

# IoT Bridge

## Hostname Support



**ZEBRA**

---

## Copyright

© 2020 ZIH Corp. and/or its affiliates. All rights reserved. ZEBRA and the stylized Zebra head are trademarks of ZIH Corp., registered in many jurisdictions worldwide. All other trademarks are the property of their respective owners.

COPYRIGHTS & TRADEMARKS: For complete copyright and trademark information, go to [www.zebra.com/copyright](http://www.zebra.com/copyright).

WARRANTY: For complete warranty information, go to [www.zebra.com/warranty](http://www.zebra.com/warranty).

END USER LICENSE AGREEMENT: For complete EULA information, go to [www.zebra.com/eula](http://www.zebra.com/eula).

---

## 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.

## Table of Contents

Copyright	-2
Terms of Use	-2
Proprietary Statement	-2
Product Improvements	-2
Liability Disclaimer	-2
Limitation of Liability	-2
Table of Contents	-3
Support For Hhostname as Part of URL	-4
URL Syntax	-4
Hhostname Character Limitations	-4
Fully Qualified Hhostname Size Limitations	-4
New JSON Attributes	-5
serviceUrlsHhostname	-5
configService	-5
Sequence of Steps For ReceiverConfig.json Change	-5
Badge_Config.Json Sample	-6
Receiver Filter Configuration Sample	-6
Jamaica Firmware Upgrade JSON Fragment	-7
Kingston Firmware Upgrade JSON Fragment	-8
Notes And Constraints	-8

## Support For Hostname as Part of URL

### URL Syntax

<protocol>://<domainOrIPAddress>:<port>/path/filename

Current supported maximum length of URL is 256.

### Hostname Character Limitations

Hostname is a text string consisting of the letters A through Z (upper or lower case), digits 0 through 9, the minus sign (-), and the period (.). Note, the period is only allowed as the last character of the host name if it is the delimiter of the full domain name (FQDN). No spaces are permitted as part of a name. The first character must be an alphabetic character and the last character must not be a minus sign or period.

The following characters are not allowed as part of hostname.

- backslash (\)
- slash mark (/)
- colon (:)
- asterisk (\*)
- question mark (?)
- quotation mark (")
- less than sign (<)
- greater than sign (>)
- vertical bar (|)

The following are some sample hostnames

- test-system
- Debian89-lab-machine
- janaki12345678901234567890123456789012345678901234567

### Fully Qualified Hostname Size Limitations

A fully qualified hostname name can have a maximum length of 255 bytes including the separators. A hostname can contain many labels. Maximum length for a label is 63 bytes.

The following are some sample fully qualified hostnames:

- abc123456789012345678901234567890123456789012345678901234567890.mweng.lan
- abc123456789012345678901234567890123456789012345678901234567890.abc123456789012345678901234567890123456789012345678901234567890.mweng.lan
- test-system.us.ca.nor-cal.zebra.mview-campus.eng
- a69-192-36-108.deploy.static.akamaitechnologies.com
- whitehouse.gov
- dfw25s13-in-f68.1e100.net
- any-in-2044.1e100.net
- janaki123456789012345678901234567890123456789012345678901234567890.mweng.lan

## New JSON Attributes

### serviceUrlsHostName

*serviceUrlsHostName* section has been introduced. The URLs that are defined under this section can have hostname or IP address in their definition. *serviceUrls* section is retained for compatibility purposes. In some deployment, units could fail during the firmware upgrade process. Under this condition the unit should still be able to use the *ReceiverConfig.json* and use the *serviceUrls* section attributes to continue with its normal operation.

**Table 1** Firmware Behavior with ServiceUrlsHostName and ServiceUrls Sections

serviceUrls	serviceUrlsHostName	Action	Notes
Present	Absent	If hostnames are present as part of URL they will be resolved. If the hostname resolution fails unit will reboot (previous behavior).	This is the recommended configuration once all units have moved to firmware that supports hostname resolution in URL.
Absent	Present	Firmware would use the hostname or IP provided as part of <i>serviceUrlsHostName</i> section.	
Present	Present	Firmware would use the hostname or IP provided as part of <i>serviceUrlsHostName</i> section. Units that fail during firmware upgrade would still be able to use the <i>serviceUrls</i> section	This is the recommended configuration for transitioning all units in a deployment running 2.7.0.0-014R to firmware with support for hostname in URL. Note that firmware with no support for hostname would use the <i>serviceUrls</i> section.
Absent	Absent	Invalid configuration for very first time. Otherwise, the units will use the previous <i>serviceUrls</i> or <i>serviceUrlsHostName</i> section configuration	

### configService

This is the URL used by the Receivers to pull their filter configuration. This is defined in *badge\_config.json*. Previously, to change this URL the units had to be bootstrapped again. By allowing the user to define *configService* URL in the Receiver Filter Configuration the URL can be changed dynamically if needed.

## Sequence of Steps For ReceiverConfig.json Change

1. Upgrade Firmware from 2.7.0.0-014R to firmware with support for hostname in URL.
2. Do not remove section
3. *serviceUrls* in *ReceiverConfig.json*. This will be used by receivers who have failed to upgrade their firmware or units that are yet to upgrade themselves.
4. Add section

5. *serviceUrlsHostName* to *ReceiverConfig.json*. This section can have hostname or IP in the URL. All units with firmware with hostname support will use this section and ignore the *serviceUrls* section.
6. If needed and if it is determined that all units have upgraded to firmware supporting the hostname then you can have
7. *ReceiverConfig.json* with just *serviceUrls* section. The *serviceUrls* section can have hostname or IP.

## Badge\_Config.Json Sample

This can be done for a new site deployment where the receivers have been shipped with 2.7.2.0-XXXXR firmware or upwards.

```
{
  "wifiProfiles": [{
    "ssid": "Test5AP7532",
    "securityKey": "abcdef1234567",
    "securityType": "wpa2",
    "wpaEnterpriseUser": "",
    "wpaEnterprisePassword": "",
    "eapType": "",
    "enable": true
  }],
  "gatewayConfigs": [{
    "user": "superuser",
    "password": "mpact123",
    "receiverConfigURL": "http://z21s-zlatest04.cluster-2003.wherenet.com:8005/ReceiverConfig.json",
    "configPullFrequencyInMins": 3,
    "enable": true
  }]
}
```

## Receiver Filter Configuration Sample

```
{
  "scanIntervalInMilliseconds": 500,
  "wifiTransmissionInterval": 10,
  "heartbeatIntervalInSeconds": 30,
  "timePullIntervalInMinutes": 5,
  "beaconWhiteList": [{
    "uuid": "FE913213-B311-4A42-8C16-47FAEAC938DC",
    "type": "mpact"
  }, {
    "uuid": "FE913213-B311-4A42-8C16-47FAEAC938AB",
    "type": "mpact"
  }, {
    "uuid": "00001200-0000-1000-8000-00805F9B34FB",
    "type": "ibeacon"
  }, {
    "uuid": "FE913213-B311-4A42-8C16-47FAEAC938FE",
    "type": "mpact"
  }],
}
```

```

"profiles": [{
  "name": "Location Beacons",
  "profileLow": 1,
  "profileHigh": 999,
  "beaconType": "Mobile",
  "proximityRanges": [{
    "name": "Near",
    "rssiLow": null,
    "rssiHigh": null,
    "beaconHitCount": 1,
    "scanRetentionInterval": 1,
    "sendClosestOnly": false,
    "suppressRepeats": true
  }]
}, {
  "name": "Asset Beacons",
  "profileLow": 1000,
  "profileHigh": 65535,
  "beaconType": "Fixed",
  "proximityRanges": [{
    "name": "Near",
    "rssiLow": null,
    "rssiHigh": null,
    "beaconHitCount": 1,
    "scanRetentionInterval": 1,
    "sendClosestOnly": false,
    "suppressRepeats": true
  }]
}],
"serviceUrlsHostName": {
  "timeService": "http://janaki-test.mweng.lan:8005/gettime",
  "eventService": "http://janaki-test.mweng.lan:8005/event/",
  "healthService": "http://janaki-test.mweng.lan:8005/health/",
  "firmwareService": "http://janaki-test.mweng.lan:8005/firmware/",
  "configService": "http://janaki-test.mweng.lan:8005/ReceiverConfig.json"
},
"serviceUrls": {
  "timeService": "http://10.21.201.29:8005/gettime",
  "eventService": "http://10.21.201.29:8005/event/",
  "healthService": "http://10.21.201.29:8005/health/",
  "firmwareService": "http://10.21.201.29:8005/firmware/"
}
}

```

## Jamaica Firmware Upgrade JSON Fragment

```

"firmware": {
  "gaugeFirmware": "2.7.2.0-002D_GAUGE-BT000311-01_9689CECC_7B2E905F.srec",

```

```
"bleFirmware": "2.7.2.0-002D_BLE-FIRMWARE_B401AF74_FFE51BA8.bin",
"wifiFirmware": "2.7.2.0-002D_WIFI-FW_B561BEFF_544129F3.bin",
"bootloader": "2.7.2.0-002D_BOOTLOADER_990426AF_211BE077.bin",
"mainFirmware": "2.7.2.0-002D_MPACT-MB6000-01-WR_1B395DE3_229B2F04.elf"
}
```

### Kingston Firmware Upgrade JSON Fragment

```
"firmware": {
  "gaugeFirmware": "2.7.2.0-002D_GAUGE-BT000311-01_9689CECC_7B2E905F.srec",
  "bleFirmware": "2.7.2.0-002D_BLE-FIRMWARE_B401AF74_FFE51BA8.bin",
  "wifiFirmware": "2.7.2.0-002D_WIFI-FW_B561BEFF_544129F3.bin",
  "bootloader": "2.7.2.0-002D_BOOTLOADER_990426AF_211BE077.bin",
  "mainFirmware": "2.7.2.0-002D_MPACT-MB5000-01-WR_FE313EC8_FA0ABE04.elf"
}
```

### Notes And Constraints

1. First upgrade to 2.7.2.0-002D firmware from firmware 2.7.0.0-014R.
2. *configService* has been added in *ReceiverConfig.json*. This will allow to change the receiver configuration URL without having to re-bootstrap a configured unit. You must re-bootstrap the units if the WIFI settings are changed.
3. *badge\_config.json* can also use the hostname. This will work only with firmware that has hostname support (2.7.2.0-002D upwards).
4. You can mix and match hostname and IP address.
5. If a hostname cannot be resolved the receiver would reboot itself after 5 retries.
6. Different Operating Systems have different limitations on the size of hostname and number of labels. Please refer to the OS manuals.
7. Once you upgrade to 2.7.2.0-002D and if you change the configService URL to include the hostname you cannot downgrade to older firmware.
8. If
9. *configService* URL is provided incorrectly the receiver would fail to pull its filter configuration and would only work after the configuration service runs at the service endpoint and provides a valid configuration.





