Silicon Labs OpenThread SDK 2.4.7.0 GA

Gecko SDK Suite 4.5

October 8, 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.

Continued Software Support for Your Silicon Labs Devices

At Silicon Labs, we're committed to delivering dependable, well-supported software for all actively supported hardware products.

The GSDK 4.5 LTS is a long-term support release tailored specifically for **Series 0 and Series 1 devices** that are **not compatible with the Simplicity SDK**. This release ensures ongoing maintenance and stability for devices that are still in active use (i.e., not marked as NRND or EOL).

For customers using **Series 2 or Series 3 devices**, all new features and updates will be provided through the **Simplicity SDK**. To take advantage of the latest innovations and continued support, we encourage you to adopt or migrate to the most recent Simplicity SDK release.

If you have questions about migration or need help choosing the right SDK for your product, our support team is here to help.

These release notes cover SDK version(s):
2.4.7.0 GA released on October 8, 2025

Thread

KEY FEATURES

- The GSDK 4.5 LTS is a long-term support release tailored specifically for Series 0 and Series 1 devices that are not compatible with the Simplicity SDK. This release ensures ongoing maintenance and stability for devices that are still in active use (i.e., not marked as NRND or EOL).

OpenThread

- Thread 1.3.0 certification compliance with Thread Test Harness v59.0 for SoC and Host-RCP architectures
- Thread 1.3.1 feature support - Experimental
- Crash Handler support
- TrustZone Evaluation support
- MR21 support for OpenThread RCP – Production

Multi-Protocol

- Concurrent Listening support (RCP) – MG21 and MG24
- Concurrent Multiprotocol (CMP) Zigbee NCP + OpenThread RCP – production quality
- Dynamic Multiprotocol Bluetooth + Concurrent Multiprotocol (CMP) Zigbee and OpenThread support on SoC

Compatibility and Use Notices

For information about security updates and notices, see the Security chapter of the Gecko 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

The **GSDK 4.5 LTS** is a long-term support release tailored specifically for **Series 0 and Series 1 devices** that are **not compatible with the Simplicity SDK**. This release ensures ongoing maintenance and stability for devices that are still in active use (i.e., not marked as NRND or EOL).

For customers using **Series 2 or Series 3 devices**, all new features and updates will be provided through the **Simplicity SDK**. To take advantage of the latest innovations and continued support, we encourage you to adopt or migrate to the most recent Simplicity SDK release.

1.1 New Components

Added in release 2.4.6.0

- 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.

Added in release 2.4.2.0

- ot_core_vendor_extension - This component implements ot::Extension::ExtensionBase for EFR32. When used with the OT Crash Handler component, crash info will be printed after the OpenThread instance is initialized.

Added in release 2.4.0.0

- ot_crash_handler - This component provides a set of APIs for printing crash info. In the case of a crash, this component captures the details and prints them on the next reboot.
- ot_rtt_log - This component adds support for RTT logging, which is the mechanism used for the platform defined logging interface.

1.2 New Features

Added in release 2.4.7.0

- The versions of OpenThread and the OpenThread Border Router have been updated. See sections 8.2 and 8.3.
- Thread 1.4 compliance for Border Router

  Thread 1.4 specification introduces the following mandatory features for the Border Router:
  - Thread Credential Sharing
    - Provides a standard protocol to allow Administrator access to Thread Border Routers.
    - Enables authentication via a short one-time/ephemeral “Thread Administrator Passcode”.
  - Diagnostics: Network Management and Troubleshooting
    - With Mesh devices having transient roles and difficult to troubleshoot, this feature:
      - Enables enumeration of network participants and means to recreate network state at any time.
      - Disambiguates distinctions between various network layers.
      - Helps classify network diagnostic information for user purposes.
  - Thread over Infrastructure, aka TREL
    - Allows Thread to utilize Wi-Fi/Ethernet links in the Thread mesh topology.
    - Enables merging of Thread Partitions over Wi-Fi/Ethernet.
  - Public Internet Connectivity IPv4/IPv6 support
    - DHCPv6 prefix delegation – To support IPv6 prefix delegation and distribute prefixes in the interior network. Also enables DHCPv6-PD client support in the external IPv6 network.
  - Outbound IPv4 connectivity
    - To support stateful NAT64 as a solution for IPv6/IPv4 translation.
  - DNS Recursive Resolver
    - Allows subscriptions over TLS/TCP (where TCP is preferred over UDP because service lists can lead to quite large packets).
  - TCP (Bulk Transfer Protocol)

- Support for TCP as a standard component/protocol on Thread stacks to remedy throughput shortcomings for Bulk Transfer.

Added in release 2.4.6.0

- The versions of OpenThread and the OpenThread Border Router have been updated. See sections 8.2 and 8.3.
- When migrating multiprotocol projects to this SDK version, users may need to update the app.c file to reflect SDK related changes.

Added in release 2.4.5.0

- Includes conformance and support for Thread 1.4 Credential sharing, Network diagnostics features, and improvements made for border router Public Internet Connectivity. Support for 1.4 commercial TCAT (Thread over Authenticated TLS) feature is present but not tested.
- Libraries and sample applications in this SDK default to Thread 1.4.

Added in release 2.4.1.0

- Support added for factory diagnostic channel and transmit power CLI commands.

Added in release 2.4.0.0

- Crash Handler support - With this release, a crash handler component has been introduced for use with OpenThread applications. By including it in an OpenThread project, it will automatically capture detailed information about core registers, information about the C stack, and reset information. On the next boot-up, this data is printed using the OpenThread logging system.
- TrustZone evaluation support - Added evaluation workspaces to enable TrustZone for OpenThread CLI applications.
- Features introduced with OpenThread up to and including commit 7074a43e4. This includes support for ongoing Thread 1.3.1 features. The default setting for Silicon Labs sample applications is still 1.3.0.

1.3 New APIs

Added in release 2.4.7.0

- The openthread API includes all changes up to and including the updates described here: https://openthread.io/reference/api-updates#may_14_2025.

Added in release 2.4.6.0

- The openthread API includes all changes up to and including the updates described here: https://openthread.io/reference/api-updates#january_24_2025.

Added in release 2.4.1.0

- otPlatDiagChannelSet - This function sets the channel to use for factory diagnostics.
- otPlatDiagTxPowerSet - This function sets the transmit power to use for factory diagnostics.

Added in release 2.4.0.0

- otPlatResetToBootloader - reset to bootloader mode. Users can either call this API directly in code or via the CLI command "reset bootloader".

1.4 New Radio Board Support

Added in release 2.4.0.0

Support has been added for the following radio boards:

- BRD4198A - EFR32MG24B210F1536IM48-B

2 Improvements

Changed in release 2.4.7.0

- OPENTHREAD_CONFIG_CHILD_SUPERVISION_CHECK_TIMEOUT
    - OPENTHREAD_CONFIG_CHILD_SUPERVISION_CHECK_TIMEOUT is now enabled by default for all sample apps so that children can supervise loss of a parent link as specified in the Thread standard.
- OTBR Improvements to host / RCP communication
    - The default OTBR agent is now configured to work at 460800 baud rate, so default ot-rcp sample app is also configured to work at the same baud rate. Make sure to match whatever baud rate you set between the OTBR radio URL and the RCP configuration. Please note that this doesn't affect CPC RCP apps as CPC defines its own baud rate setting.

Changed in release 2.4.6.0

- Increased default stack size for baremetal non-rcp projects to account for observed stack issues with thread joiner operations.
- TCPIp support can be included or excluded from certification builds by excluding or including the ot_stack_tcp_stubs component.
- When upgrading projects from to this release, users will need to remove the sl_openthread_generic_config.h file from the new <project-dir>/config directory and may need to update the app.c file to reflect SDK changes.

Changed in release 2.4.5.0

- OpenThread posix projects built with **OT_MULTIPAN_RCP** enabled now require both an **iid** and **iid-list** radio URL parameter.

Changed in release 2.4.0.0

- Logging - Default log mechanism is changed from RTT to UART. RTT logging and associated libraries are removed from ot_third-party and added to a new component, ot_rtt_log.
- Posix vendor extension options:
    - Changed OT_POSIX_CONFIG_RCP_VENDOR_DEPS_PACKAGE value from SilabsRcpDeps to posix_vendor_rcp.cmake.
    - Removed CMAKE_MODULE_PATH option.
- NCP sample applications - Prebuilt NCP sample applications are considered experimental and are no longer packaged as prebuilt demos.
- SL_OPENTHREAD_CSL_TX_UNCERTAINTY, SL_OPENTHREAD_HFXO_ACCURACY, and SL_OPENTHREAD_LFXO_ACCURACY are now configurable.
- Improvements introduced with OpenThread up to and including commit 7074a43e4.
    - The OT_CONFIG CMake option has been replaced in favor of two new options: OT_PLATFORM_CONFIG and OT_PROJECT_CONFIG.
    - Changes and clarifications in CSL APIs. The major CLI/API change is that now csl period requires microsecond value instead of 10 symbol units.
    - Changes in the Spinel Interface. The Spinel interface is now created based on radio URL protocol to support multiple interfaces (hdlc/spi/vendor) at the same time. Silicon Labs CPC builds will continue to support only one interface, which is the vendor interface (CPC).
    - During build time, setting OT_POSIX_CONFIG_RCP_BUS is now deprecated. Instead turn on one or more of OT_POSIX_RCP_HDLC_BUS, OT_POSIX_RCP_SPI_BUS, or OT_POSIX_RCP_VENDOR_BUS as needed.

3 Fixed Issues

Fixed in release 2.4.7.0

ID #	Description
1187748	Use the RAIL API RAIL_IEEE802154_SetRxToEnhAckTx to selectively set higher RxToTx for enhanced ACKs only. Previously, we were globally setting this higher RxToTx value for all ACKs.
1363723	Invalid SPI headers will no longer result in a RCP reset.
1408409 / 1412413	Addressed a CSL race condition that occurred when scheduling receive and transmit requests which could periodically affect the transmit state machine and outgoing packets.
1338644 / 1414967	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).
1507808	Fixed a hard fault which could occur when utilsSoftSrcMatchShortFindEntry was called with an invalid Iid.

Fixed in release 2.4.6.0

ID #	Description
1295725 / 1381950	Fixed an issue where an SSED could potentially run out of Message buffers during Matter Commissioning.
1357055 / 1368198	Resolved issue using the openthread certification libraries with MGM240SD22VNA2.
1358563 / 1358565	Fixed the phase calculation during Enhanced ack generation by considering correct incoming packet length.
1360787 / 1360790	Fixed ccaBackoff value to have a predictable radio warmup time during CSL schedule transmits.
1362028 / 1364540	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 / 1371514	Fixed issue in which efr32 radio counter interfaces were inaccessible by RCP requests over spinel.
1390396 / 1396441	Addressed a failing Network Diagnostics test for MTDs in 1.4 certification by enabling the uptime feature OPENTHREAD_CONFIG_UPTIME_ENABLE.
1392898 / 1400642 / 1400644	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.
1405795 / 1408457	Addressed an issue where old package specific version numbers and git hash commits defined in sl_openthread_generic_config.h are 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.

Fixed in release 2.4.5.0

ID #	Description
1249346	Addressed an issue where the RCP could incorrectly dequeue packets destined for the host resulting in a parse error in the OTBR and unexpected termination.
1251926	When using the crash handler feature in a Host / RCP environment, the RCP was attempting to send the crash info to the Host too early after reboot, causing the info to be dropped prior to logging. This issue is addressed by introducing a new spinel property to the upstream openthread stack. The property allows the Host to request crash logs from the RCP once the Host is ready for them. See https://github.com/openthread/openthread/pull/10061 for more information on the new spinel property.

1255247	Addressed an issue where the RCP could incorrectly dequeue packets destined for the host resulting in a parse error in the OTBR and unexpected termination.
1358944	The ot-ble-dmp sample application is now only supported on parts with at least 768k of flash.
1289835 1360471	Fixed issue with the openthread border router dropping more packets than expected when running within a docker container and under duress.

Fixed in release 2.4.4.0

ID #	Description
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 / 1334039	Removed the "-Werror=unused-parameter" GCC compiler option so that applications that have unused parameters can build.

Fixed in release 2.4.2.0

ID #	Description
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.
1249346 / 1255247	Addressed an issue where the RCP could incorrectly dequeue packets destined for the host, resulting in a parse error in the OTBR and unexpected termination.
1251926	When using the crash handler feature in a host / RCP environment, the RCP was attempting to send the crash info to the host too early after reboot, causing the info to be dropped prior to logging. This issue is addressed with the introduction of the ot_core_vendor_extension component, which delays the processing of the crash info after reboot until the OpenThread instance is initialized.
1251952	Fixed undefined reference to otInstanceResetToBootloader when building with ot_cert_libs and bootloader_interface.
1255595	Fixed a type promotion issue when handling large values in otPlatAlarm APIs.
1249492	The dBus FactoryReset command no longer causes the openthread border router to terminate.

Fixed in release 2.4.1.0

ID #	Description
1208578	Added support for linking the CPC library to Posix host apps by using provided paths, and by using pkg-config.
1235923	Fixed a wrapping bug in calls to otPlatAlarmMilliStartAt and otPlatAlarmMicroStartAt.
1243597	Removed extra ot-ble-dmp-no-buttons sample apps from demos folder.
1251932	Increased OPENTHREAD_CONFIG_CSL_RECEIVE_TIME_AHEAD to 750 µsec for default FTD and MTD certification libraries for use with devices under test using library-based certification.

Fixed in release 2.4.0.0

ID #	Description
1124161	High traffic environments will no longer cause buffers containing received packet payloads to be overwritten before being processed.
1148720	SED current draw has been improved.
1169011	Increased the stack size of OpenThread task to 4608 bytes (SL_OPENTHREAD_OS_STACK_TASK_SIZE) for DMP applications to avoid stack overflow while joining the thread network.
1193597	OpenThread Radio PAL now maintains Max channel power table.
1227529	Fixed the typo in OPENTHREAD_SPINEL_CONFIG_TX_WAIT_TIME_SECS to OPENTHREAD_SPINEL_CONFIG_RCP_TX_WAIT_TIME_SECS in lower-mac-spinel-config header.

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.

5 Deprecated Items

Deprecated in release 2.4.0.0

- The ot_thirdparty component has been deprecated.

6 Removed Items

None.

7 Multiprotocol Gateway and RCP

7.1 New Items

Added in release 2.4.0.0

Concurrent listening, the ability for the Zigbee and OpenThread stacks to operate on independent 802.15.4 channels when using an EFR32xG24 or xG21 RCP, is released. Concurrent listening is not available for the 802.15.4 RCP/Bluetooth RCP combination, the Zigbee NCP/OpenThread RCP combination, or for the Zigbee/OpenThread system-on-chip (SoC). It will be added to those products in a future release.

The OpenThread CLI vendor extension has been added to the OpenThread host apps of multiprotocol containers. This includes the coex cli commands.

7.2 Improvements

Changed in release 2.4.0.0

The Zigbee NCP/OpenThread RCP multiprotocol combination is now production quality. This sample application is not supported on Series-1 EFR devices.

7.3 Fixed Issues

Fixed in release 2.4.7.0

ID #	Description
1375724	Fixed a race condition on the multipan RCP that resulted in a transmit complete message not being returned to the host, causing a 5 second pause in further transmissions prior to recovering. (Other ref: 1454200)

Fixed in release 2.4.6.0

ID #	Description
1275378	Fixed an issue where calling emberRadioSetSchedulerPriorities() prior to emberInit() could result in a crash (Other ref: 1381882).
1361436	Fixed an issue that caused dmp_gp_proxy app (with CLI added) to fail to join a network on time.
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.
1365665	Fixed an issue where the host would report receiving a packet with an invalid checksum on end-point 12. (Other ref: 1366154)
1392787	Fixed an issue that caused Zigbeed not restart when performing a Trust Center Backup and Restore Reset Node action.
1405226	Fixed project migration issue and included OT project upgrade rule to reflect newer SDK changes. Note that when customers upgrade their Multiprotocol project, files like app.c will need to be manually ported to reflect newer SDK changes.

Fixed in release 2.4.5.0

ID #	Description
1328799	The soft reset triggered by the Spinel RESET command now clears the buffers of the 15.4 RCP.

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). (Other ref: 1339032)
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. (Other ref: 1339032)
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. (Other ref: 1349176)
1346849	Adding the rail_mux component to a project will now cause it to automatically build with the associated stack library variants. (Other ref: 1349102)

Fixed in release 2.4.4.0

ID #	Description
1184065	Reduced RAM footprint for zigbee_ncp-ot_rcp-spi and zigbee_ncp-ot_rcp_uart on MG13 and MG21.
1282264	Fixed an issue that could have interrupted radio transmit operations by clearing the transmit fifo prematurely causing underflow.
1292537	DMP Zigbee-BLE NCP application now properly showing up in Simplicity Studio UI. (Other ref: 1292540)
1230193	Fixed incorrect node type issue when joining network on end device. (Other ref: 1298347)
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. (Other ref: 1333156)

Fixed in release 2.4.2.0

ID #	Description
1022972	Added coexistence plugin back to Zigbee-OpenThread NCP/RCP sample application.
1231021	Avoid an assert in OTBR that has been observed when joining 80+ zigbee devices by recovering the RCP rather than by passing unhandled transmit errors to the sub mac.
1249346	Addressed an issue where the RCP could incorrectly dequeue packets destined for the host, resulting in a parse error in the OTBR and unexpected termination.

Fixed in release 2.4.1.0

ID #	Description
1213701	zigbeed didn't allow a source match table entry to be created for a child if MAC indirect queue has data already pending for that child. This behavior could lead to application layer transactions between the child and some other device failing due to lack of APS Ack or app-layer response, most notably the disruption and unexpected termination of ZCL OTA Upgrades targeting the child device.
1244461	Source match table entry for child being could be removed despite messages pending.

Fixed in release 2.4.0.0

ID #	Description
1081828	Throughput issue with FreeRTOS-based Zigbee/BLE DMP sample applications.

ID #	Description
1090921	Z3GatewayCpc had trouble forming a network in a noisy environment.
1153055	An assert on the host was caused when there was a communication failure when reading the NCP version from the zigbee_ncp-ble_ncp-uart sample app.
1155676	The 802.15.4 RCP discarded all received unicast packets (after MAC acking) if multiple 15.4 interfaces shared the same 16-bit node ID.
1173178	The host falsely reported hundreds of packets received with mfglib in the Host-RCP setup.
1190859	EZSP error when sending mfglib random packets in the Host-RCP setup.
1199706	Data polls from forgotten end device children were not properly setting a pending frame on the RCP to queue a Leave & Rejoin command to the former child.
1207967	The "mfglib send random" command was sending out extra packets on Zigbeed.
1208012	The mfglib rx mode did not update packet info correctly when receiving on the RCP.
1214359	The coordinator node crashed when 80 or more routers tried to join simultaneously in the Host-RCP setup.
1216470	After relaying a broadcast for address mask 0xFFFF, a Zigbee RCP acting as a parent device would leave the pending data flag set for each child. This resulted in each child staying awake expecting data after each poll, and required some other pending data transaction to each end device to eventually clear this state.

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/gecko-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.
1124140	z3-light_ot-ftd_soc sample app is not able to form the Zigbee network if the OT network is up already.	Start the Zigbee network first and the OT network after.
1170052	CMP Zigbee NCP + OT RCP and DMP Zigbee NCP + BLE NCP may not fit on 64KB and lower RAM parts in this current release. (Other ref: 1393057).	64KB RAM parts are not recommended for NCP + RCP apps.
1209958	The ZB/OT/BLE RCP on MG24 can stop working after a few minutes when running all three protocols.	Will be addressed in a future release.
1221299	Mfglib RSSI readings differ between RCP and NCP.	Will be addressed in a future release.
1334477	Starting and stopping the BLE stack several times might result in the BLE stack not able to restart advertisement again on low RAM (64kB) Series 1 EFR devices in the DMP Zigbee-BLE sample application.	N/A

7.5 Deprecated Items

None

7.6 Removed Items

Removed in release 2.4.0.0

The "NONCOMPLIANT_ACK_TIMING_WORKAROUND" macro has been removed. All RCP apps now by default support 192 µsec turnaround time for non-enhanced acks while still using 256 µsec turnaround time for enhanced acks required by CSL.

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 Thread Fundamentals.

8.1 Installation and Use

The OpenThread SDK is part of the Gecko SDK (GSDK), the suite of Silicon Labs SDKs. To quickly get started with OpenThread and the GSDK, start by installing 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 installation location has changed beginning with Simplicity Studio 5.3.

- Windows: C:\Users\<NAME>\SimplicityStudio\SDKs\gecko_sdk
- MacOS: /Users/<NAME>/SimplicityStudio/SDKs/gecko_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 **fb0446f53**. 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 **622f5ecfc**. 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 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 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.

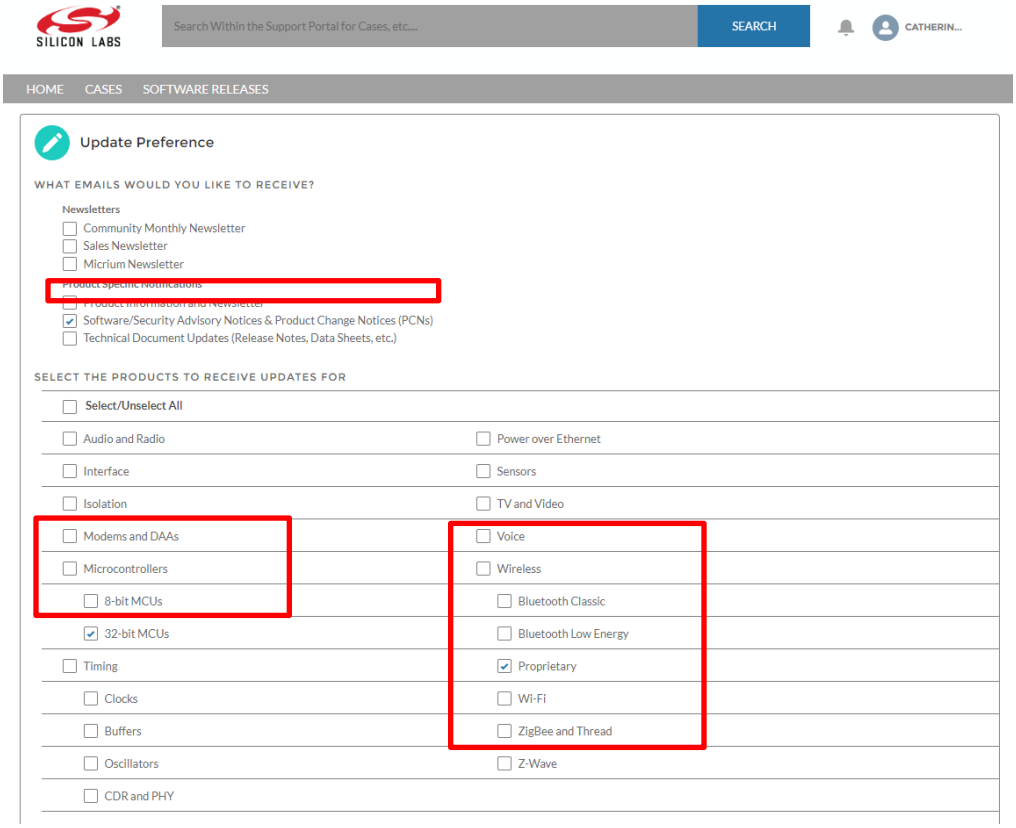
Wrapped keys that are marked as “Exportable” can be used or shared at runtime but remain encrypted while stored in flash.

For more information on Secure Vault Key Management functionality, see 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.

The following figure is an example:



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 Thread Certification

This release has been qualified for the Host-RCP architectures for Thread 1.4.0 with Thread Test Harness v63.0 (Member Release). For Thread Product certifications tied to this major release and associated patch releases (with no OpenThread stack updates), Silicon Labs recommends using the above TH version for qualification. This release is not planned for Thread certification for Soc devices.

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