Silicon Labs OpenThread SDK 2.1.6.0 GA
Gecko SDK Suite 4.1
September 13, 2023

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 in order 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):

2.1.6.0 released on September 13, 2023 (underlying platform changes only)
2.1.5.0 released on June 28, 2023 (support for EFR32xG21, Revision C and later)
2.1.4.0 released on January 18, 2023 (multiprotocol updates)
2.1.3.0 released on October 19, 2022 (early access part support)
2.1.2.0 released on September 14, 2022
2.1.1.0 released on August 17, 2022
2.1.0.0 GA released on June 8, 2022

Thread

KEY FEATURES

- Alpha SPI support for OpenThread RCP without CPC
- Thread 1.2 and 1.3 support for OpenThread
- Updated GCC compiler version to 10.3.1
- Alpha Concurrent Multiprotocol Zigbee in NCP mode and OpenThread in RCP mode
- Alpha Dynamic Multiprotocol Bluetooth and multi-PAN 802.15.4 in RCP mode

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 10.3-2021.10, provided with Simplicity Studio.

Contents

1 New Items

1.1 New Components

Added in release 2.1.2.0

ot_coap_cert_libs - OpenThread CoAP Certification Libraries

Include this component in your project to build using CoAP API support and the configuration required for certification and to link against the OpenThread CoAP certification libraries provided with the GSDK.

Added in release 2.1.1.0

ot_cert_libs - OpenThread Certification Libraries

Include this component in your project to build using the configuration required for certification and to link against the OpenThread certification libraries provided with the GSDK.

ot_reference_device - OpenThread Reference Device configuration

Include this component in your project to build from source using the configuration required to run as a reference device in the Thread test harness.

Added in release 2.1.0.0

NCP SPIDRV

This component provides SPIDRV (SPI) support for the OpenThread stack. It requires that the OpenThread NCP component be included with the project. Currently only the OpenThread RCP stack is supported.

1.2 New Features

Added in release 2.1.2.0

- This release includes the following additional libraries. These libraries may be included in your project by including the ot_coap_cert_libs component to allow your device to use CoAP and to potentially obtain Thread certification by inheritance. Please contact Silicon Labs to understand which libraries are available for Thread certification by inheritance and to the Thread group for details on applying for certification by inheritance.
    - libsl_ot_stack_ftd_coap_efr32mg12_gcc.a
    - libsl_ot_stack_ftd_coap_efr32mg13_gcc.a
    - libsl_ot_stack_ftd_coap_efr32mg21_gcc.a
    - libsl_ot_stack_ftd_coap_efr32mg24_gcc.a
    - libsl_ot_stack_mtd_coap_efr32mg12_gcc.a
    - libsl_ot_stack_mtd_coap_efr32mg13_gcc.a
    - libsl_ot_stack_mtd_coap_efr32mg21_gcc.a
    - libsl_ot_stack_mtd_coap_efr32mg24_gcc.a

Added in release 2.1.1.0

- This release includes the following libraries. These libraries may be included in your project by including the ot_cert_libs component to allow your device to potentially obtain Thread certification by inheritance. Please contact Silicon Labs to understand which libraries are available for Thread certification by inheritance and to the Thread group for details on applying for certification by inheritance.
    - libsl_ot_stack_ftd_efr32mg12_gcc.a
    - libsl_ot_stack_ftd_efr32mg13_gcc.a
    - libsl_ot_stack_ftd_efr32mg21_gcc.a
    - libsl_ot_stack_ftd_efr32mg24_gcc.a
    - libsl_ot_stack_mtd_efr32mg12_gcc.a
    - libsl_ot_stack_mtd_efr32mg13_gcc.a
    - libsl_ot_stack_mtd_efr32mg21_gcc.a
    - libsl_ot_stack_mtd_efr32mg24_gcc.a

- libsl_platform_ftd_dmp_efr32mg12_gcc.a
- libsl_platform_ftd_dmp_efr32mg13_gcc.a
- libsl_platform_ftd_dmp_efr32mg21_gcc.a
- libsl_platform_ftd_dmp_efr32mg24_gcc.a
- libsl_platform_ftd_efr32mg12_gcc.a
- libsl_platform_ftd_efr32mg13_gcc.a
- libsl_platform_ftd_efr32mg21_gcc.a
- libsl_platform_ftd_efr32mg24_gcc.a
- libsl_platform_mtd_dmp_efr32mg12_gcc.a
- libsl_platform_mtd_dmp_efr32mg13_gcc.a
- libsl_platform_mtd_dmp_efr32mg21_gcc.a
- libsl_platform_mtd_dmp_efr32mg24_gcc.a
- libsl_platform_mtd_efr32mg12_gcc.a
- libsl_platform_mtd_efr32mg13_gcc.a
- libsl_platform_mtd_efr32mg21_gcc.a
- libsl_platform_mtd_efr32mg24_gcc.a

- This release includes support for TCP. Refer to UG103.11 Thread fundamentals and AN1372 Configuring Sample Apps for Thread 1.3 for additional details.

Added in release 2.1.0.0

- The versions of OpenThread and the OpenThread Border Router have been updated. See sections 8.2 and 8.3.
- Support for Thread 1.3. This version of OpenThread includes low-power TCP support. This is not a mandatory Thread 1.3 component, so it is turned off by default in our sample apps.
  - Our OpenThread sample apps are built with the newly introduced protocol version OT_THREAD_VERSION_1_3 as default. Note that 1.3.0 features should in theory work with OT_THREAD_VERSION_1_2 or higher. However, we are changing the default version as mandated by the specification.
- Alpha SPI support for OTBR RCP.

1.3   New Radio Board Support

Added in release 2.1.0.0

Support has been added for the following radio boards:

- BRD4319 – MGM240L (MGM240L022RNF2)
- BRD4195 - EFR32MG21B010F1024IM32
- BRD4196 - EFR32MG21B020F1024IM32

2 Improvements

None

3   Fixed Issues

Fixed in release 2.1.2.0

ID #	Description
825185	Button 0, as always, allows you to transition between rxOnWhenIdle=false and rxOnWhenIdle=true. However now, when transitioning to rxOnWhenIdle=false, the application will detach from its parent and then reattach either to the current parent or another to allow for increased buffering capacity required with an rx-off-when-idle device.
855805	Ensure the radio power level is restored to the appropriate level when performing an energy scan or switching bands.
1020629	The following warning is now displayed when viewing the stack configuration settings when the project is set to link against the certification libraries. WARNING: Changing configuration values in FTD/MTD certification headers will have ramifications on inheriting Thread certification using pre-built certification libraries. If you must change any of these values, note that it requires rebuilding the FTD/MTD certification libraries. Note: If intending to use the "OpenThread CoAP Certification configuration" component for CoAP library inheritance, please make sure to turn on CoAP API options if you require CoAP API CLI.
1028083	Add multiprotocol filtering options in Simplicity Studio for OpenThread RCP sample applications.
1034513	An issue where OpenThread Stack FTD/MTD software component configure page was blank has been addressed. In release 2.1.1.0, a change was made that inadvertently moved the configuration from the OpenThread Stack component to the OpenThread Reference Device configuration component. This is fixed in release 2.1.2.0 and custom configuration can again be made through the OpenThread Stack component.

Fixed in release 2.1.1.0

ID #	Description
821837	Compilation warnings when compiling with TCP are now ignored by default.
1017551	The following configuration values are now set by default for all openthread sample applications. Please note that if you want different values for these parameters, they need to be overridden in your application's .slcp file. • OPENTHREAD_CONFIG_BACKBONE_ROUTER_ENABLE:1 • OPENTHREAD_CONFIG_BORDER_ROUTER_ENABLE:1 • OPENTHREAD_CONFIG_COAP_API_ENABLE;1 • OPENTHREAD_CONFIG_COAP_OBSERVE_API_ENABLE:1 • OPENTHREAD_CONFIG_COAP_SECURE_API_ENABLE:1 • OPENTHREAD_CONFIG_COMMISSIONER_ENABLE:1 • OPENTHREAD_CONFIG_DHCP6_CLIENT_ENABLE:1 • OPENTHREAD_CONFIG_DHCP6_SERVER_ENABLE:1 • OPENTHREAD_CONFIG_DNSSD_SERVER_ENABLE:1 • OPENTHREAD_CONFIG_JOINER_ENABLE:1 • OPENTHREAD_CONFIG_MAC_FILTER_ENABLE:1 • OPENTHREAD_CONFIG_REFERENCE_DEVICE_ENABLE:1 • OPENTHREAD_CONFIG_SRP_SERVER_ENABLE:1 • OPENTHREAD_CONFIG_TMF_NETDATA_SERVICE_ENABLE:1 • OPENTHREAD_CONFIG_LOG_OUTPUT: OPENTHREAD_CONFIG_LOG_OUTPUT_PLATFORM_DEFINED

Fixed in release 2.1.0.0

ID #	Description
620720	The 'diag radio state' CLI command used with the Factory Diagnostics Module now operates correctly and no longer just returns 'invalid'.

ID #	Description
757884	Addressed issues with failing Thread 1.2 certification tests. The ID listed is one of several IDs associated with Thread 1.2 certification issues. All SoC Thread 1.2 certification issues have been resolved in this release. One note: It is recommended customers use the OT_RCP_RESTORATION_MAX_COUNT option for OTBRs to enable RCP recovery for long-running OTBR use cases.
758932	Allow the Bluetooth GATT Configurator to be included when building DMP apps from scratch in Simplicity Studio.
764331	Addressed an issue for Raspberry Pi and Linux docker container users by requiring them to run "sudo modprobe ip6table_filter" for OTBR firewall support before starting the docker container. This allows OTBR scripts to create rules inside the docker container before otbr-agent starts.
816037	Fixed an issue where the ot-ble-dmp sample app would fail to compile in Simplicity Studio with the following: "error: 'SL_OPENTHREAD_RTOS_TASK_PRIORITY' undeclared here (not in a function); did you mean 'SL_OPENTHREAD_RTOS_TASK_PRIORITY_STARTUP'?"
818599	Addressed an issue where OpenThread-supported IC OPNs and radio boards associated with the EFR32MG24 family were not showing up in Simplicity Studio.
827315	Addressed an issue where a device would encounter an assert when first booting a PSA application.

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
482915 495241	A known limitation with the UART driver can cause characters to be lost on CLI input or output. This can happen during particularly long critical sections that may disable interrupts, so it can be alleviated by repeating the CLI or waiting long enough for state changes.	No known workaround
754514	Double ping reply observed for OTBR ALOC address.	No known 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.
829618	Sample apps should not define OPENTHREAD_CONFIG_REFERENCE_DEVICE_ENABLE.	Can be safely ignored or can be manually removed from the sample app.

830554	RAIL PA ramp time is hard-coded to 10.	Modify the following code and replace 10 with SL_RAIL_UTIL_PA_RAMP_TIME_US.
		RAIL_TxPowerConfig_t txPowerConfig =
		    ...
		    { SL_RAIL_UTIL_PA_SELECTION_2P4GHZ, SL_RAIL_UTIL_PA_VOLTAGE_MV, 10 }
		    ...
		;

1015604	Cannot join OpenThread network after enabling NetworkTimeSync	No known workaround
1021181	Error with using external heap and having message buffers use the heap allocator	This GitHub pull request addresses the issue but is not included in the version of openthread included with this release: https://github.com/openthread/openthread/pull/7933
1023725	"ifconfig up" may fail after a network reset	No known workaround
1030815	**The OpenThread Border Router can display incorrect version string for otbr-agent (`sudo otbr-agent --version`) or for the POSIX stack (`sudo ot-ctl version`).**	**This issue can be avoided by deleting all previous build artifacts (in the build/ folder) before re-installing the OpenThread Border Router.**
1041112	**OTBR / EFR32 RCP can miss forwarding packets from a CSL child if it configures an alternate channel for CSL communication.** **Due to this issue, OTBRs based on GSDK 4.1.2.0 are not expected to pass Thread 1.2 certification unless the customer use cases demand a waiver to exclude all tests that require changing the primary channel.**	**Avoid configuring alternate CSL channels until this issue is addressed.**

5   Deprecated Items

Deprecated in release 2.1.0.0

Beacon Support

Per the Thread specification beacons support has been deprecated in this release. To enable support for deployed devices that may still require beacons support, two configuration macros are added to control the beacon payload generation and parsing.

- *OPENTHREAD_CONFIG_MAC_BEACON_PAYLOAD_PARSING_ENABLE* will enable the device to receive and parse any beacons that contain beacon payload.

- *OPENTHREAD_CONFIG_MAC_OUTGOING_BEACON_PAYLOAD_ENABLE* will enable the device to add Thread payload to outgoing beacons.

Our Platform Abstraction Layer enables *OPENTHREAD_CONFIG_MAC_BEACON_PAYLOAD_PARSING_ENABLE* by default but does not enable *OPENTHREAD_CONFIG_MAC_OUTGOING_BEACON_PAYLOAD_ENABLE* to prevent issues with certification.

Note: *OPENTHREAD_CONFIG_MAC_OUTGOING_BEACON_PAYLOAD_ENABLE* should be enabled if you are intending to deploy products that are backward-compatible with some 1.1 legacy devices

6   Removed Items

None

7 Multiprotocol Gateway and RCP

7.1 New Items

Added in release 2.1.4.0

The Stack API Trace now can be enabled for Zigbeed by setting the debug-level to 4 or 5 in the zigbeed.conf file.

Zigbeed stack version as well as build date and time are now printed in the logs.

Added in release 2.1.1.0

851653 Added option to start cpcd conditional to a firmware version. Also allows firmware update process to be conditional to a firmware version. (-a/--app-version <version>). Added option to restart cpcd after firmware update. (-r/--restart-cpcd)

Added manufacturing library support on a Host-CMP RCP setup. This change allows RF testing on the manufacturing line on a Host-CMP RCP setup by sending mfglib EZSP frames from the Z3Gateway host to the RCP.

Added 802.15.4 CSL support to the multiprotocol RCP.

Added in release 2.1.0.0

A new concurrent multiprotocol configuration is available: Zigbee NCP and OpenThread RCP running concurrently on the EFR32, using the Co-Processor Communication (CPC) architecture. It is released as alpha quality. See *AN1333: Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol RCP* for details.

7.2 Improvements

Changed in release 2.1.0.0

CPC security is now enabled by default in the cpcd.conf file and in the SLCP project files. This means data sent over the serial line between the host and the EFR32 is encrypted. A security commissioning step is required to bind the host to the EFR32. See https://github.com/SiliconLabs/cpc-daemon/blob/main/readme.md for details.

For convenience, the run.sh script in *app/host/multiprotocol/zigbeed/multiprotocol-container/* includes a -K argument for commissioning cpcd security when using the multiprotocol docker container.

The multiprotocol container has been upgraded to use ubuntu 22.04 and BlueZ 5.64.

The zigbee_trust_center_backup component now supports migrating from a Zigbee Host + NCP setup to a Zigbee Host + Zigbeed + RCP setup. See *AN1333: Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol RCP* for details.

Zigbeed built from GSDK sources no longer requires the /accept_silabs_msla_file at runtime. Only the Zigbeed binary from the multiprotocol docker container requires it.

7.3 Fixed Issues

Fixed in release 2.1.4.0

ID #	Description
829675	Fixed bidirectional Green Power commissioning with Z3GatewayGPCombo + Zigbeed + RCP.
1066422	Fixed an intermittent buffer leak in Zigbeed.
1068429	Fixed a race condition that could cause the CMP RCP to assert.
1068435	The CMP RCP now passes compliance test GPP 5.4.1.23.
1068942	Fixed a leak in the RCP source match table that could prevent zigbee devices from joining.
1074172	Fixed sending leave request from Zigbeed when receiving a poll from a non-child.
1074290	Stopped Zigbeed from processing un-acked polls.

ID #	Description
1074593	Fixed issue in which Just-in-time (JIT) messages to sleepy end devices were not sent correctly by the CMP RCP.
1079903	Fixed a bug in the CMP RCP that could cause SPINEL messages to be dispatched incorrectly, resulting in Zigbeed and OTBR crashing or exiting.
1080482	Made SPINEL timeout recovery more robust to heavy transmit traffic by the other protocol stack in CMP RCP.
1081455	Added workaround for non-compliant Zigbee devices that timeout 802.15.4 acks too quickly. To enable the workaround, define NONCOMPLIANT_ACK_TIMING_WORKAROUND.

Fixed in release 2.1.2.0

ID #	Description
1025713	Increased the maximum length of the Zigbeed device path from 40 to 4096.
1030557	Fixed build errors when adding the legacy bootloader interface component to the multiprotocol RCP projects.
1030557	Fixed a build issue to allow the legacy ember_bootloader_interface component to be added to MG1-based multiprotocol RCP projects, for customers using the legacy Ember bootloader. Also added support for CPCd to bootload devices with the legacy ezsp-spi Ember bootloader.

Fixed in release 2.1.1.0

ID #	Description
834191	Fixed cpc-hci-bridge so that it does not consume excessive CPU.
859224	Fixed an issue where CPC security was failing during initialization on MG1.
859301	Fixed an issue where Thread and Zigbee sleepy end devices failed to join to the RCP on MG1.
988216	Fixed issue where Zigbeed failed to run on a 64 bit Raspberry Pi.
824100	Fixed an issue where Z3GatewayCPC could not be started more than once without an error.
851331	Z3GatewayCpc can reconnect to cpcd service without needing a restart.
858153	Fixed a coex component configuration for MG1.
858503	Enabled the coex component by default in the multiprotocol RCP images, and fixed a RAM usage issue that prevented building on MG1.
1019947	Added support for building RCP projects for efr32mg1b and efr32mg1v parts.

Fixed in release 2.1.0.0

ID #	Description
760596	Reduced the CPU consumption of Zigbeed when it is idle.
811566	Fixed an issue where Zigbee sleepy end devices failed to join to an RCP parent.
817698	Fixed a Zigbeed crash due to a null buffer in the lower mac layer.
822233	Fixed an issue that caused dropped CPC packets over VCOM, especially at higher baud rates.
829614	The multi-PAN/multiprotocol 802.15.4 RCP now sets the radio tx power to the maximum of the power levels requested by all 15.4 host applications. This avoids the problem of one application reducing the power and inadvertently causing network connectivity problems for the other application.
830596	Fixed an issue where joining a sleepy end device to the multiprotocol RCP caused Z3Gateway to crash in some circumstances.
831689	Fixed an issue in Zigbeed that resulted in a fixed pan id and other parameters being chosen when forming a network.

7.4   Known Issues in the Current Release

ID #	Description	Workaround
811732	Custom token support is not available when using Zigbeed.	Support is planned in a future release.
828785	There is a known issue with the cpc-hci-bridge that causes the second HCI packet to be dropped if BlueZ sends two HCI packets to the RCP in rapid succession.	A fix is targeted for the next patch release.
937562	Bluetoothctl advertise on command fails with rcp-uart-802154-blehci app on Raspberry Pi OS 11 with Linux kernel 5.15.	Use Linux kernel 5.10 or use btmgmt command.
1031607	The rcp-uart-802154.slcp project is running low on RAM on an MG1 part. Adding components may reduce the heap size below what is needed to support ECDH binding in CPC.	A workaround is to disable CPC security via the SL_CPC_SECURITY_ENABLED configuration.
1032183	Zigbeed does not support EZSP coex commands.	Support is planned in a future release.
1036622	There is a problem using cmake to build ot-cli using the multi-PAN RCP. It will be fixed in a future release.	Use the make build target as described in **AN1333** section 2.3.2.2.
1040127	CPC security fails to initialize for the rcp-uart-802154 and rcp-spi-802154 projects on mg13 and mg14 series parts.	Either disable CPC security, or add the mbedtls_entropy_adc component to the RCP image.

7.5   Deprecated Items

None

7.6   Removed Items

Removed in release 2.1.0.0

Pre-built ARM binaries for multiprotocol host applications are no longer distributed within the GSDK (cpcd, otbr-agent, zigbeed, Z3Gateway, etc). These should be built from sources on the target platform using the instructions in *AN1333: Running Zigbee, OpenThread, and Bluetooth Concurrently on a Linux Host with a Multiprotocol RCP.*

A copy of sl_cpc.h that was being included in the OpenThread sources as a convenience has been removed. This header file is placed in the standard system location when cpcd is installed.

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 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/2.1/.

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 **2ce3d3bf0**. 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 **181335224**. 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.

Simplicity Studio

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

IoT Portfolio	SW/HW	Quality	Support & Community
www.silabs.com/IoT	www.silabs.com/simplicity	www.silabs.com/quality	www.silabs.com/community

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

SILICON LABS

www.silabs.com