Silicon Labs OpenThread SDK 2.6.2.0 GA

Simplicity SDK Suite 2024.12.2

April 1, 2025

Thread is a secure, reliable, scalable, and upgradeable wireless IPv6 mesh networking protocol. It provides low-cost bridging to other IP networks while optimized for low-power / battery-backed operation. The Thread stack is designed specifically for Connected Home applications where IP-based networking is desired, and a variety of application layers may be required.

OpenThread released by Google is an open-source implementation of Thread. Google has released OpenThread to accelerate the development of products for the connected home and commercial buildings. With a narrow platform abstraction layer and a small memory footprint, OpenThread is highly portable. It supports system-on-chip (SoC), network co-processor (NCP), and radio co-processor (RCP) designs.

Silicon Labs has developed an OpenThread-based SDK tailored to work with Silicon Labs hardware. The Silicon Labs OpenThread SDK is a fully tested enhanced version of the GitHub source. It supports a broader range of hardware than does the GitHub version, and includes documentation and example applications not available on GitHub.

These release notes cover SDK version(s):

2.6.2.0 GA released on April 1, 2025
2.6.1.0 GA released on February 5, 2025
2.6.0.0 GA released on December 16, 2024

Thread

KEY FEATURES

OpenThread

- Thread 1.4 certification compliance for Thread devices (SoC)
- Thread 1.3 certification compliance, and 1.4 Alpha support for OTBR
- OTBR NCP mode support - Alpha
- OTBR RCP mode with MCU host (x917) - Alpha
- KNX IOT API support (delivered in Silabs GitHub)
- xG26 Module support
- TrustZone secure key storage support for Thread / MP SoC projects

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/thread . 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 OpenThread SDK, see Using This Release.

Compatible Compilers:

GCC (The GNU Compiler Collection) version 12.2.1, provided with Simplicity Studio.

 Copyright © 2025 by Silicon Laboratories

Contents

1 New Items

1.1 New Components

Added in release 2.6.0.0

- ot_diags - This component provides diagnostic functionalities for OpenThread. This component will enable the standard OpenThread Factory Diagnostics Module in the project and implement the `diag` command in the OpenThread CLI. See https://github.com/openthread/openthread/blob/main/src/core/diags/README.md for a detailed reference on the diagnostics CLI commands.
- ot_stack_tcp_stubs - If this component is installed, TCPIp implementation will be stubbed out for an FTD/MTD application. Otherwise, the TCPIp implementation is included by default.

1.2 New Features

Added in release 2.6.0.0

- The versions of OpenThread and the OpenThread Border Router have been updated. See sections 8.2 and 8.3.
- Libraries and sample applications in this SDK default to Thread 1.4.
- OTBR NCP mode support – Alpha
    - This feature has been created to add support to the OTBR that has an ability to let the host processor sleep, while maintaining connectivity on the Thread mesh network.
    - Keeping the OT Network alive while the host is off/sleeping is accomplished using NCP support.
    - For additional detail see util/third_party/wpantund/README.md.
- OTBR RCP mode with MCU host (x917) – Alpha
    - Located in protocol/openthread/embedded-otbr.
    - Builds an OpenThread Border Router (OTBR) as MCU host on SoC SiWx917. Supports BRD4338A and BRD4342A for host. Supports MG24 (4187c) RCP.
    - Supported MCU projects are under ./si91x_project:
        - There is an option to build a simple OpenThread FTD CLI (openthread-si91x-mcu-host.slcp) or OpenThread Border Router application (openthread-si91x-otbr-mcu-host.slcp).
    - Various OTBR components are under ./si91x_component.
    - Border router application code is under: ./examples/border_router and ./examples/cli (for CLI app).
    - PAL code for SiWx917 and MCU portions of the code are under: ./src/mcu.
    - To Build:
        - See README for instructions on build and setup.
        - Build scripts are under : ./script.
        - Build sets up this space (embedded-otbr) as an SLC extension for the SiSDK it links to.
        - Also links with WiseConnect SLC extension.
    - Notes:
        - This project has dependencies on WiseConnect, SDK, and openthread code.
        - Requires WiseConnect v3.3.2, which does not work with sisdk-2024.12 due to freertos component issues. Therefore, sisdk-2024.12 will contain PoC artifacts of embedded OTBR which will depend on sisdk-2024.6.2 SDK to build the PoC.
- KNX IOT API support (delivered in Silabs GitHub)
    - This repo contains example platform drivers for Siliabs: https://github.com/SiliconLabs/KNX-IOT-STACK.
    - The example platform drivers are intended to present the minimal code necessary to support KNX over OpenThread.
    - The platform drivers are used by the ot-lightswitch sample apps.
    - To learn more about building and running the sample apps, refer to:
        - KNX IoT OpenThread Light Switch Demo
- xG26 Module support – Refer to the list of MGM260 modules in the New Platform Support section.
- TrustZone secure key storage support for Thread / MP SoC projects
    - protocol/openthread/sample-apps/ot-ble-dmp/trustzone/ot-ble-dmp-tz-ns.slcp
    - protocol/openthread/sample-apps/ot-ble-dmp/trustzone/ot-ble-dmp-tz.slcw

1.3 New APIs

Added in release 2.6.0.0

- None.

1.4 New Platform Support

Added in release 2.6.0.0

- New modules
  - MGM260PD32VNA2
  - MGM260PD32VNN2
  - MGM260PD22VNA2
  - MGM260PB32VNA5
  - MGM260PB32VNN5
  - MGM260PB22VNA5
- New radio boards
  - MGM260P-RB4350A
  - MGM260P-RB4351A
- Explorer Kit
  - BRD2709A
  - MGM260P-EK2713A

2 Improvements

Changed in release 2.6.2.0

- Enabled OPENTHREAD_CONFIG_CHILD_SUPERVISION_CHECK_TIMEOUT for all sample apps so that children can supervise loss of a parent link as specified in the Thread standard.

Changed in release 2.6.1.0

- Increased default stack size for baremetal non-rcp projects to account for observed stack issues with thread joiner operations.

Changed in release 2.6.0.0

- The openthread API includes all changes up to and including the updates described here: https://openthread.io/reference/api-updates#november_18_2024.
- Added port and pin validation to the otPlatDiagGpioSet, otPlatDiagGpioGet, otPlatDiagGpioSetMode, and otPlatDiagGpioGetMode APIs.
- Upgrade to Mbed TLS v3.6.2.
- For certification inheritance, the Platform Abstraction Layer (PAL) is no longer shipped as a library. Instead, it will always be built from source, giving more control to application developers to take advantage of HW specific features.
- Changed instances of uartdrv_usart component to iostream_usart in openthread sample apps.
- TCPIp support can be included or excluded from certification builds by excluding or including the ot_stack_tcp_stubs component.
- Radio receive sensitivity, RSSI averaging time, and RSSI averaging timeout can now be configured using the new SL_OPENTHREAD_RECEIVE_SENSITIVITY, SL_OPENTHREAD_RSSI_AVERAGING_TIME, and SL_OPENTHREAD_RSSI_AVERAGING_TIMEOUT configuration options.
- TrustZone sample applications are now production quality.
- A bootloader and the ‘bootloader_interface’ component are now required for detailed crash info. Without these, the crash handler will only print out the reset reason as determined by the EMU/RMU.
- Added workspace to enable trustzone for the OpenThread BLE DMP application.
- The following was added to NCP README to better reflect the actual status of the NCP samples apps. “Note that the NCP design model is not supported or tested, although experimental support is available with the OpenThread stack.”
- Updated the description of the ot_ncp_source component to make it clear that NCP support is experimental and RCP support is production.
- There is now a new radio_priorities_15_4 component. This component enables dynamic adjustment of active transmit priority using a starting priority and a “step” value. The first try starts at the starting active Tx priority, and every subsequent retry decrements the starting value by the “step” value, effectively increasing the priority.
- OPENTHREAD_CONFIG_DIAG_ENABLE config option has been deprecated. Diagnostic support can be enabled by including the ot_diags component. Thread certification libraries are built with diagnostic support.

3 Fixed Issues

Fixed in release 2.6.2.0

ID #	Description
1286531 1408409 1408885 1412308	Addressed a CSL race condition that occurred when scheduling receive and transmit requests which could periodically affect the transmit state machine and outgoing packets.
1377923	Fixed an issue where a coex-enabled 15.4 RCP device could sometimes run into an assert if an OpenThread ongoing TX ACK flag was never cleared.
1405795 1408459	Addressed an issue where old package specific version numbers and git hash commits defined in sl_openthread_generic_config.h were copied from an old project to a new project during the upgrade process, making it appear as if the new project was still associated with the previous SDK. In this release we have moved these definitions from sl_openthread_generic_config.h and placed them in a new include file in the SDK (platform-abstraction/efr32/package-info.h) thus deprecating sl_openthread_generic_config.h. Note: after an upgrade, please manually remove sl_openthread_generic_config.h from the new <project-dir>/config directory.
1412533 1413298	Fixed an invalid project context error when selecting a new OpenThread trustzone workspace from the Studio new project wizard.

Fixed in release 2.6.1.0

ID #	Description
1360675	Fixed failing SRP issue. While adding a service from the SRP client (MTD) to the SRP server (FTD), the SRP registering onto the server outputs an error as shown below from the logs on the FTD: [W] SrpServer-----: Failed to verify message signature: NotCapable [W] SrpServer-----: Failed to process DNS Additional section: NotCapable [W] SrpServer-----: Send fail response: 5
1390396	Addressed a failing Network Diagnostics test for MTDs in 1.4 certification by enabling the uptime feature OPENTHREAD_CONFIG_UPTIME_ENABLE.
1400642	Fixed a bug in flash driver, where on non-PSA enabled devices some NVM settings were not cleared. Updated the flash abstraction to cycle through and delete all Openthread related keys from the device on Factory Reset.

Fixed in release 2.6.0.0

ID #	Description
482915 495241 1295252	Fixed a limitation with the UART driver which could cause characters to be lost on CLI input or output. The issue was resolved by replacing the underlying uartdrv driver with the iostream driver.
1238120	In a multiprotocol RCP environment, certain incoming packets were wrongly interpreted as Zigbee Green Power packets, triggering a random outgoing packet. The issue is fixed by strengthening the Zigbee Green Power packet detection on the RCP.
1289835	Fixed issue with the openthread border router dropping more packets than expected when running within a docker container and under duress.
1295725	Fixed an issue where an SSED could potentially run out of Message buffers during Matter Commissioning.
1295848	Fixed occasional occurrence of duplicate header bytes that were causing rcp resets in busy networks with a lot of traffic over SPI. Also, it is recommended to use a higher spi-bus speed (such as 4Mhz - depending on the network traffic) with a reduced debug logging level to have stable communication between host and rcp. Higher spi speed on OTBR can be set in radio-url as 'spi-speed=<>' argument.
1329286	Removed the "-Werror=unused-parameter" GCC compiler option so that applications that have unused parameters can build.
1355458	Fixed typo where last LQI was mistakenly set to RSSI which would result in an in correct value in the link metrics enhanced ACK.

ID #	Description
1357055	Resolved issue using the openthread certification libraries with MGM240SD22VNA2.
1358944	The ot-ble-dmp sample application is now only supported on parts with at least 768k of flash.
1362028	Removed the check in radio PAL for ongoing ACK transmission before calling RAIL_Idle, as this was causing sleepy devices to miss sleep windows sporadically. When IDLE mode is RAIL_IDLE, the RAIL waits for the radio to finish the ongoing Tx operation before idling. Therefore, this check is unnecessary.
1365554	Fixed issue in which efr32 radio counter interfaces were inaccessible by RCP requests over spinel.
1379694	Fixed issue that would result in a bus fault when full logging is enabled with RTT.

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/thread in the Tech Docs tab.

ID #	Description	Workaround
815275	Ability to modify the Radio CCA Modes at compile-time using a configuration option in Simplicity Studio is currently not supported.	Use the SL_OPENTHREAD_RADIO_CCA_MODE configuration option defined in openthread-core-efr32-config.h header file included with your project.
1356462	In heavy traffic situations an RCP reset can occasionally cause the OTBR to assert.	No known workaround.
1362023	When building a Matter app and adding the Mikroe IOStream USART component, the openthread file, iostream_uart.c, is added to the project, leading to the build errors.	Add a component that provides ot_uart_implemention.
1363723	A corrupted SPI header may occur as a result of an OTBR/RCP communicating with ~30 MTD End Devices, polling at 500ms and sending pings to the OTBR/RCP every minute.	No known workaround.

5 Deprecated Items

Deprecated in release 2.6.0.0

Use of the UARTDRV driver has been deprecated and replaced with the IOSTREAM driver.

6 Removed Items

Removed in release 2.6.1.0

- Removed upgrade rule to change uart component instances for Thread apps.

Removed in release 2.6.0.0

None.

7 Multiprotocol Gateway and RCP

7.1 New Items

Added in release 2.6.2.0

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 2.6.0.0

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 2.6.2.0

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 2.6.1.0

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:

1. 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.
2. 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 2.6.0.0

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 2.6.2.0

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 2.6.1.0

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 2.6.0.0

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 up-coming 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 re-moval of the container.

7.6 Removed Items

Removed in release 2.6.1.0

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

Removed in release 2.6.0.0

None.

8 Using This Release

This release contains the following

- Silicon Labs OpenThread stack
- Silicon Labs OpenThread sample applications
- Silicon Labs OpenThread border router

For more information about the OpenThread SDK see QSG170: Silicon Labs OpenThread QuickStart Guide. If you are new to Thread see UG103.11: Thread Fundamentals.

8.1 Installation and Use

The OpenThread SDK is part of the Simplicity SDK, the suite of Silicon Labs SDKs. To quickly get started with OpenThread and the Simplicity SDK, start by installing Simplicity Studio 5, which will set up your development environment and walk you through Simplicity SDK 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.

The GSDK default installation location has changed beginning with Simplicity Studio 5.3.

- 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. API references and other information about this release are available on https://docs.silabs.com/openthread/latest/. Select your SDK version in the upper right.

8.2 OpenThread GitHub Repository

The Silicon Labs OpenThread SDK includes all changes from the OpenThread GitHub repo (https://github.com/openthread/openthread) up to and including commit **7f6723ffb**. An enhanced version of the OpenThread repo can be found in the following Simplicity Studio 5 GSDK location:

<GSDK Installation Location>\util\third_party\openthread

8.3 OpenThread Border Router GitHub Repository

The Silicon Labs OpenThread SDK includes all changes from the OpenThread border router GitHub repo (https://github.com/openthread/ot-br-posix) up to and including commit **7d327005e**. An enhanced version of the OpenThread border router repo can be found in the following Simplicity Studio 5 GSDK location:

<GSDK Installation Location>\util\third_party\ot-br-posix

8.4 Using the Border Router

For ease of use, Silicon Labs recommends the use of a Docker container for your OpenThread border router. Refer to AN1256: Using the Silicon Labs RCP with the OpenThread Border Router for details on how to set up the correct version of OpenThread border router Docker container. It is available at https://hub.docker.com/r/siliconlabsinc/openthread-border-router.

If you are manually installing a border router, using the copies provided with the Silicon Labs OpenThread SDK, refer to AN1256: Using the Silicon Labs RCP with the OpenThread Border Router for more details.

Although updating the border router environment to a later GitHub version is supported on the OpenThread website, it may make the border router incompatible with the OpenThread RCP stack in the SDK.

8.5 NCP/RCP Support

The OpenThread NCP support is included with OpenThread SDK but any use of this support should be considered experimental. The OpenThread RCP is fully implemented and supported.

8.6 Security Information

Secure Vault Integration

When deployed to Secure Vault High devices, sensitive keys are protected using the Secure Vault Key Management functionality. The following table shows the protected keys and their storage protection characteristics.

Wrapped Key	Exportable / Non-Exportable	Notes
Thread Master Key	Exportable	Must be exportable to form the TLVs
PSKc	Exportable	Must be exportable to form the TLVs
Key Encryption Key	Exportable	Must be exportable to form the TLVs
MLE Key	Non-Exportable	
Temporary MLE Key	Non-Exportable	
MAC Previous Key	Non-Exportable	
MAC Current Key	Non-Exportable	
MAC Next Key	Non-Exportable	

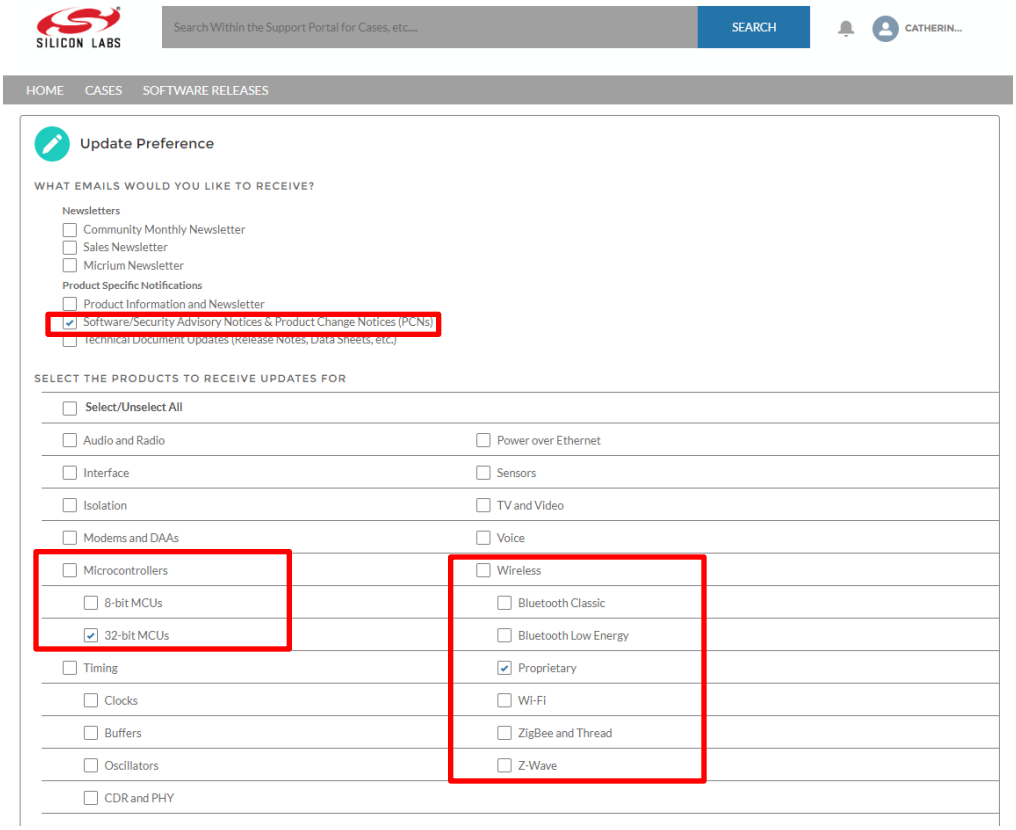
Wrapped keys that are marked as “Non-Exportable” can be used but cannot be viewed or shared at runtime.

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.



8.7 Support

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

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

8.8 SDK Release and Maintenance Policy

For details, see SDK Release and Maintenance Poilcy.

8.9 Thread Certification

This release has been qualified for Thread 1.4.0 for the SoC architectures with Thread Test Harness v62.0 (Member Release). For the Host-RCP architectures, this release has been qualified for Thread 1.3.0 with Thread Test Harness v62.0 (Member Release) with the 1.4.0 qualification currently in alpha release status. For Thread Product certifications tied to this major release and associated patch releases (with no Open-Thread stack updates), Silicon Labs recommends using the above TH version for qualification. Also included with this release is a set of OpenThread stack libraries that may be used for Thread certification by inheritance.

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.

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, 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 LABS

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

www.silabs.com