



Bluetooth[®] mesh SDK 2.2.1.0 GA

Gecko SDK Suite 4.0

January 26, 2022

Bluetooth mesh is a new topology available for Bluetooth Low Energy (LE) devices that enables many-to-many (m:m) communication. It's optimized for creating large-scale device networks, and is ideally suited for building automation, sensor networks, and asset tracking. Our software and SDK for Bluetooth development supports Bluetooth Mesh and Bluetooth 5.3 functionality. Developers can add mesh networking communication to LE devices such as connected lights, home automation, and asset tracking systems. The software also supports Bluetooth beaconing, beacon scanning, and GATT connections so Bluetooth mesh can connect to smart phones, tablets, and other Bluetooth LE devices.

These release notes cover SDK versions:

2.2.1.0 released January 26, 2022

2.2.0.0 released December 15, 2021



KEY FEATURES

- New example embedded Provisioner application
- NCP Commander support for Mesh
- Multiple improvements to the Mesh BGAPI
- Support for Amazon Bluetooth Mesh Simple Setup (BSS)

Compatibility and Use Notices

For more information about security updates and notices, see the Security chapter of the Gecko Platform Release notes installed with this SDK or on the [Silicon Labs Release Notes page](#). Silicon Labs also strongly recommends that you subscribe to Security Advisories for up-to-date information. For instructions, or if you are new to the Silicon Labs Bluetooth mesh SDK, see [Using This Release](#).

Compatible Compilers:

IAR Embedded Workbench for ARM (IAR-EWARM) version 8.50.9

- Using wine to build with the IarBuild.exe command line utility or IAR Embedded Workbench GUI on macOS or Linux could result in incorrect files being used due to collisions in wine's hashing algorithm for generating short file names.
- Customers on macOS or Linux are advised not to build with IAR outside of Simplicity Studio. Customers who do should carefully verify that the correct files are being used.

GCC (The GNU Compiler Collection) version 10.2.1, provided with Simplicity Studio.

Link-time optimization feature of GCC has been disabled, resulting in slight increase of image size

Contents

- 1 New Items2
 - 1.1 New Features.....2
 - 1.2 New APIs.....2
- 2 Improvements.....3
 - 2.1 Changed APIs.....3
- 3 Fixed Issues4
- 4 Known Issues in the Current Release5
- 5 Deprecated Items6
- 6 Removed Items7
- 7 Using This Release8
 - 7.1 Installation and Use.....8
 - 7.2 Security Information8
 - 7.3 Support.....9

1 New Items

1.1 New Features

Added in release 2.2.0.0

Support for Amazon Bluetooth Mesh Simple Setup (BSS) added and verified for compatibility.

The length of the GATT proxy filter list can now be configured in the application, instead of being a constant size.

Multiple small API improvements were made; details are in the following sections.

New Example Applications

A Provisioner NCP Host example was added to demonstrate how to create a mesh network, provision and configure nodes.

New Development Tools

Simplicity Studio's NCP Commander supports BT Mesh APIs and a GUI for provisioning and configuration.

Simplicity Studio displays short documentation for each example.

1.2 New APIs

Added in release 2.2.0.0

Diagnostic APIs for the provisioning process have been added. On the node side, the event **sl_btmesh_node_start_received()** is generated when the node receives the Provisioning Start PDU; the corresponding **sl_btmesh_prov_start_sent()** event is generated on the Provisioner side. Both events record the provisioning parameters the Provisioner has chosen to use for the provisioning session.

Added APIs to track vendor model message delivery by adding **sl_btmesh_vendor_model_send_tracked()** and **sl_btmesh_vendor_model_set_publication_tracked()** as well as **sl_btmesh_vendor_model_send_complete()** event. The APIs provide the application with a handle that can be matched with the event indication message that sending has completed. Note that the stack can only provide the information that the message has been sent; not whether a receiving application has received or processed the message. An application layer protocol message is needed for that.

Test API has been extended by multiple new calls: **sl_btmesh_test_set_beacon()**, **sl_btmesh_test_set_default_ttl()**, **sl_btmesh_test_set_friend()**, **sl_btmesh_test_set_gatt_proxy()**, **sl_btmesh_test_set_identity()**, **sl_btmesh_test_set_nettx()**, and **sl_btmesh_test_set_relay()** which replace the previously existing **sl_btmesh_test_set_local_config()**; and correspondingly **sl_btmesh_test_get_beacon()**, **sl_btmesh_test_get_default_ttl()**, **sl_btmesh_test_get_friend()**, **sl_btmesh_test_get_gatt_proxy()**, **sl_btmesh_test_get_identity()**, **sl_btmesh_test_get_nettx()**, and **sl_btmesh_test_get_relay()** which replace the previously existing **sl_btmesh_test_get_local_config()**. Finally, **sl_btmesh_test_get_model_option()** has been provided as a counterpart to **sl_btmesh_test_set_model_option()**.

Time server model API has been amended by the addition of **sl_btmesh_time_server_publish()** for unsolicited publishing of Time Status messages.

2 Improvements

2.1 Changed APIs

Changed in release 2.2.0.0

Defined the *status* parameter for `sl_btmesh_prov_set_provisioning_suspend_event()` to be a bitmask instead of just 0 or 1, in order to allow provisioning suspension at multiple points during the provisioning process.

Corrected the Provisioner APIs `sl_btmesh_prov_send_oob_pkey_response()` and `sl_btmesh_prov_send_oob_auth_response()` to contain the UUID of the device being provisioned, so that these APIs can be used when provisioning multiple devices in parallel.

3 Fixed Issues

Fixed in release 2.2.1.0

ID #	Description
739169	Fixed memory leak when initializing models multiple times instead of only once
748782	Improved proxy connection stability when transmitting Mesh advertisements at a quick pace
754910, 754931	Fixed issues with friend node handling of segmented messages
756963	Fixed missing message length check in Scene Setup server model
761990	Fixed an issue with non-default memory configuration for simultaneous GATT connections

Fixed in release 2.2.0.0

ID #	Description
418636	Fixed issues with mesh_test local configuration state API (node identity, relay, network retransmission)
736054	Added a check that the key used for publication is bound to the model
739523	Fixed an issue with inserting segmented multicast messages to the Friend queue
749981	Corrected parsing and generation of Config Network Transmit Set, Config Network Transmit Status, Config Relay Set, and Config Relay Status messages
752756	Fixed an issue with SAR timing configuration API
756361	Fixed Sensor model handling of long messages
756418	Fixed backward compatibility issue with device key storage when updating firmware from 1.x SDK
756629	Fixed issue with AID derivation of application keys that are generated from random data on Provisioner; does not affect AID derivation on nodes or keys that are generated from application-supplied data
732312	Bt Mesh Configurator no longer generates multiple macros for the same model if new elements are added

4 Known Issues in the Current Release

Issues in bold were added since the previous release.

ID #	Description	Workaround
401550	No BGAPI event for segmented message handling failure	Application needs to deduce failure from timeout / lack of application layer response; for vendor models an API has been provided
454059	A large number of key refresh state change events are generated at the end of KR process, and that may flood NCP queue	Increase NCP queue length in the project
454061	Slight performance degradation compared to 1.5 in round-trip latency tests was observed	
624514	Issue with re-establishing connectable advertising if all connections have been active and GATT proxy is in use	Allocate one more connection than is needed
650825	Issue with retransmissions when a model is publishing periodically	Set up retransmissions in the model state and trigger periodic publishing by an application timer
752802	No API for setting unprovisioned device beacon or GATT proxy service advertisement cadence	

5 Deprecated Items

Two test BGAPI class commands, `sl_btmesh_test_get_local_config()` and `sl_btmesh_test_set_local_config()`, have been deprecated. Replacements are listed in the API reference as well as in section [1.2 New APIs](#) of this document.

6 Removed Items

None

7 Using This Release

This release contains the following

- Silicon Labs Bluetooth mesh stack library
- Bluetooth mesh sample applications

If you are a first time user, see *QSG176: Silicon Labs Bluetooth Mesh SDK v2.x Quick-Start Guide*.

7.1 Installation and Use

The Bluetooth mesh SDK is provided as part of the Gecko SDK (GSDK), the suite of Silicon Labs SDKs. To quickly get started with the GSDK, install [Simplicity Studio 5](#), which will set up your development environment and walk you through GSDK installation. Simplicity Studio 5 includes everything needed for IoT product development with Silicon Labs devices, including a resource and project launcher, software configuration tools, full IDE with GNU toolchain, and analysis tools. Installation instructions are provided in the online [Simplicity Studio 5 User's Guide](#).

Alternatively, Gecko SDK may be installed manually by downloading or cloning the latest from GitHub. See https://github.com/SiliconLabs/gecko_sdk for more information.

The GSDK default install location has changed with Simplicity Studio 5.3, used with this release.

- Windows: C:\Users\\SimplicityStudio\SDKs\gecko_sdk
- MacOS: /Users/<NAME>/SimplicityStudio/SDKs/gecko_sdk

Documentation specific to the SDK version is installed with the SDK. Additional information can often be found in the [knowledge base articles \(KBAs\)](#). API references and other information about this and earlier releases is available on <https://docs.silabs.com/>.

7.2 Security Information

Secure Vault Integration

This version of the stack is integrated with Secure Vault Key Management. When deployed to Secure Vault High devices, mesh encryption keys are protected using the Secure Vault Key Management functionality. The table below shows the protected keys and their storage protection characteristics.

Key	Exportability on a node	Exportability on Provisioner	Notes
Network key	Exportable	Exportable	Derivations of the network key exist only in RAM while network keys are stored on flash
Application key	Non-exportable	Exportable	
Device key	Non-exportable	Exportable	In Provisioner's case, applied to Provisioner's own device key as well as other devices' keys

Keys that are marked as "Non-Exportable" can be used but cannot be viewed or shared at runtime.

Keys that are marked as "Exportable" can be used or shared at runtime but remain encrypted while stored in flash.

For more information on Secure Vault Key Management functionality, see [AN1271: Secure Key Storage](#)

Security Advisories

To subscribe to Security Advisories, log in to the Silicon Labs customer portal, then select **Account Home**. Click **HOME** to go to the portal home page and then click the **Manage Notifications** tile. Make sure that 'Software/Security Advisory Notices & Product Change Notices (PCNs)' is checked, and that you are subscribed at minimum for your platform and protocol. Click **Save** to save any changes.

SILICON LABS Search Within the Support Portal for Cases, etc... SEARCH CATHERIN...

HOME CASES SOFTWARE RELEASES

Update Preference

WHAT EMAILS WOULD YOU LIKE TO RECEIVE?

Newsletters

- Community Monthly Newsletter
- Sales Newsletter
- Micrium Newsletter

Product Specific Notifications

- Product Information and Newsletter
- Software/Security Advisory Notices & Product Change Notices (PCNs)
- Technical Document Updates (Release Notes, Data Sheets, etc.)

SELECT THE PRODUCTS TO RECEIVE UPDATES FOR

Select/Unselect All

<input type="checkbox"/> Audio and Radio	<input type="checkbox"/> Power over Ethernet
<input type="checkbox"/> Interface	<input type="checkbox"/> Sensors
<input type="checkbox"/> Isolation	<input type="checkbox"/> TV and Video
<input type="checkbox"/> Modems and DAAs	<input type="checkbox"/> Voice
<input type="checkbox"/> Microcontrollers	<input type="checkbox"/> Wireless
<input type="checkbox"/> 8-bit MCUs <input checked="" type="checkbox"/> 32-bit MCUs	<input type="checkbox"/> Bluetooth Classic <input type="checkbox"/> Bluetooth Low Energy <input checked="" type="checkbox"/> Proprietary
<input type="checkbox"/> Timing	<input type="checkbox"/> Wi-Fi
<input type="checkbox"/> Clocks	<input type="checkbox"/> ZigBee and Thread
<input type="checkbox"/> Buffers	<input type="checkbox"/> Z-Wave
<input type="checkbox"/> Oscillators	
<input type="checkbox"/> CDR and PHY	

7.3 Support

Development Kit customers are eligible for training and technical support. Use the [Silicon Labs Bluetooth mesh web page](#) to obtain information about all Silicon Labs Bluetooth products and services, and to sign up for product support.

Contact Silicon Laboratories support at <http://www.silabs.com/support>.

Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



IoT Portfolio
www.silabs.com/IoT



SW/HW
www.silabs.com/simplicity



Quality
www.silabs.com/quality



Support & Community
www.silabs.com/community

Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice to the product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Without prior notification, Silicon Labs may update product firmware during the manufacturing process for security or reliability reasons. Such changes will not alter the specifications or the performance of the product. Silicon Labs shall have no liability for the consequences of use of the information supplied in this document. This document does not imply or expressly grant any license to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any FDA Class III devices, applications for which FDA premarket approval is required or Life Support Systems without the specific written consent of Silicon Labs. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons. Silicon Labs disclaims all express and implied warranties and shall not be responsible or liable for any injuries or damages related to use of a Silicon Labs product in such unauthorized applications.

Note: This content may contain offensive terminology that is now obsolete. Silicon Labs is replacing these terms with inclusive language wherever possible. For more information, visit www.silabs.com/about-us/inclusive-lexicon-project

Trademark Information

Silicon Laboratories Inc.[®], Silicon Laboratories[®], Silicon Labs[®], SiLabs[®] and the Silicon Labs logo[®], Bluegiga[®], Bluegiga Logo[®], EFM[®], EFM32[®], EFR, Ember[®], Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Redpine Signals[®], WiSeConnect, n-Link, ThreadArch[®], EZLink[®], EZRadio[®], EZRadioPRO[®], Gecko[®], Gecko OS, Gecko OS Studio, Precision32[®], Simplicity Studio[®], Telegesis, the Telegesis Logo[®], USBXpress[®], Zentri, the Zentri logo and Zentri DMS, Z-Wave[®], and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. Wi-Fi is a registered trademark of the Wi-Fi Alliance. All other products or brand names mentioned herein are trademarks of their respective holders.



Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

www.silabs.com