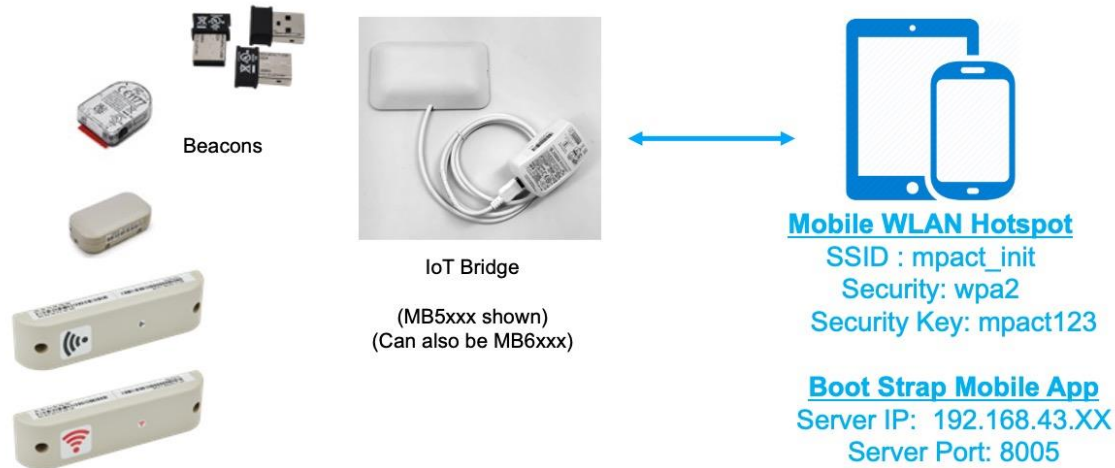
- The background app gives notifications when an IoT Bridge pulls config:
 - Common config: green
 - Special config based on MAC ID: orange
- Same behaviors for both app versions





IoT BRIDGE BOOTSTRAP APPLICATION OVERVIEW

Using Android Bootstrap Mobile Application as the WLAN AP and Bootstrap File Server





IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Hotspot and Mobile App

- Works with IoT Bridge FW releases **2.7.6.2-001R** onwards
- New IoT Bridges come out of the box looking to join a WiFi network with following details
 - SSID : mpact_init
 - Security: wpa2
 - Security Key: mpact123
- New IoT Bridges look to connect to a bootstrapping mobile device running with following details
 - Server IP: Android Hotspot device IP address (Like 192.168.43.XX for Android mobile hotspot)
 - Server Port: 8005
- The IoT Bridges pull bootstrap configuration file called *badge_config.json*





IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Hotspot and Mobile App

- The badge_config.json is hosted in the Android mobile App named “MPact Bootstrap”
- The wifi setting represents the wifiProfiles in badge_config.json
 - The unique username-password pairs based on MAC ID will be put into wifi setting
- The gateway setting represents the gatewayConfigs in badge_config.json
- The MPact Bootstrap app takes effect upon loading, even if it is in background. It gives notifications in both foreground and background.

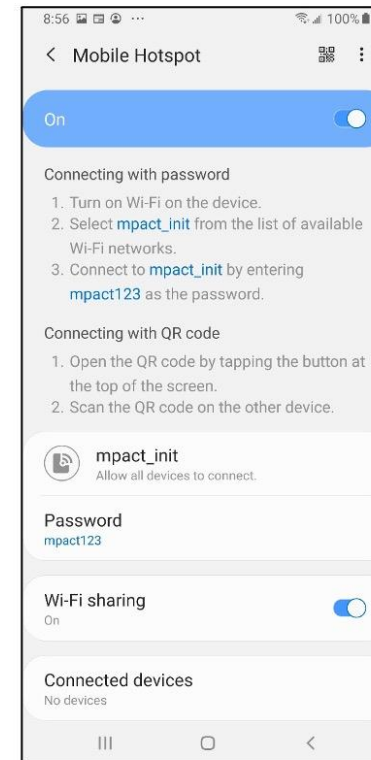




IoT BOOTSTRAP PROCESS OVERVIEW

Bootstrapping with Android Hotspot and Mobile App

- Sample Hotspot settings in the Android device:
- All the rest settings and configuration processes are the same as previous one



IoT BOOTSTRAP PROCESS OVERVIEW

Differences between IoT Bridge FW releases

- **2.7.6.2-001R** before and onwards:
 - Notification shows IoT Bridge's IP Address
 - IoT Bridge can only pull the common config
 - Same behaviors for both app versions

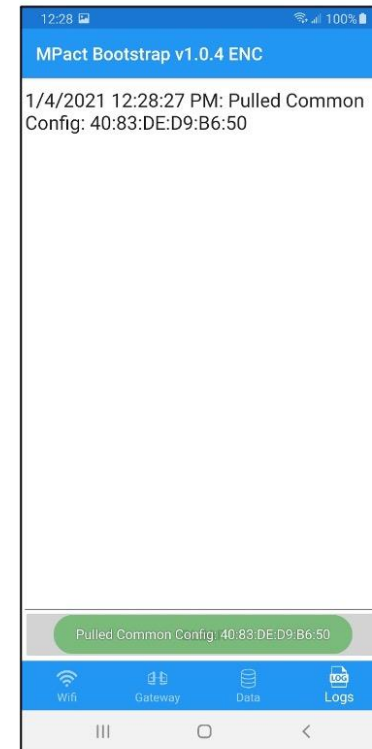




IoT BOOTSTRAP PROCESS OVERVIEW

Differences between IoT Bridge FW releases

- **3.0.0.0-007R** or later:
 - Notification shows IoT Bridge’s MAC ID
 - IoT Bridge can pull both common config and specific config
 - Same behaviors for both app versions

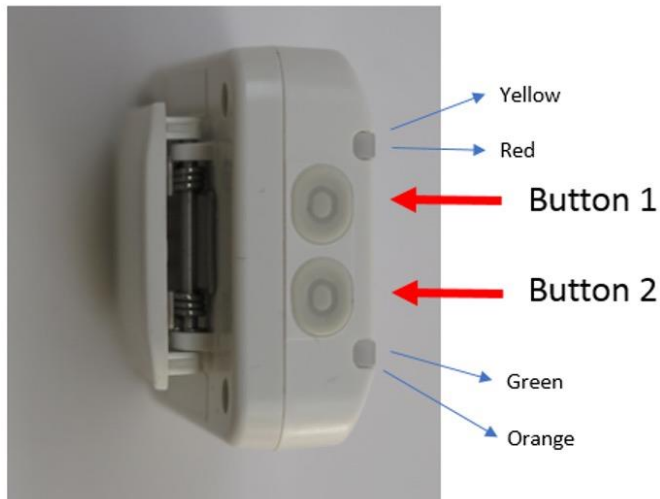






GENERAL BOOTSTRAP PROCESS DETAILS

Buttons, Button pushes and Re-Bootstrapping



Button 2 – Restart, Off, Factory Default

- Hold for 10-20 seconds (reboot)
- Hold for 20-30 seconds (switch off)
- Hold for 30-40 seconds (restore to factory default)





GENERAL BOOTSTRAP PROCESS DETAILS

Bootstrap Checklist Items

IOT Bridges (Fixed, Mobile)	Configuration Files
Chargers	<ul style="list-style-type: none">- badge_config.json- IOT Bridge beacon filter configuration
Cables	Documents
Access Point	Beacons
Software	Toolbox
<ul style="list-style-type: none">- Restserver	Laptop
<ul style="list-style-type: none">- Python	WIFI Network configuration details
<ul style="list-style-type: none">- Wireshark	SLE/MWE/ZLA/Server connection details
Android device	Applications and related files

This is an example. What is actually needed for a particular effort depends on the bootstrap approach used





GENERAL BOOTSTRAP PROCESS DETAILS

Bootstrap Checklist Items

IOT Bridges (Fixed, Mobile)	Configuration Files
Chargers	<ul style="list-style-type: none">- badge_config.json- IOT Bridge beacon filter configuration
Cables	Documents
Access Point	Beacons
Software	Toolbox
<ul style="list-style-type: none">- Restserver	Laptop
<ul style="list-style-type: none">- Python	WIFI Network configuration details
<ul style="list-style-type: none">- Wireshark	SLE/MWE/ZLA/Server connection details
Android device	Applications and related files

This is an example. What is actually needed for a particular effort depends on the bootstrap approach used





BOOT STRAP PROCESS DETAILS

Common problems faced during bootstrapping

Incorrect SSID	SSID mpact_init disabled
Mis-typed security key	DHCP server disabled
Complicated passwords	Invalid beacon filter configuration URL
Incorrect outer identity	Missing port number in URL
Invalid badge_config.json syntax	Invalid eap type





BOOTSTRAP PROCESS DETAILS

Special characters usage in badge_config.json

Special and control characters must be denoted by escaping them.

Special Character	Character	Escaped Notation
Quotation mark	"	\ " (the \ is added in front of the ")
Backslash or Reverse Solidus	\	\\
Forward slash or Solidus	/	\/
Backspace	\b	\\b
Form feed	\f	\\f
New line	\n	\\n
Carriage return	\r	\\r
Horizontal tab	\t	\\t





BOOSTSTRAP PROCESS DETAILS

Special characters usage in badge_config.json ...

- Bootstrap Configuration information can get complicated and confusing with special characters.
- Things To **Avoid**
 - Do not have single quotes and double quotes next to one another (`"'\\"`)
 - Do not have sequence of forward and backward slashes together (`"\\//\\\\\\//\\\\\\//"`)
 - Do not have sequence of double quotes (`"\"`)
 - Do not have sequence of spaces (`" "`)
 - Do not have spaces, tabs and carriage returns together (`" \r \t "`)
 - Always verify JSON strings in a tool such as <https://jsonlint.com>





BOOTSTRAP PROCESS DETAILS

Sample special characters in password

Sample Un-Escaped Password (“incorrect”)

```
5>/NK5MogOc6lrt}m3\.:dN/rbR:zzpylGy5niZqVh"A~W5.h<},2b2?T'?w'Xf2
```

Fully Escaped Sample Password (“correct”)

```
5>/NK5MogOc6lrt}m3\\.:dN/rbR:zzpylGy5niZqVh"A~W5.h<},2b2?T'?w'Xf2
```

Added the “\” in front the special characters to “escape” them.

Escaping makes the password valid as intended



7. Technical Publications

Please refer to the Manuals portion of the Zebra Bluetooth Low Energy (BLE) Devices Support section under the Location Solutions portion of the Support and Downloads tab of Zebra.com for the associated user guide, Schema document and other technical publications to help with the configuration and operation of the IoT Bridges

--- END ---

This document data and information is subject to change without notice

© Zebra Technologies, Inc. 2022. All rights reserved.