



Cassia E1000-WP1 Wirepas Gateway Instructions

[Home](#) » [Cassia](#) » Cassia E1000-WP1 Wirepas Gateway Instructions 



**Cassia Wirepas Gateway Configuration
Instructions**
Release date: Dec 5, 2023

Contents

- [1 Introduction](#)
- [2 Troubleshooting](#)
- [3 Documents / Resources](#)
 - [3.1 References](#)
- [4 Related Posts](#)

Introduction

1. Currently, all Cassia Wirepas enabled gateways (E1000-WP, X2000-WP, and ATX2000WP) come with two Bluetooth radio chips. For the Wirepas enabled gateway, Cassia offers two different models, e.g.
 - a. E1000-WP1 / X2000-WP1 / ATX2000-WP1 gateway: one radio chip (chip1) supports Wirepas protocol and another radio chip (chip 0) supports Bluetooth protocol.
 - b. E1000-WP2 / X2000-WP2 / ATX2000-WP2 gateway: both Bluetooth radio chips are converted to support Wirepas protocol.

2. To obtain a Cassia Wirepas gateway, please contact your Cassia sales representative. At this time, customers cannot self-upgrade their existing gateway to support Wirepas protocol. This option will be available in the near future. Currently, Wirepas stack and BLE stack can only be loaded by Cassia engineers in Cassia China office and factories before shipping, therefore, it is not possible to switch the gateway between Wirepas function and Bluetooth Low Energy function in the field.
3. The Cassia Wirepas gateway will come with preconfigured dedicated firmware and container APP. Please ONLY use the gateway firmware listed in Step 3, see below. Please don't delete or reset the container and don't delete the APP.
4. When you require technical support, please assign this gateway to a Cassia IoT Access Controller (AC) server hosted on the Cloud. This will allow Cassia's support team to remotely troubleshoot your Wirepas gateway.

Step 1. Connect Gateway's Wi-Fi Hotspot and Login Gateway's Local Console

Please connect to the gateway's Wi-Fi hotspot with SSID: cassia-xxxxxx (the xxxxxx corresponds to the last 6 digits of the gateway's MAC address). The password of the Wi-Fi hotspot is the same as the SSID. For example, if the gateway's MAC address is "CC:1B:E0:E0:96:DC", the Wi-Fi hotspot SSID and its default password will be "cassiaE096DC".

Please open Google Chrome on your laptop, enter the gateway's default IP address 192.168.40.1, and then log in. The first time you log in, you need to set the gateway's login password. The password should include numbers, letters, and special characters. The password length should be between 8 to 20 characters. Proceed to logging into the gateway's web page.

Step 2. Network Connection Setup

By default, a gateway has been preconfigured to DHCP for Ethernet connection. Please plug in an Ethernet cable to the Cassia gateway and check the Ethernet IP. In case of the Wi-Fi or cellular connection configuration, please refer to the Cassia Installation Guide:

(https://www.cassianetworks.com/download/docs/Cassia_Gateway_Installation_Guide.pdf).

Step 3. Conduct Version Check the Firmware, Container, and App

Please check whether the correct version of firmware, container, and App are installed. Note that resetting the container will remove the App and configurations. Please download the gateway firmware and the APP from the link below: <https://home.mycloud.com/action/share/bebdc792-2220-45a9-8a43-6f98bd59390b>

- E1000-WP gateway firmware: E1000_2.1.2303wp2.2308171812.gz.gpg
- X2000-WP and ATX2000-WP gateway firmware: X2000_2.1.wp.2311160813.gz.gpg
- APP version for dual chip model: wirepasDual.1.24.tar.gz
- APP version for one chip model: wirepasOne.1.22.tar.gz

Please download the container version 2.0.1 from the link below:

https://www.cassianetworks.com/download/firmware/container/ubuntu_XE1000_2.0.1.tar.gz

Status	Basic	Container	Events	Other
Model	X2000			
MAC	CC:1B:E0:E3:50:20			
Working Mode	AC Managed			
AC-Gateway Protocol	CAPWAP			
Uplink	Wired			
ETH IP	192.168.168.25			
WLAN IP	192.168.40.1			
Cellular IP				
Country/Region	United States			
Firmware Version	2.1.wp.2310191713			
Up Time	6min 23sec			
AC Online Time	5min 29sec			
CPU Usage	18.85%			
Memory Usage	7.56%			
Storage Usage	18.19MB / 111.20MB			

Figure 1 Check Firmware

Status

Basic

Container

Events

Other

Operating System

Ubuntu 20.04 LTS

Container Status

running

Container Version

2.0.1

CPU Usage

18.46%

Memory Usage

3.96%

Storage Usage

1.26GB / 2.28GB

Transmit Rate

0.57KB

Transmit Bytes

11.73KB

Receive Rate

0.61KB

Receive Bytes

11.64KB

Installed APPs (1)

#	Name	Version	Action
1	wirepasDual	1.22	<div>Del</div>


Figure 2, Check container and App

Step 4. Wirepas Configuration

Please configure Wirepas-related parameters in the Wirepas configuration section in the container tab of the AC/gateway console.

Table 1: Parameter description

Parameter	Description
Mqtt_hostname	The IP address of the Wirepas MQTT broker
Mqtt_port	The MQTT communication port to the Wirepas background. The default value is 8883 (secure) or 1883 (local)
Mqtt_username	The MQTT username
Mqtt_password	The MQTT password
Gateway_id	Each gateway in a Wirepas mesh network has a unique ID. This parameter should be a decimal number
Network_address [1,2]	The network address is used by the radio to detect valid transmissions and to filter out both the noise and other transmissions that do not belong to the same network. A network address must be identical for all nodes within the same network. The available Network addresses range is between 1 and 5570559, between 5636096 and 11141119, and between 11206656 and 16777215.
Network_discovery_channel [1,2]	The network channel must be identical for all nodes within the same network. The radio channel range is between 1 and 40.
Node_address [1,2]	Each node in the network must have a unique node address within the network. The two chip should have different node addresses (Node_address_1 and Node_address_2). The node address range is between 1 and 2147483647 and between 2164260864 and 4294967293.
Node_roles [1,2]	It can be configured as “sink csma-ca” or “sink”. SINK: A device that is usually connected to a server backbone. This is the final destination for all the data packets sent to the AnySink address. Similarly, all diagnostic data generated by the network itself are transmitted to a sink device. CSMA-CA mode sink: When this is enabled, the sink keeps the receiver enabled all the time when it is not transmitting. Then, the latency on sending data to the sink is way faster with the expense of higher power consumption. Intended to be used only with mains-powered devices.
Authentication_key, Cipher_key, Ca_certs	Security related parameters. When the user set parameter Authentication_key, Cipher_key to thirty-two F letters (which means FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF) the original configuration will be cleared
Output_power	It can be configured as FCC (20 dbm) for North America or ETSI (10 dbm) for Europe
Extra	For secure mode (TLS cert file provided), please configure this parameter as “-mqtt_reconnect_delay 20”. Please set the Mqtt_port parameter accordingly. For unsecured mode (TLS cert file not provided), please configure this parameter as “-mqtt_reconnect_delay 20 -mqtt_force_unsecure”. Parameter mqtt_force_unsecure will disable the TLS handshake and establish connections to the unsecured port (default: 1883). Please remember to set the Mqtt_port parameter accordingly

 **WirepasDual Config**

Mqtt_hostname	Mqtt_port
<input type="text" value="cassia.prod-wirepas.com"/>	<input type="text" value="8883"/>
Mqtt_username	Mqtt_password
<input type="text" value="mosquittouser"/>	<input type="password" value="....."/>
Gateway_id	Ca_certs
<input type="text" value="1234567890098765"/>	<input type="text"/>
Network_address_1	Network_discovery_channel_1
<input type="text" value="2866082"/>	<input type="text" value="30"/>
Node_address_1	Authentication_key_1
<input type="text" value="98765432"/>	<input type="password" value="....."/>
Cipher_key_1	Node_role_1
<input type="password" value="....."/>	<input type="text" value="CSMA-CA Mode Sink"/>
Network_address_2	Network_discovery_channel_2
<input type="text" value="2866084"/>	<input type="text" value="10"/>
Node_address_2	Authentication_key_2
<input type="text" value="876543276"/>	<input type="password" value="....."/>
Cipher_key_2	Node_role_2
<input type="password" value="....."/>	<input type="text" value="CSMA-CA Mode Sink"/>
Output_power	Extra
<input type="text" value="FCC(20)"/>	<input type="text" value="--mqtt_reconnect_delay 20"/>

Apply

Figure 3, Dual Wirepas chip version gateway example (WP2)

Step 5. Check Sensor Data in the Wirepas Network Tool (WNT)

NETWORK	ADDRESS	NODE NAME	ROLE	POSITION ROLE	MODE	AUTO ROLE	BATTERY VOLTAGE	MEMORY ALLOCATION FAILURES	NC
2866072	23		Sink		Low latency	Off			
2866082	42		Sink		Low latency	Off			
2866072	2147483603		Router		Low energy	Off	2.80 V		
2866082	2147483601		Router		Low energy	Off	2.98 V		

Figure 4, WNT Tool, Nodes view

NETWORK	ADDRESS	TREE MEMBERS	MEMORY ALLOCATION FAILURES	NORMAL PRIORITY DROPPED PACKETS	MAX BUFFER USAGE
<input type="checkbox"/> 2866072	23	1			0.00 %
<input type="checkbox"/> 2866082	42	1			0.39 %

Figure 5, WNT Tool, Topology view

Please refer to link <https://developer.wirepas.com/support/solutions/articles/77000499190wirepas-network-tool-v4-client-user-guide#Nodes-View-Screen> for WNT details.

Troubleshooting

- **If a customer resets the gateway by mistake**

Please login to the local console and reconfigure the network configuration, review Step 1 and Step 2 above. Resetting the gateway will not remove or reset the Wirepas container APP.

- **If a customer deletes the container by mistake**

Please download container 2.0.1 and then re-install the container (https://www.cassianetworks.com/download/firmware/container/ubuntu_XE1000_2.0.1.tar.gz).

For container installation instructions, please check Chapter 5 of the Custom

(https://www.cassianetworks.com/download/docs/CassiaCustom_Application_Deployment_Instructions_v

- If a customer resets the container, or deletes the container App

Please download the Wirepas container App first (see link in Step 3), install the App again, and configure parameters according to Step 4. Check the gateway installation guide

(https://www.cassianetworks.com/download/docs/Cassia_Gateway_Installation_Guide.pdf).

Appendix A: Self-Signed Certificate Configuration for WNT4

1. Install Oracle VM VirtualBox 7.0

2. Install Ubuntu 20.04 on VM VirtualBox

Please refer to Wirepas Services Installer for WNT4 User Guide for details:

<https://developer.wirepas.com/support/solutions/articles/77000499208-wirepasservices-installer-for-wnt4-user-guide>

3. Install WNT backend version 4.3 in Ubuntu

o Copy the WNT backend package to Ubuntu scp <name of the installer>.tar.gz

Your Virtual Machine User name@ Your Virtual Machine Ip address:/home/Your Virtual Machine Username/

o Un compress the package tar -xzf <name of the installer>.tar.gz

o Edit ansible/setup_host.yml file with nano text editor (command given below):

- Set the username in the field "instance_admin_user" corresponding to your Ubuntu machine username. This user must have admin rights. A Ubuntu User usually has the correct rights to execute the Wirepas Service installer commands.
- Set 'self_signed' in the certificates field.
- No email is needed for this type of installation
- If missing the WPE license, set the field "wpe_version" to "None".

```
weijun@weijun:~/ansible/vars$ cat setup_host.yml
---
ansible_python_interpreter: /usr/bin/python3
instance_admin_user: "weijun" # your remote install admin user
ansible_ssh_private_key_file: "/.keys.pem" # private key to access remote
# If you wish to skip wnt or wpe installation set wnt/wpe_version to None or empty
wnt_version: "4.3"
wpe_version: "None"

# report and credentials
encrypt_local_files: false
wnt_storage: "{{playbook_dir}}/wnt" # wnt files will be copied here
wpe_storage: "{{playbook_dir}}/wpe" # wpe files will be copied here

# These variables are populated from your host name. They assume it
# follows the pattern: name.domain.country
instance_split: "{{inventory_hostname.split('.')}}"
instance_name: "{{instance_split[0]}}"
instance_dns_zone: "{{ '.'.join(instance_split[1:]) }}"
instance_domain_name: "{{inventory_hostname}}"

# Letsencrypt: Default solution installs cert automatically for your backend generated by letsencrypt.org
# Pregenerated means that you provide cert bundle which you have bought or generated yourself and placed in
# the path described in wnt_keychain: parameter
#
# Self signed certs are not recommended for production env.
# It is much faster and easier in terms of configuration to use trusted certificate from
# trusted certificate provider.
#
# It creates a new self-signed certificate to server and client and binds the created bundle.pem to
# wnt_keychain.
# With self signed backend you need to copy the client_bundle.pem from WNT backend into your WIN machine
# and add it as trusted root certificate authorities in Microsoft Management Console
# This enables WNT-Client to be able to connect to this this backend with self signed certificate.
# NOTE: you should modify playbooks/roles/certs/defaults/main.yml according to the your
# backend server and organization
#
# No cert means that backend is unsecure and no cert is needed. Not for production!
# valid choices are [letsencrypt, pregenerated, self_signed, no_cert]
certificate: "self_signed"
# your domain certificate to allow haproxy's TLS handshake
# letsencrypt and self_signed cert options overwrite this variable
wnt_keychain: "/extwirepasbundle.pem"

# this field is mandatory when letsencrypt certificate is selected. owner for the domain certificate
owner_email:
```

Figure 6, Configurations 1

o Edit vars/aws_account.yml with nano (# fill aws_access_key_id/aws_secret_access_key).

```
weijun@weijun:~/ansible/vars$ cat aws_account.yml
---
aws_cli_output: "text"
aws_region: "eu-west-1"

# Please fill these with your client credentials
aws_access_key_id: "AKIA..."
aws_secret_access_key: "..."
aws_ecr_account: "751834743493" # Wirepas ECR production account
aws_ecr_region: "eu-west-1"
aws_ecr_repository: "{{ aws_ecr_account }}.dkr.ecr.{{ aws_ecr_region }}.amazonaws.com"

# This parameter is not needed for Customer Selfhosting plays even if it remote would be installed in AWS
# not good idea to run against default any automation, therefore any profile will do
aws_profile: ""
weijun@weijun:~/ansible/vars$
```

Figure 7, Configurations 2

o sudo apt install -y ansible

o ansible-playbook --connection=local -i yourdomainforthisinstance, cli_setup_host.yml --tags wm-host --ask-become-pass -v

o (#Enter your user password, setup is done, machine reboots, reconnect with ssh and continue installation)

o cd ansible/

o ansible-playbook --connection=local -i your domain for this instance, cli_setup_host.yml --tags services -v

o docker ps (#ensure everything is running)

NOTE: Other files except for aws_account.yml and setup_host.yml , will use the Wirepas default configuration.

4. Download customer_report.rst and client_bundle.pem from WNT Backend

User should execute Windows command. Command shown as:

* scp username@instanceipaddress:/home/username/wnt/customer_report.rst destinationFolder

* scp username@instanceipaddress:/home/username/ansible/temp_certs_*/client_bundle.pem destinationFolder

5. WNT Client Windows configuration

o Importing certificate

When using a self-signed certificate and TLS, the WNT client will refuse to establish a connection. To allow a connection to be established, User must add the root certification authority (under cert_output_dir) to windows certificate store (client_bundle.pem). User can also enable this function from the Windows administrator command prompt with following command: certutil -addstore -f -enterprise -user root client_bundle.pem where the client_bundle.pem is the generated bundle file from temp_certs_* folder.

This enables the WNT-Client to be able to connect to the WNT backend with selfsigned certificate.

o Configure domain

Edit hosts in C:\Windows\System32\drivers\etc and add domain for WNT backend. i.e. 10.100.157.212 **exampleserver.anydomain.com**

6. Configure the Cassia Gateway Container

o SSH into Ubuntu Container of E1000 or X2000 as cassia user ssh -p 20022 cassia@gateway_ip

Password is cassia_xxxxxx (Gateway MAC address last 6 digits in lower case)

o vi /etc/hosts, add '10.100.157.212 **exampleserver.anydomain.com**'

root@ubuntu:~# cat /etc/hosts

127.0.0.1 localhost

127.0.1.1 ubuntu

The following lines are desirable for IPv6 capable hosts

::1 ip6-localhost ip6-loopback

fe00::0 ip6-localnet

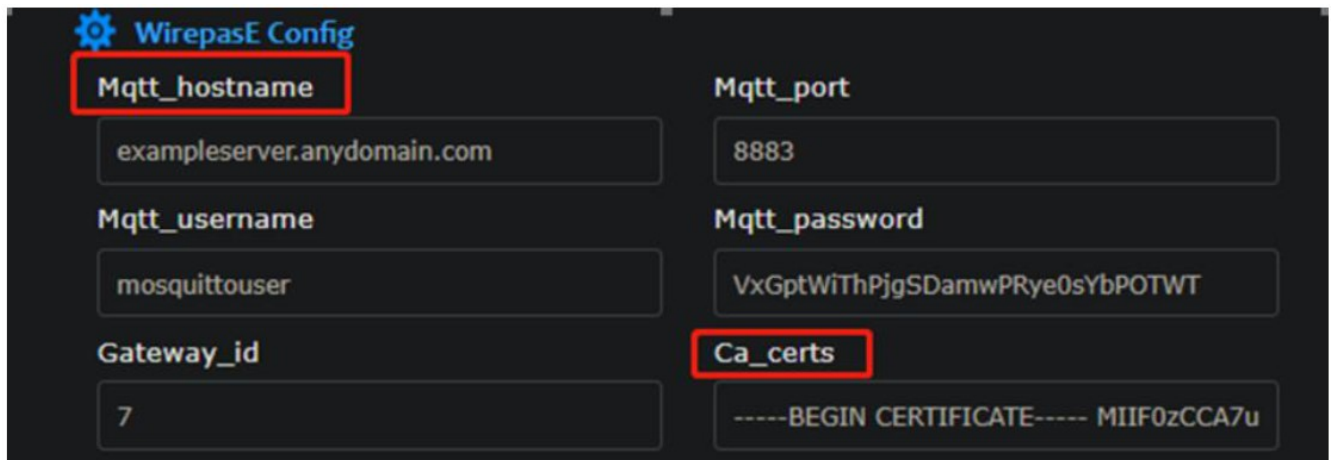
ff00::0 ip6-mcastprefix

ff02::1 ip6-allnodes

ff02::2 ip6-allrouters

10.100.157.212 **exampleserver.anydomain.com**

7. Configure Wirepas Parameters in the Cassia Container Tab

The image shows a 'WirepasE Config' window with a dark background. It contains several input fields for configuration. The 'Mqtt_hostname' field is highlighted with a red box and contains 'exampleserver.anydomain.com'. The 'Mqtt_port' field contains '8883'. The 'Mqtt_username' field contains 'mosquittouser'. The 'Mqtt_password' field contains 'VxGptWiThPjgSDamwPRye0sYbPOTWT'. The 'Gateway_id' field contains '7'. The 'Ca_certs' field is highlighted with a red box and contains '-----BEGIN CERTIFICATE----- MIIF0zCCA7u'.

Mqtt_hostname	Mqtt_port
exampleserver.anydomain.com	8883
Mqtt_username	Mqtt_password
mosquittouser	VxGptWiThPjgSDamwPRye0sYbPOTWT
Gateway_id	Ca_certs
7	-----BEGIN CERTIFICATE----- MIIF0zCCA7u

Mqtt_hostname **exampleserver.anydomain.com**

Ca_certs copy the content of client_bundle.pem into the box.

Appendix B: Pre-generated Certificate Configuration for WNT4

Configuration is the same with 'self_signed' certificate except the followings,

1. Generate certificate:

<https://gist.github.com/fntlnz/cf14feb5a46b2eda428e000157447309>

2. Upload certificate to WNT backend:

* Create a new directory with user with root permission

mkdir cert

* Upload certificate to the directory: scp bundle.pem

YourVirtualMachineUsername@YourVirtualMachineIpaddress:/home/YourVirtualMachineUsername/cert/.

3. Edit setup_host.yml file

* Set 'pregenerated' in the certificates field.

* Overwrite the Variable "wnt_keychain" with "/home/weijun/cert/bundle.pem".

```

weijun@weijun:~/ansible/vars$ cat setup_host.yml
---
ansible_python_interpreter: /usr/bin/python3
instance_admin_user: "weijun" # your remote install admin user
ansible_ssh_private_key_file: "/keys.pem" # private key to access remote

# If you wish to skip wnt or wpe installation set wnt/wpe_version to None or empty
wnt_version: "4.3"
wpe_version: "None"

# report and credentials
encrypt_local_files: false
wnt_storage: "{{playbook_dir}}/wnt" # wnt files will be copied here
wpe_storage: "{{playbook_dir}}/wpe" # wpe files will be copied here

# These variables are populated from your host name. They assume it
# follows the pattern: name.domain.country
instance_split: "{{inventory_hostname.split('.')}}"
instance_name: "{{instance_split[0]}}"
instance_dns_zone: "{{ '.'.join(instance_split[1:]) }}"
instance_domain_name: "{{inventory_hostname}}"

# Letsencrypt: Default solution installs cert automatically for your backend generated by letsencrypt.org
# Pregenerated means that you provide cert bundle which you have bought or generated yourself and placed in
# the path described in wnt_keychain: parameter
#
# Self signed certs are not recommended for production env.
# It is much faster and easier in terms of configuration to use trusted certificate from
# trusted certificate provider.
#
# It creates a new self-signed certificate to server and client and binds the created bundle.pem to
# wnt_keychain.
# With self signed backend you need to copy the client_bundle.pem from WNT backend into your WIN machine
# and add it as trusted root certificate authorities in Microsoft Management Console
# This enables WNT-Client to be able to connect to this this backend with self signed certificate.
# NOTE: you should modify playbooks/roles/certs/defaults/main.yml according to the your
# backend server and organization
#
# No cert means that backend is unsecure and no cert is needed. Not for production!

# valid choices are [letsencrypt, pregenerated, self_signed, no_cert]
certificate: "pregenerated"
# your domain certificate to allow haproxy's TLS handshake
# letsencrypt and self signed cert options overwrite this variable
wnt_keychain: "/home/weijun/ca/bundle.pem"

# this field is mandatory when letsencrypt certificate is selected. owner for the domain certificate
owner_email:

```

Figure 9, Setup_host

Appendix C: Configuration Letsencrypt Certificate for WNT4

WNT backend on cloud for Letsencrypt certificate are provided by Wirepas.

WirepasE Config

Mqtt_hostname <input type="text" value="cassia.prod-wirepas.com"/>	Mqtt_port <input type="text" value="8883"/>
Mqtt_username <input type="text" value="mosquitouser"/>	Mqtt_password <input type="password" value="....."/>
Gateway_id <input type="text" value="30"/>	Ca_certs <input type="text"/>
Network_address_1 <input type="text" value="2866072"/>	Network_discovery_channel_1 <input type="text" value="20"/>
Node_address_1 <input type="text" value="23"/>	Authentication_key_1 <input type="text"/>
Cipher_key_1 <input type="text"/>	Node_role_1 <input type="text" value="CSMA-CA Mode Sink"/>
Network_address_2 <input type="text" value="2866082"/>	Network_discovery_channel_2 <input type="text" value="10"/>
Node_address_2 <input type="text" value="42"/>	Authentication_key_2 <input type="text"/>
Cipher_key_2 <input type="text"/>	Node_role_2 <input type="text" value="CSMA-CA Mode Sink"/>
Output_power <input type="text" value="FCC(20)"/>	Extra <input type="text" value="--mqtt_reconnect_delay 20"/>

Apply

Cassia Networks





Figure 10, Gateway Configuration

Copyright © 2023 Cassia Networks, Inc.
Version: EN-20231205-YJ

Documents / Resources

<p>Cassia Wirepas Gateway Configuration Instructions</p> <p>Version: EN-20231205-YJ</p> <p>Contents</p> <ul style="list-style-type: none"> 1. Introduction 2. Getting Started 3. Gateway Configuration 4. Network Configuration 5. Node Configuration 6. Troubleshooting 7. Appendix 	<p>Cassia E1000-WP1 Wirepas Gateway [pdf] Instructions</p> <p>E1000-WP1 Wirepas Gateway, E1000-WP1, Wirepas Gateway, Gateway</p>
---	--

References

-  [Wirepas Network Tool v4 Client User Guide : Wirepas Developer Portal](#)
-  [Wirepas Network Tool v4 Client User Guide : Wirepas Developer Portal](#)
-  [Self Signed Certificate with Custom Root CA · GitHub](#)
-  [Wirepas](#)
- [User Manual](#)

[Manuals+](#), [Privacy Policy](#)

This website is an independent publication and is neither affiliated with nor endorsed by any of the trademark owners. The "Bluetooth®" word mark and logos are registered trademarks owned by Bluetooth SIG, Inc. The "Wi-Fi®" word mark and logos are registered trademarks owned by the Wi-Fi Alliance. Any use of these marks on this website does not imply any affiliation with or endorsement.