Bluetooth® mesh SDK 2.1.3.0 GA

Gecko SDK Suite 3.2

September 24, 2021

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 5.2 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:

2.1.3.0 released September 24, 2021 (underlying Bluetooth changes only)
2.1.2.0 released September 8, 2021
2.1.1.0 released July 21, 2021
2.1.0.0 released June 16, 2021

Bluetooth

KEY FEATURES

- Integration with Secure Vault
- Addition of HSL models
- Support for dynamic GATT database
- Support for updated compiler vesions

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 10.2.0, provided with Simplicity Studio.

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

Contents

1 New Items

1.1 New Features

Added in release 2.1.0.0

Secure Vault Integration

Beginning with release 2.1.0.0, the Bluetooth Mesh SDK uses the Secure Vault Key Management functionality for storing mesh cryptographic keys when Secure Vault High devices are used.

The Secure Vault integration is visible to the customer in a number of ways on Series 2 devices:

- The layout of NVM3 data for cryptographic keys and their related metadata changes. Key migration functionality is provided for projects that have been created using SDK versions 2.0 or earlier. A one-time key migration needs to be done when the firmware on a device is updated.
- Key data visibility is intentionally limited on regular mesh nodes. An application on a regular mesh node is not permitted to view application or device key data using the `sl_btmesh_node_get_key()` BGAPI command, while an application on an embedded Provisioner node is allowed to do so.

For further information on key storage in Secure Vault please refer to AN1271: Secure Key Storage.

Compiler Support

Supported compilers have been updated to GCC version 10.2.0 and IAR version 8.50.9.

New Example Applications

HSL Lighting example (**Bluetooth Mesh - SoC HSL Light**) was added to demonstrate a light node that is controllable over HSL server models.

IOP demos (**Bluetooth Mesh - IOP Test - *)** were added for the Radio boards in the Pro Development Kits (SLWRB4104A, SLWRB4181A, SLWRB4181B, SLWRB4182A). The demos allow testing interoperability with mobile phones. The test requires four examples, each example representing one of the Mesh features: proxy, relay, friend, and LPN.

New Components

- HSL Server Component was added.
- Support for dynamic GATT database (a Bluetooth LE feature) was added.

1.2 New APIs

Added in release 2.1.2.0

By default a normal Mesh device that is not a Provisioner is not able to export security key data over the BGAPI. If key export is needed on such a device it should use a new BGAPI command, **sl_btmesh_node_set_exportable_keys()**, before any keys are created on the node. This includes keys created during provisioning of the device.

A diagnostic event notifying the application of a scheduled scene change, **sl_btmesh_scheduler_server_scene_changed()**, has been added.

Added in release 2.1.1.0

To optimize buffer usage with Scene models, an optional API to enable compacted scene recall events has been added (ref. issue ID 706555). It is recommended to use the new API when a node has a large amount of models, or the amount of network traffic the node is expected to hear is high.

To activate the new API, use the BGAPI command **sl_btmesh_scene_server_enable_compact_recall_events()**. Afterwards, **sl_btmesh_evt_scene_server_compact_recall_events** will signal scene recall requests. To retrieve the cached model states after a scene recall request, use the command **sl_btmesh_generic_server_get_cached_state()**.

Added in release 2.1.0.0

Because of Secure Vault integration, the details of storing encryption keys and their related metadata has changed on Series 2 devices. A new BGAPI class for migrating encryption keys and the embedded Provisioner's device database after a firmware update on Series 2 devices has therefore been added. It has the following commands:

- sl_btmesh_migration_migrate_keys
- sl_btmesh_migration_migrate_ddb

2 Improvements

2.1 Changed APIs

Changed in release 2.1.2.0

The timezone parameter in **sl_btmesh_time_server_get_datetime()** has been corrected to be a signed 16-bit integer.

Clock accuracy parameter, **sl_btmesh_lpn_clock_accuracy**, has been added to LPN configuration. This parameter can be used to tune LPN sleep behavior when the clock drift on the device would otherwise cause LPN to miss its poll timeout.

Changed in release 2.1.1.0

The event **sl_btmesh_evt_friend_friendship_terminated** will now be generated when a Configuration Client disables the node's Friend feature when a friendship is active. Previously the termination of the friendship in this situation was implicitly signalled by the **sl_btmesh_evt_node_config_set** event. (Ref. issue iD 627811)

Changed in release 2.1.0.0

The following BGAPI commands in the prov class now return after parameter validation, and the actual requested operation takes place after a BGAPI response is given. The completion of the requested operation is signalled by a corresponding BGAPI event:

- **sl_btmesh_prov_add_ddb_entry()** – completion of addition is signalled by sl_btmesh_evt_prov_add_ddb_entry_complete
- **sl_btmesh_prov_delete_ddb_entry()** – completion of deletion is signalled by sl_btmesh_evt_prov_delete_ddb_entry_complete

The following BGAPI command in the prov class has an additional event that may be generated after it is called:

sl_btmesh_prov_init – in addition to sl_btmesh_evt_prov_initialized, sl_btmesh_evt_prov_initialization_failed may be generated.

A BGAPI command has been added to the generic client model BGAPI:

mesh_generic_client_init_hsl()

A BGAPI command has been added to the generic server model BGAPI:

mesh_generic_server_init_hsl()

3 Fixed Issues

Fixed in release 2.1.2.0

ID #	Description
627811	Generate a friendship terminated event when termination is locally requested
676798	Take clock inaccuracy into account with LPN poll wakeup timing
683518	Generate friendship termination event immediately at the time of receiving a Friend Clear message
703974	Fixed a qualification test issue with heartbeats
709948	Provided an API to control exportability of security keys on a mesh node
724511	Fixed an issue with registering vendor opcodes over 0x1F
730273	Fixed an issue with negative time zone offset handling
731713	Fixed a potential memory leak with sending segmented messages when the device is low on memory
734034	Fixed friend-to-LPN communication when TTL is zero
734858	Corrected a potential stack variable issue with PSA structure handling
736054	Fixed a qualification test issue with model-application key binding

Fixed in release 2.1.1.0

ID #	Description
692961	Fixed the node becoming unresponsive when relay retransmissions were enabled when under heavy load
713152	Fixed a problem where limited precision of calculation caused rounding errors in the binding between Light Lightness Actual and Light Lightness Linear

Fixed in release 2.1.0.0

ID #	Description
3878	Application should ignore GATT events for Mesh characteristics
342521	Math library does not grow the size of the image unnecessarily
358019	Corrected result code given when model publication with friendship credentials is requested but friendship is not supported
404070	Corrected result code given when provisioner command to create a network key is called on a non-provisioner device
454332	The LE GAP API should be used for device local name advertising
464907	Removed unnecessary ‘heartbeat started’ BGAPI event when configuration client disables heartbeat on a node
653405	Out-of-the-box switch sample application current consumption is now at the expected level
654477	DCD correctly decoded by Network Analyzer
660048	Button Press UC component doesn’t require IO Stream component unneccesarily
687105	BT Mesh Commands work with NCP Target example and NCP Commander
690803	Fixed duplicate vendor model IDs in code generator
690862	SoC empty example now starts beaconing on xG22 hardware
707497	Corrected PSA cryptographic context allocation
707524	Fixed a regression with IV recovery guard timer, not allowing another recovery too soon
710381	Fixed lighting default state handling when a non-default range for the corresponding model was set
711359	Fixed parameter checking for provisioning session creation BGAPI call

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
418636	Issues with mesh_test local configuration state API (node identity, relay, network retransmission)	
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

5 Deprecated Items

The following BGAPI command in the node class has been deprecated:

sl_btmesh_node_erase_mesh_nvm() – use sl_btmesh_node_reset() instead.

6 Removed Items

None

7 Using This Release

This release contains the following

- Silicon Labs Bluetooth mesh stack library
- Bluetooth 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

A registered account at Silicon Labs is required in order to download the Silicon Labs Bluetooth SDK. You can register at https://siliconlabs.force.com/apex/SL_CommunitiesSelfReg?form=short.

Stack installation instruction are covered in *QSG176: Silicon Labs Bluetooth Mesh SDK v2.x Quick-Start Guide*.

Use the Bluetooth mesh SDK with the Silicon Labs Simplicity Studio V4 development platform. Simplicity Studio ensures that most software and tool compatibilities are managed correctly. Install software and board firmware updates promptly when you are notified.

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 Provisionerr's own device key as well as other devices' keys

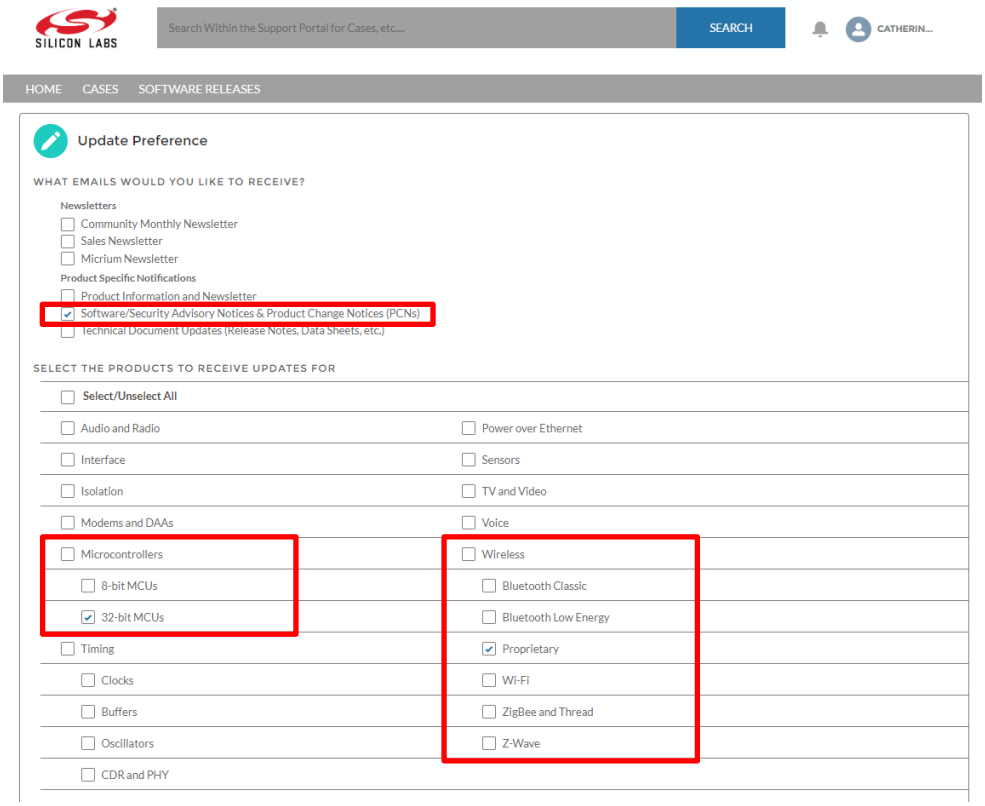
Keys that are marked as "Non-Exportable" can be used but cannot be viewed or shared at runtime.

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.



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

SILICON LABS

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

www.silabs.com