Bluetooth® LE SDK 9.1.0.0 GA
Simplicity SDK Suite 2024.12.2
April 1, 2025

Silicon Labs is a leading vendor in Bluetooth hardware and software technologies, used in products such as sports and fitness, consumer electronics, beacons, and smart home applications. The core SDK is an advanced Bluetooth 5.4-compliant stack that provides all of the core functionality along with multiple API to simplify development. The core functionality offers both standalone mode, allowing a developer to create and run their application directly on the SoC, or in NCP mode allowing for the use of an external host MCU.

These release notes cover SDK version(s):

9.1.0.0 GA released April 1, 2025
9.0.1.0 GA released February 5, 2025
9.0.0.0 GA released December 16, 2024

Bluetooth

KEY FEATURES

Bluetooth

- GA release of Periodic Advertisement BGAPI event
- BT LE Connection Subrating
- Accept List Based Auto-connect
- BT Atlanta (v6.0) LL and Host Qualification
- Channel Sounding sparse channel map support
- Channel Sounding antenna switching support
- CBAP - CPMS integration

Multiprotocol

- ZigbeeD and OTBR support on OpenWRT – GA
- DMP BLE + CMP ZB & Matter/OT with Concurrent Listening on MG26 for SoC – GA
- 802.15.4 Unified radio scheduler priority component
- Debian packaging support for MP host applications - Alpha

Compatibility and Use Notices

For information about security updates and notices, see the Security chapter of the Platform Release Notes installed with this SDK or on the TECH DOCS tab on https://www.silabs.com/developers/bluetooth-low-energy. Silicon Labs also strongly recommends that you subscribe to Security Advisories for up-to-date information. For instructions as well as notes on using Secure Vault features, or if you are new to the Silicon Labs Bluetooth SDK, see Using This Release.

Compatible Compilers:

IAR Embedded Workbench for ARM (IAR-EWARM) version 9.40.1.

- 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 12.2.1, provided with Simplicity Studio.

Contents

1 New Items

1.1 New Features

Added in release 9.1.0.0

GATT Client for ATT MTU Exchange Only

Added component bluetooth_feature_gatt_client_att_mtu_request_only. This component provides a minimal GATT Client to automatically initiate an ATT MTU exchange procedure when the GATT connection is open. This component does not provide the GATT Client API. Use the GATT Server API sl_bt_gatt_server_set_max_mtu to set the maximum size of ATT MTU in the BLE Host Stack.

Components for Specific Connection Roles

Added new components bluetooth_feature_connection_role_central and bluetooth_feature_connection_role_peripheral. These components provide support for a specific connection role. When an application includes bluetooth_feature_connection, the application should also include one or both of the role-specific components based on the application's needs. If the application includes only bluetooth_feature_connection, both connection roles will be supported for backwards compatibility.

Better Code Optimization in Bluetooth Security Manager

The Bluetooth security manager now automatically drops the central or peripheral state machine if either the bluetooth_feature_connection_role_central or the bluetooth_feature_connection_role_peripheral component, respectively, is not included, in the application.

Added in release 9.0.0.0

TX power higher than 10 dBm in low power mode

Using TX power higher than 10 dBm in low power mode is supported in NCP and SoC modes. The power limits in low power mode can be configured using **sl_bt_system_linklayer_configure()** with key **sl_bt_system_linklayer_config_key_low_power_mode_power_limit**.

New scanner option

Added a new scanner option, SL_BT_SCANNER_IGNORE_BONDING, for use with **sl_bt_scanner_set_parameters_and_filter c**ommand. If the application does not need the bonding information in advertisement reports, it can set this scanner option to avoid unnecessary searching of the bondings.

New custom address option

Added a new option, SL_BT_CONFIG_SET_CUSTOM_ADDRESS_FROM_NVM3, for configuring whether the stack should use a custom address stored in the defined key in Bluetooth region of NVM3 as the device identity address. By default, this option is enabled.

Event System IPC support for Bluetooth events

The new optional component, bluetooth_event_system_ipc, provides support for getting Bluetooth events via the Event System mechanism in an application that uses an RTOS.

Connection Subrating

The new optional component, bluetooth_feature_connection_subrate, provides Bluetooth Connection Subrating feature. The feature is provided in experimental quality in this release.

LTO support in host stack libraries

Selected host stack GCC libraries are built with LTO options (-flto, -ffat-lto-objects). This enables better code size optimization if the application uses LTO.

HCI Event Filtering

Allows defining custom event filters for HCI event processing. The filter callback is called before the event is sent to the host stack. This can be used to limit unnecessary traffic over an HCI connection.

Extended RTOS support

Added RTOS variant of multiple existing example applications.

ESL: Accept-List based Auto-Connect

Added a new connection method to the Electronic Shelf Label (ESL) Access Point (AP) to speed up large network setup.

ESL: Re-synchronisation by scanning

Tags can now attempt scan-based re-synchronization on sync loss before starting to advertise (vendor specific, opt-in enhancement of ESL Tag Core component) This feature also requires the AP to advertise the PAwR parameters.

ESL: preset network configuration

The ESL AP can import and export network sessions to and from JSON format describing ESL groups and IDs. After importing a complete network configuration to be used by the auto mode for automatic addressing, an exclusive mode can also be set to discard any nearby advertising ESLs that are not in the configuration.

The ESL AP can continue a previous network session after a power cycle by reconfiguring the same devices with the same ESL ID in the same group as before.

ESL: extended key library

The functionality of the ESL AP Key Library Python class has been extended to utilize the ESL Address, Response Key Material, and bonded AP fields of the key database.

CS Initiator: configurable channel map

The channel map of the CS Initiator example is now configurable.

BRD2608A Development Kit:

BRD2608A Dev Kit example application now supports the IMU sensor.

1.2 New APIs

Added in release 9.0.0.0

sl_bt_gap_get_identity_address() command: Get the Bluetooth identity address used by the device.

sl_bt_gatt_read_variable_length_characteristic_values() command: Read multiple variable-length characteristic values from a remote GATT server.

sl_bt_gatt_server_read_attribute_properties() command: Read the properties of an attribute from local GATT database.

sl_bt_gattdb_get_attribute_state() command: Get the attribute state from local GATT database when using the dynamic GATT database feature.

sl_bt_gatt_server_find_primary_service() command: Find primary services with UUID from local GATT database.

sl_bt_connection_set_default_acceptable_subrate() command: Set the default acceptable parameters for subrating requests.

sl_bt_connection_request_subrate() command: Request a change to the subrating factor and other parameters.

sl_bt_evt_connection_subrate_changed event: Report the completion of subrating procedure or subrating parameter changes on a connection.

sl_bt_evt_connection_request_subrate_failed event: Report that a subrating procedure failed.

sl_bt_evt_periodic_advertiser_status event: Report the latest status of periodic advertising on an advertising set.

sl_bt_system_linklayer_config_key_set_periodic_advertising_status_report (0x8): New key to **sl_bt_system_linklayer_configure()** to enable or disable the status report of periodic advertising.

2 Improvements

2.1 Changed Items

Changed in release 9.0.0.0

ID #	Description
1233899	Selected Bluetooth host stack libraries are compiled with LTO options (-flto -ffat-lto-objects) for allowing better dead code elimination in applications.
1234000	Certificate fields have been updated for Certificate Based Authentication and Pairing. Database added for issued certificates.
1298645	Changed returned error codes to use the actual values from NVM3 when accessing it using BGAPI.
1318468	Certificate Based Authentication and Pairing is now supported on xG22 devices.
1321901	Object tracking mode 'stationary object tracking' can now be selected in bt_cs_host.
1329672	Released the CBAP library source. CBAP refactoring enabled. Connection handling improved.
1332939	A bonding confirmation event is sent even when the connection has already been encrypted and pairing request is received from the central device.
1334523	BLE host stack can now operate without the presence of NVM3. To drop NVM3 from a Bluetooth application, the application must not use bluetooth_feature_builtin_bonding_database, bluetooth_feature_nvm, or bluetooth_feature_sm components.
1324517	Added link layer configuration flag to report number of transmitted packets in direct test modes.

2.2 Changed APIs

Changed in release 9.0.0.0

None.

2.3 Intended Behavior

Changed in release 9.0.0.0

None.

3 Fixed Issues

Fixed in release 9.1.0.00

ID #	Description
1404920	Fixed an issue where a macro to reset the terminal colors was defined but not used. **_app_log_reset_color** was defined but not used in app_log.h This fix adds it to the end of macros that use **_app_log_print_color**.
1405476	Fixed an issue that prevented the NCP from initializing properly when BGAPI message encryption is enabled.
1417581	Fixed init time issue when one advertising handle is used by both legacy and extended advertisements. This caused a race condition which stopped the advertiser prematurely.

Fixed in release 9.0.1.0

ID #	Description
1381647	In Channel Sounding (CS) multi-connection use case, where the reflector is running CS procedures with more than one initiator, the reflector sometimes selected parameters that would lead to procedures overlap. This has been fixed.
1355908	Periodic Advertisement did not work properly in dynamic multiprotocol with OpenThread. This has been fixed.
1383315	Unused bluetooth_feature_extended_scanner component was removed from ESL AP NCP.
1388519	Fixed a regression in the built-in battery level measurement of the ESL Tag that was preventing the correct value from being measured.
1393811	Fixed a regression in the QR code generator for the ESL demo that was introduced with Pillow v11.x Python module update.

Fixed in release 9.0.0.0

ID #	Description
845506	The adaptivity of AFH can be disabled or enabled with command **sl_bt_system_linklayer_configure()** and key **sl_bt_system_linklayer_config_key_set_channelmap_flags.**
1082103, 1141041, 1212061	Using TX power higher than 10 dBm in low power mode is supported in NCP and SoC modes. The power limits in low power mode can be configured using **sl_bt_system_linklayer_configure()** key **sl_bt_system_linklayer_config_key_low_power_mode_power_limit.**
1284611	Clarified in the API documentation that starting a DTM transmitter or receiver test while other Bluetooth activities are active will either result in an error response or will cause functionality issues. The API documentation now suggests that the application should make sure other Bluetooth activities are stopped when performing a test.
1328923	Fixed an issue in dynamic GATT database feature that, after adding a new descriptor to a GATT characteristic that has been enabled to be visible to remote GATT clients, the remote GATT client cannot see the new descriptor.
1335919	Make bluetooth_feature_advertiser component configurable on Simplicity Studio component browser GUI.
1349058	Fixed an issue which allowed security properties to downgrade during re-pairing. Now the security properties must match or exceed ones used during previous pairing.
1356037	**sl_bt_nvm_save(), sl_bt_nvm_load(),** and **sl_bt_nvm_erase()** now properly validate that a specified NVM3 key is in the range assigned for user data. The key range for user data stored in the Bluetooth region of NVM3 is extended to 0x4000 - 0x5fff.
1371005	Fixed an issue in the Bluetooth LE linklayer, where an advertising device that is sending out connectable extended advertisements replies to an AUX_CONN_REQ with an invalid AUX_CONN_RSP. This issue happened when the user set a random address to the extended advertising set only.
1362681	Fixed PAwR subevent_start. It was not incrementing correctly.
1336266	Fixed an issue with **sl_bt_advertiser_set_timing()** not working properly with higher than 16000 (10sec). Interval_max.

ID #	Description
1330263	Fixed an issue in the Bluetooth LE link-layer that caused the PAwR advertiser to stop accepting subevent data setting from the host.

4 Known Issues in the Current Release

Issues in bold were added since the previous release. If you have missed a release, recent release notes are available on https://www.si-labs.com/developers/bluetooth-low-energy in the Tech Docs tab.

ID #	Description	Workaround
361592	The sync_data event does not report TX power.	None
368403	If setting CTE interval to 1, a CTE request should be sent in every connection interval. But it is sent only in every second connection interval.	None
641122	The Bluetooth stack component does not provide a configuration for RF antenna path.	This is an issue specifically for BGM210P. One workaround is to manually update the configuration in sl_bluetooth_config.h in text edit mode.
682198	The Bluetooth stack has an interoperability issue on the 2M PHY with a Windows PC.	No workaround exists. For application development and testing, the disconnection can be avoided by disabling 2M PHY with sl_bt_connection_set_preferred_phy() or sl_bt_connection_set_default_preferred_phy().
756253	The RSSI value on a Bluetooth connection returned by the Bluetooth API is incorrect on EFR32M|B21 devices. It is about 8~10 dBm higher than the actual value, according to a measurement.	Install the "RAIL Utility, RSSI" component in the application project. This component provides a default RSSI offset for the chip that is applied at the RAIL level and can help to achieve more accurate RSSI measurements.
1031031	Changing the configuration in the bt_aoa_host_locator application results in the application crashing.	None
1334418 1335263	Channel sounding minor jitter issue when transmitting RTT sync packets on the initiators side. This might be visible when doing channel sounding to some other vendor devices.	None
1373310	RTOS Priorities are not set correctly in SoC/NCP case if not using dynamic multiprotocol. This might cause radio priorities to be blocked by less important priorities.	Radio priorities need to set radio priorities in the application software.
1383864	Scan event may contain additional garbage bytes in a noisy environment, where lots of BLE devices advertise at the same time.	None
1397074	In the Channel Sounding use case, if the unsupported Free-Running mode is selected in the configuration header file, some connections will be closed by the application and re-established.	Set the maximum procedure count to 0.
1399177	In case of Channel Sounding multi-connection scenario, where the reflector is connected to more than one initiator, some connections may close due to operation timeout.	None

5 Deprecated Items

Deprecated in release 9.0.0.0

Overriding the HFXO CTUNE with a value in the NVM3 Bluetooth region is deprecated.

By default, CTUNE override using NVM3 Bluetooth region is disabled. Enable it with the SL_BT_CONFIG_SET_CTUNE_FROM_NVM3 configuration.

Since the Simplicity SDK 2024.12.1 release, the alternative of CTUNE override is to use the platform component clock_manager_oscillator_calibration_override. This platform component supports both the HFXO and LFXO CTUNE overrides.

6 Removed Items

Removed from release 9.0.1.0

ID #	Description
1382948	Bluetooth RTOS examples are no longer supported on xG22 devices.

Removed from release 9.0.0.0

sl_bt_connection_get_rssi

sl_bt_rtos_has_event_waiting

sl_bt_rtos_event_wait

sl_bt_rtos_get_event

sl_bt_rtos_set_event_handled

Parameter ‘txsize’ of event sl_bt_evt_connection_parameters

7 Multiprotocol Gateway and RCP

7.1 New Items

Added in release 9.1.0.0

The zigbee_throughput plugin start command now includes an optional uint8_t argument “plugin throughput start 0” which will not clear the stack counters before a throughput test begins. This is intended for testing purposes. If no additional arugment is included and/or not 0, the current behavior remains the same and will clear the device counters when a throughput test begins.

Added in release 9.0.0.0

Enabled GA SoC support for BLE DMP with Zigbee + Openthread CMP with concurrent listening on xG26 parts.

Debian alpha support has been added for zigbeed, OTBR and Z3Gateway applications. Zigbeed and OTBR are provided in DEB package format for the chosen reference platform (Raspberry PI 4) as well. See *Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol Co-Processor*, for details.

Added Zigbeed support for Tizen-0.1-13.1 for arm32 and aarch64 as well as Android 12 for aarch64. More information on Zigbeed can be found at docs.silabs.com.

Added the new “802.15.4 Unified radio scheduler priority” component. This component is used to configure the radio priorities of a 15.4 stack. The component also requires the new “radio_priority_configurator” component. This component allows projects to use the Radio Priority Configurator tool in Simplicity Studio to configure the radio priority levels of the stacks that require it.

7.2 Improvements

Changed in release 9.1.0.0

The Zigbee-NCP + OpenThread-RCP (UART & SPI) samples, as well as the Zigbee-NCP + BLE-NCP (UART & SPI) samples, are now only allowed for generation on parts with sufficient RAM (>=96kB).

Changed in release 9.0.1.0

The Zigbee BLE - DynamicMultiprotocolLightSed sample project can now be built for boards with only one LED if the LED1 component is excluded from the project.

This revision has the following improvements for multi-protocol applications when, for example, running Zigbee or OpenThread for the custom use-case of running one protocol at a time:

- The sl_zigbee_af_zll_unset_factory_new() API has been added to allow applications to unset a Zigbee node from the default factory new state when necessary.
- A callback, sl_rail_mux_invalid_rx_channel_detected_cb(), has been added to Zigbee+OT applications. This callback notifies the application when there has been an attempted RX on two different channels while concurrent listening is not enabled. The application can then implement its own logic to handle this situation.

Changed in release 9.0.0.0

Application note *Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol Co-Processor* (AN1333) has been moved to docs.silabs.com.

OpenWRT support is now GA quality. OpenWRT support has been added for zigbeed, OTBR and Z3Gateway applications. Zigbeed and OTBR are provided in IPK package format for the reference platform (Raspberry PI 4) as well. See *Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol Co-Processor*, for details.

7.3 Fixed Issues

Fixed in release 9.1.0.0

ID #	Description
1392015	SL_OPENTHREAD_ENABLE_SERIAL_TASK is now set to 0 by default to reduce the task memory size which is not required for RCP applications. (Other ref: 1424440)
1393057	Fixed an issue where the Zigbee-NCP + OpenThread-RCP (UART & SPI) samples, as well as the Zigbee-NCP + BLE-NCP (UART & SPI) samples, were allowed for generation on parts with insufficient RAM.
1399687	Fixed an issue where the Zigbee-BLE DMP Light app may fail to go to EM2 sleep.
1420018	Fixed an issue where a CPC message from an RTOS-enabled RCP to host could be waiting in a send queue until something woke up the serial task.

Fixed in release 9.0.1.0

ID #	Description
1363050	Zigbee stack initialization no longer activates the radio (or RCP for host stacks) prior to stack APIs being called by the application. This prevents unwanted multi-PAN operation on Channel 11 (the default channel) when using a multi-PAN-capable RCP configuration. (Other ref: 1390724)
1378298	Fixed an issue that caused a crash when entering “keys print” on an DMP Light app with LTO enabled.
1381165	Fixed an issue on Zigbee-NCP + OT-RCP, when disabling PTA would have caused NCP/RCP reset.

Fixed in release 9.0.0.0

ID #	Description
1275378	Fixed an issue where calling sl_802154_radio_set_scheduler_priorities() prior to sli_mac_lower_mac_init() could result in a crash.
1300848	Fixed an issue where Z3Gateway in OpenWRT environment couldn't start EZSP communication caused by mismatching termios control characters running on OpenWRT and other environments.
1332330	Fixed an issue where a 15.4+BLE RCP operating in an environment with heavy network traffic could occasionally encounter a race condition that would leave it unable to send messages up to CPCd until rebooting the device.
1337101	Incomplete 15.4 transmit operations (Tx waiting for an ack, Tx an ack in response to a message, etc) are no longer prematurely considered as failed upon radio interruption due to DMP. This allows said operation to be given a chance to be rescheduled after the interruption or permanently failed by RAIL (scheduler status error events).
1337228	In Zigbeed the halCommonGetInt32uMillisecondTick() tick API is now updated to use MONOTONIC clock, so that it does not get affected by the NTP in a host system.
1337295	The DMP CLI command “plugin ble gap print-connections” will now print “No BLE connections” if the connection table is empty, instead of providing no response.
1346785	Fixed a race condition which could cause concurrent listening to be disabled on the 802.15.4 RCP when both protocols were transmitting simultaneously.
1346849	Adding the rail_mux components to a project will now cause it to automatically build with the associated stack library variants.
1365665	Fixed an issue where the host would report receiving a packet with an invalid checksum on end-point 12.

7.4 Known Issues in the Current Release

Issues in bold were added since the previous release. If you have missed a release, recent release notes are available on https://www.silabs.com/developers/simplicity-software-development-kit.

ID #	Description	Workaround
937562	Bluetoothctl 'advertise on' command fails with rcp-uart-802154-blehci app on Raspberry Pi OS 11.	Use btmgmt app instead of bluetoothctl.
1074205	The CMP RCP does not support two networks on the same PAN id.	Use different PAN ids for each network. Support is planned in a future release.
1122723	In a busy environment, the CLI may become unresponsive in the z3-light_ot-ftd_soc app.	No known workaround.
1209958	The ZB/OT/BLE RCP using concurrent listening on MG24 and MG21 may stop working in endurance test (lasts ~2 hours) with constant and concurrent traffic on all 3 stacks.	Disable concurrent listening in use cases involving constant and concurrent traffic across all 3 protocols.
1221299	Mfglib RSSI readings differ between RCP and NCP.	Will be addressed in a future release.
1385052	**Coex-enabled RCP may still occasionally transmit TX ACK after losing the Grant even when Acking is disabled and TX Abort is enabled.**	**Will be addressed in a future release.**
1385486	**TX from RCP may infrequently happen without the request after turning on the non-802.15.4 compliant MAC Holdoff coex option.**	**Will be addressed in a future release.**

7.5 Deprecated Items

The "Multiprotocol Container" which is currently available on DockerHub (siliconlabsinc/multiprotocol) will be deprecated in an upcoming release. The container will no longer be updated and able to be pulled from DockerHub. The Debian-based packages for cpcd, zigbeed, and ot-br-posix, along with natively generated and compiled projects, will replace the functionality lost with the removal of the container.

7.6 Removed Items

Removed in release 9.0.1.0

sl_sec_man_init() has been removed, since it no longer serves a purpose.

Removed in release 9.0.0.0

None.

8 Using This Release

This release contains the following

- Silicon Labs Bluetooth stack library
- Bluetooth sample applications

For more information about the Bluetooth SDK see https://docs.silabs.com/bluetooth/latest/ . If you are new to Bluetooth see UG103.14: Bluetooth LE Fundamentals.

8.1 Installation and Use

The Bluetooth SDK is provided as part of the Simplicity SDK, the suite of Silicon Labs SDKs. To quickly get started with the Simplicity SDK, 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, Simplicity SDK may be installed manually by downloading or cloning the latest from GitHub. See https://github.com/SiliconLabs/simplicity_sdk for more information.

Simplicity Studio installs the Simplicity by default in:

- (Windows): C:\Users\<NAME>\SimplicityStudio\SDKs\simplicity_sdk
- (MacOS): /Users/<NAME>/SimplicityStudio/SDKs/simplicity_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/.

8.2 Security Information

Secure Vault Integration

When deployed to Secure Vault High devices, sensitive keys such as the Long Term Key (LTK) are protected using the Secure Vault Key Management functionality. The table below shows the protected keys and their storage protection characteristics.

Wrapped Key	Exportable / Non-Exportable	Notes
Remote Long Term Key (LTK)	Non-Exportable	
Local Long Term Key (legacy only)	Non-Exportable	
Remote Identity Resolving Key (IRK)	Exportable	Must be Exportable for future compatibility reasons
Local Identity Resolving Key	Exportable	Must be Exportable because the key is shared with other devices.

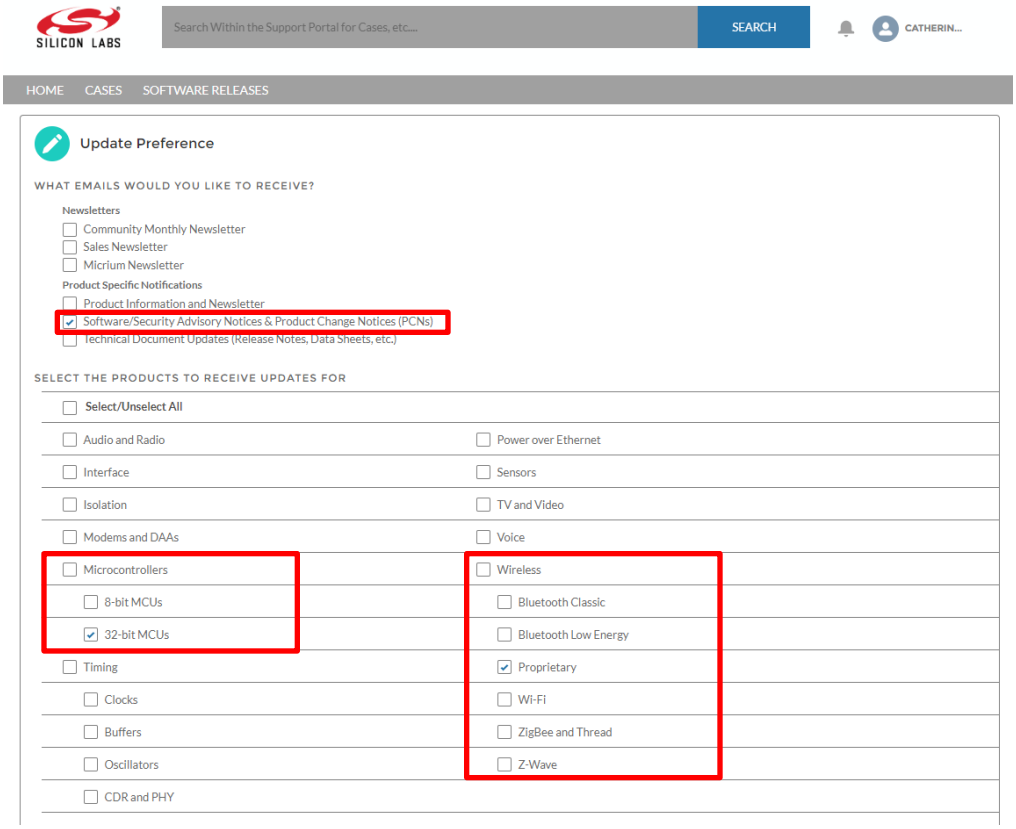
Wrapped keys that are marked as "Non-Exportable" can be used but cannot be viewed or shared at runtime.

Wrapped 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.



8.3 Support

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

You can contact Silicon Laboratories support at http://www.silabs.com/support.

8.4 SDK Release and Maintenance Policy

For details, see SDK Release and Maintenance Poilcy.

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