The Gecko Platform provides infrastructure support for applications developed with higher-level protocols, and it provides an interface with the underlying hardware. It is composed of the following modules:

CMSIS Device is a vendor-independent hardware abstraction layer for the Cortex®-M processor series.

Peripherals provides a complete peripheral API for all Silicon Labs EFM32, EZR32 and EFR32 MCUs and SoCs.

Drivers is the Gecko Platform driver library for EFM32, EZR32 and EFR32 on-chip peripherals. Drivers are typically DMA-based and use all available low-energy features.

Services includes common services such as NVM3 and Power Manager.

Common components are used throughout the SDKs.

Middleware includes the Capacitive Sensing Firmware Library and the GLIB graphics library, along with Micrium OS stacks like CAN/CANopen, File System, Networking and USB Device and Host.

Security includes mbed TLS and other security services.

Operating System includes Micrium OS Kernel as well as other things related to Operating Systems such as a CMSIS-RTOS2 layer.

The Gecko Bootloader is a code library configurable through Simplicity Studio’s IDE to generate bootloaders that can be used with a variety of Silicon Labs protocol stacks. The Gecko Bootloader can be used with EFM32 and EFR32 Series 1 and later devices.

Examples are example applications illustrating platform functionality.

Boards and External Devices cover supported hardware.

Other Gecko Platform Components regroups changes to documentation, project building and configuration, as well as any other aspects related to Gecko Platform.

RAIL (Radio Abstraction Interface Layer) provides a customizable radio interface layer that supports proprietary or standards-based wireless protocols. RAIL use by application protocols such as Silicon Labs Zigbee or Silicon Labs Connect is managed through the stack library. Direct RAIL use is exposed through the Flex SDK.

These release notes cover SDK version(s):

Gecko Platform 3.2.0.0 released June 16, 2021
## Contents

1 CMSIS Device ...................................................................................................................... 2
2 Peripherals .......................................................................................................................... 3
3 Drivers ............................................................................................................................... 4
4 Services .............................................................................................................................. 5
5 Common ............................................................................................................................. 7
6 Middleware .......................................................................................................................... 8
7 Security ............................................................................................................................... 9
8 Operating System .............................................................................................................. 11
9 Gecko Bootloader ............................................................................................................. 12
10 Examples .......................................................................................................................... 13
11 Boards and External Devices ............................................................................................ 14
12 Other Gecko Platform Software Components ..................................................................... 15
13 RAIL Library ..................................................................................................................... 17
1 CMSIS Device

1.1 New Items

Added in release 3.2.0.0
- Added support for the following new OPNs:
  - EFR32ZG13P531F512GM48

1.2 Improvements
None

1.3 Fixed Issues
None

1.4 Known Issues in the Current Release
None

1.5 Deprecated Items
None

1.6 Removed Items
None
2 Peripherals

2.1 New Items
None

2.2 Improvements

Changed in release 3.2.0.0
- Improved the disable and reset sequence for the VDAC peripheral, to make sure the disabling is properly reported in the peripheral’s registers, under all clocking conditions.

2.3 Fixed Issues

Fixed in release 3.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>698009</td>
<td>Fixed CMU_DPLLLock() hard faults by making sure the DPLL unit clock is enabled, under every condition.</td>
</tr>
<tr>
<td>580315</td>
<td>Fixed an issue when using OPA0 as VDAC channel 0 buffer on EFM32MG11.</td>
</tr>
</tbody>
</table>

2.4 Known Issues in the Current Release
None

2.5 Deprecated Items
None

2.6 Removed Items
None
3 Drivers

3.1 New Items

Added in release 3.2.0.0

- Introduced a Coulomb Counter driver to measure coulombs flowing through EFP's outputs.
- Added support for EFP Voltage scaling and Direct mode on EFR32xG22 and EFM32xG22.

3.2 Improvements

Changed in release 3.2.0.0

- On Series 2 chips (except EFR32xG21), voltage scaling in EM0 is now supported when EFP provides DECOUPLE.
- The initialization function of the gpiointerrupt driver can now be called multiple times without any problem.
- gpiointerrupt driver, which used to only support EXTI pins, now supports EM4WU pins.

3.3 Fixed Issues

Fixed in release 3.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>688457</td>
<td>Fixed UARTDRV LEUART port location in autogenerated code.</td>
</tr>
<tr>
<td>661098</td>
<td>Removed requirement on energy level from UARTDRV when aborting transfers.</td>
</tr>
<tr>
<td>520149</td>
<td>When creating SPIDRV instances using the Simplicity Studio Project Configurator, code to initialize the instance is not automatically generated and must instead be added manually.</td>
</tr>
<tr>
<td>674173</td>
<td>Fixed an issue in SPIDRV preventing the creation of a USART or EUSART SPIDRV instance in Simplicity Studio.</td>
</tr>
<tr>
<td>646417</td>
<td>Added EUSART signal support in emdrv's DMADV.</td>
</tr>
<tr>
<td>693334</td>
<td>Disabled RGBW PWM LED component for Series 2 devices because it is not supported on these parts.</td>
</tr>
</tbody>
</table>

3.4 Known Issues in the Current Release

None

3.5 Deprecated Items

None

3.6 Removed Items

None
4 Services

4.1 New Items

**Added in release 3.2.0.0**

- In IO Stream, added compiler option SL_IOSTREAM_USART_FLUSH_TX_BUFFER to flush USART buffer before returning from write.

4.2 Improvements

**Changed in release 3.2.0.0**

- The max interrupt disable counter now pauses when the device is in sleep, for projects that use the Power Manager.
- Improved accuracy of Sleep Timer where timers could expire one tick later than requested.
- The CLI now allows a command named ‘group’.
- The CLI now allows the same subgroup name in multiple groups.
- Added GPIO Interrupt initialization call to generated driver_init() function.
- Resolved a few MISRA rules violations in various Service components.
- Improved the documentation of the CLI by adding some missing information on argument separator and fixed a few typos.

4.3 Fixed Issues

**Fixed in release 3.2.0.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>656786</td>
<td>Fixed a bug in Sleep Timer where a variable was not declared using the volatile type qualifier and could lead to minor inaccuracies in timer timeouts when compiler optimization was enabled.</td>
</tr>
<tr>
<td>653374</td>
<td>The help is now displayed correctly for CLI commands added by application.</td>
</tr>
<tr>
<td>662722</td>
<td>EMU_EM23PresleepHook is now defined only once (in Power Manager), even when adding both EFP and Power Manager components to the project.</td>
</tr>
<tr>
<td>688784</td>
<td>Fixed an issue in the Power Manager related to creating a Sleep Timer requiring HF clock restored at timeout, but with a timeout smaller than the restore time. In some conditions, the HF clock restore was not done.</td>
</tr>
<tr>
<td>703231</td>
<td>Fixed an issue in Power Manager on Series 2 devices where we could get stuck waiting for HFXO to be ready when a hardware request on HFXO was removed in the middle of the restore process in Power Manager.</td>
</tr>
<tr>
<td>688482</td>
<td>Fixed a corner case issue in Power Manager that could lead to not restoring the HF clock when needed.</td>
</tr>
<tr>
<td>675253</td>
<td>Fixed an issue in Power Manager where, if the HFXO startup failed while actively waiting for it to be ready in a critical section, we could wait indefinitely.</td>
</tr>
<tr>
<td>677395</td>
<td>Fixed IO Stream-USART from signaling the semaphore before the kernel is started.</td>
</tr>
<tr>
<td>670648</td>
<td>Removed incorrect call to clear RXDATAV interrupt in IO Stream.</td>
</tr>
<tr>
<td>688275</td>
<td>Added option to configure LFXO timeout through Simplicity Studio.</td>
</tr>
<tr>
<td>695912</td>
<td>Fixed the symbols used to determine if the EM01GRPACLKC and EM01GRPBCLK are available, in device init.</td>
</tr>
<tr>
<td>697483</td>
<td>Fixed bug affecting CTUNE for HFXO on Series 2 devices.</td>
</tr>
<tr>
<td>638429</td>
<td>Fixed a broken link to Power Manager component documentation.</td>
</tr>
<tr>
<td>627936</td>
<td>Fixed some formatting and typos in the CLI documentation.</td>
</tr>
</tbody>
</table>

4.4 Known Issues in the Current Release

None
4.5 Deprecated Items
None

4.6 Removed Items
None
5 Common

5.1 New Items
None

5.2 Improvements
None

5.3 Fixed Issues
None

5.4 Known Issues in the Current Release
None

5.5 Deprecated Items
None

5.6 Removed Items
None
6 Middleware

6.1 New Items

Added in release 3.2.0.0

- CSLIB components are now released with the Gecko SDK.
- CSEN framework uses DMADRV rather than LDMA to avoid conflicts with other components.

6.2 Fixed Issues

None

6.3 Known Issues in the Current Release

None

6.4 Deprecated Items

None

6.5 Removed Items

None
7 Security

7.1 New Items

**Added in release 3.2.0.0**

- Updated the version of Mbed TLS to 2.26 with additional patches related to PSA Crypto support. The Mbed TLS base commit (https://github.com/ARMmbed/mbedttls) is 54650b389250728795005f928f9b8eb493ef011904d
- The TLS library is set up to use PSA Crypto APIs when the PSA Crypto UC component is included. The TLS library in this configuration is in Beta quality for this release.
- Added OS abstraction for the CMSIS RTOS2 API, which replaces the FreeRTOS and MicriumOS abstractions in SE Manager.
- In `psa_crypto_ecdh` and `psa_crypto_ecdsa` components, added auto-inclusion of fallback to standard Mbed TLS libraries when one or more non-accelerated ECC curves are included.
- There is a new entropy source implemented for this release. The new source can provide a device-unique non-volatile seed on devices which do not have other hardware entropy sources available. Due to the requirement of being able to store and update this seed, this implementation depends on the presence of NVM3. For more details, see the description of the component "mbed TLS Support for non-volatile entropy seed", which fulfills the requirement of having at least one entropy source available for CTR-DRBG and entropy collector capabilities.
- Added PSA Crypto components for PSA cipher, mac, aead, hash APIs and PSA ECC curves.
- Added support for the new PSA Crypto configuration options, MBEDTLS_PSA_CRYPTO_CONFIG, PSA_WANT_xxx and MBEDTLS_PSA_ACCEL_xxx, in the UC components for PSA Crypto. That is, the `psa_crypto` component is slimmed down, and a few new `psa_crypto_xxx` components have been added. This enables more fine-grained configuration of PSA Crypto, e.g. for excluding unused code, fallback to Mbed TLS library, and tune code size. Additional PSA Crypto configuration options for even better configuration granularity will be added in the near future releases.
- Added support for hardware-accelerated HMAC through PSA Crypto.

7.2 Improvements

**Changed in release 3.2.0.0**

- Changed the storage format for persistent keys stored through PSA Crypto, while retaining backwards compatibility for reading the old storage format. If you haven't stored any keys through PSA Crypto yet, you could save some code size by removing the storage format backwards compatibility support through setting the define SL_ITS_REMOVE_V1_HEADER_SUPPORT.
- Allow ITS files of size limited by NVM3_DEFAULT_MAX_OBJECT_SIZE in function `psa_its_set()`. In previous versions the file size was limited to 256 bytes. When using Mbed TLS with the default ECC acceleration settings, MBEDTLS_ECP_NIST_OPTIM will no longer be included when the only ECC curves requested are curves that can be accelerated by hardware. Previously, including MBEDTLS_ECP_NIST_OPTIM unnecessarily would cause an approximate 1.5 kB of code size overhead.
- When using Simplicity Configurator to configure Mbed TLS, the cipher modes CBC, CTR and OFB are no longer automatically included when adding AES support to the project. These modes can now be individually selected or deselected.
- In PSA Crypto API function `psa_key_derivation_setup()`, return PSA_ERROR_NOT_SUPPORTED from when no KDF is enabled.
- Reduced entropy stack usage by providing MBEDTLS_CTR_DRBG_MAX_SEED_INPUT configuration.
- Other code size optimizations.

7.3 Fixed Issues

**Fixed in release 3.2.0.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>679916</td>
<td>In the <code>psa_import_key()</code> function, allow import of un-clamped Montgomery keys.</td>
</tr>
<tr>
<td>679435</td>
<td>In the PSA Crypto HKDF component, always include HKDF fallback to Mbed TLS because the accelerated HKDF driver currently cannot operate standalone.</td>
</tr>
<tr>
<td>667205</td>
<td>In the <code>psa_import_key()</code> function, allow import of un-clamped Montgomery keys.</td>
</tr>
<tr>
<td>ID #</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>651954</td>
<td>Set MBEDTLS_MPI_MAX_SIZE correctly to support ECC curves secp384r1 and secp521r1 when selected with UC components mbedtls_ecc_secp384r1.slcc and mbedtls_ecc_secp521r1.slcc. In config-device-acceleration.h, add support for ECC curves secp384r1 and secp521r1 combined with ECDH and/or ECDSA for Series-2 'A' devices (treated like CURVE25519 and other non-accelerated curves).</td>
</tr>
<tr>
<td>479967</td>
<td>The APIs implemented by em_aes.c now correctly account for potentially unaligned buffers.</td>
</tr>
</tbody>
</table>

7.4 Known Issues in the Current Release

None

7.5 Deprecated Items

None

7.6 Removed Items

Removed in release 3.2.0.0

- Software fallback algorithms that were introduced as a workaround for a bug in EFR32xG21 SE Firmware versions before 1.2.2 are now removed by default. See the accompanying security advisory. This means ECDH operations and public key validity checks will by default return an error code on EFR32xG21 parts with SE firmware before 1.2.2. It is recommended to upgrade to the latest SE firmware instead of re-enabling the software workaround. However, in case these fallback methods are still needed, they can be turned back on by defining the symbol ‘SL_SE_SUPPORT_FW_PRIOR_TO_1_2_2’. 
8 Operating System

8.1 New Items

Added in release 3.2.0.0

- Added `configPRINT_STRING` to configuration file to enable Amazon FreeRTOS.
- Integrated File System driver for NOR Flash using `SPIDRV`.
  - Added macro to register NOR FS `SPIDRV` driver with Platform Manager.

8.2 Improvements

Changed in release 3.2.0.0

- Updated the version of FreeRTOS from 10.3.0 to 10.4.3 in order to fix a bug on Cortex-M33 device where the interrupts could be disabled when entering the first scheduled task.

8.3 Fixed Issues

Fixed in release 3.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>660929</td>
<td>CMSIS-RTOS2: <code>osMutexAcquire</code> will no longer ignore the <code>osMutexRecursive</code> attribute.</td>
</tr>
<tr>
<td>661498</td>
<td>In Micrium OS, calling a blocking pend on an OS object with interrupts disabled now returns an error.</td>
</tr>
<tr>
<td>704971</td>
<td>Fixed an issue in FreeRTOS's SLEEPtimer/Power Manager port where interrupts would be wrongly re-enabled.</td>
</tr>
<tr>
<td>687142</td>
<td>Modified the <code>tick_powermanager.c</code> FreeRTOS port file to use a standard macro provided by <code>portmacro.h</code> to call a <code>yield</code> instead of using Cortex-M specific macros that may not always be available.</td>
</tr>
<tr>
<td>687939</td>
<td>Fixed missing component dependency for Micrium OS Shell.</td>
</tr>
</tbody>
</table>

8.4 Known Issues in the Current Release

None

8.5 Deprecated Items

None

8.6 Removed Items

Removed in release 3.2.0.0

- Removed older FS NOR SPI driver.
9 Gecko Bootloader

9.1 New Items
None

9.2 Improvements
None

9.3 Fixed Issues
None

9.4 Known Issues in the Current Release
The Bootloader SPI-EZSP example only builds successfully for parts whose design supports a Serial Peripheral Interface (SPI). See the datasheet for your part to verify if it supports SPI.

9.5 Deprecated Items
None

9.6 Removed Items
None
10 Examples

10.1 New Items

Added in release 3.2.0.0

- Added an example application to demonstrate the Coulomb Counter. It uses Coulomb Counter driver API to read coulomb and NVM3 to store value in memory. The example also uses a CLI commands interface to read coulomb counter values.
- Added example applications for I2C with Micrium OS Kernel, FreeRTOS and no RTOS. These examples make use of the I2CSPM API's.
- Added example applications for the CLI with Micrium OS Kernel, FreeRTOS and no RTOS. These examples make use of the CLI API's.
- Added empty C++ project to platform examples.

10.2 Improvements

None

10.3 Fixed Issues

Fixed in release 3.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>663929</td>
<td>Fix a bug in PSA Crypto asymmetric key example.</td>
</tr>
<tr>
<td>682804</td>
<td>UARTDRV example is now running correctly on BRD4206A. Adding support for LFRCO oscillator was required, because BRD4206A does not have an LFXO oscillator.</td>
</tr>
<tr>
<td>711560</td>
<td>Fixed an issue with the metadata of some NVM3 examples.</td>
</tr>
<tr>
<td>703945</td>
<td>BRD4180 (A and B) have been removed from the list of the compatible boards for the SPIDRV sample apps because of a pin allocation conflict.</td>
</tr>
<tr>
<td>692903</td>
<td>Fixed the list of boards that are compatible with the dmadr2_baremetal example.</td>
</tr>
</tbody>
</table>

10.4 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/products/software.

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>664803</td>
<td>Se_manager and psa_crypto sample apps do not work correctly in Simplicity Studio 5's launch console.</td>
</tr>
</tbody>
</table>

Workaround: In the launch console, change the line terminator selection to None.

10.5 Deprecated Items

None

10.6 Removed Items

None
11 Boards and External Devices

11.1 New Items

**Added in release 3.2.0.0**

- Added support for the following new OPNs:
  - BG22-EK4108A/BRD4108A
  - BRD4183C
  - BRD4321A
  - WGM160P
- Added new display driver for the LPM013M126A in memlcd driver.
- Added support for CoreHW 12x12 PCB8 for our Bluetooth AoX Solutions.

11.2 Improvements

**Changed in release 3.2.0.0**

- Clarified that calling `sl_board_enable_sensor()` could have some side-effects on some boards (BRD4166A, BRD4184A, and BRD4184B).

11.3 Fixed Issues

**Fixed in release 3.2.0.0**

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>685556</td>
<td>Metadata about radio bands have been correctly added to BRD4206A board.</td>
</tr>
<tr>
<td>695875</td>
<td>Fixed PA value for MGM12P32F1024GA.</td>
</tr>
<tr>
<td>660844</td>
<td>Fixed CS and CLK route settings for I2S microphone for Series 2 devices boards.</td>
</tr>
</tbody>
</table>

11.4 Known Issues in the Current Release

None

11.5 Deprecated Items

**Deprecated in release 3.2.0.0**

- Functions to configure EFP regulators peak current have been deprecated: `sl_efp_set_voa_em01_peak_current`, `sl_efp_set_voa_em23_peak_current`, `sl_efp_set_vob_em01_peak_current`, `sl_efp_set_vob_em23_peak_current`.

11.6 Removed Items

None
12 Other Gecko Platform Software Components

12.1 New Items

Added in release 3.2.0.0

- Updated IAR compiler to version 8.50.9.
- Updated gcc compiler to version 10.2.
- Amazon FreeRTOS Libraries have been added to the platform SDK. They are provided with the Gecko SDK for running aws_test and validating the low-level implementation of IoT API. This module is not used internally by any Silicon Labs component. Be aware of the following security vulnerabilities:
  - CVE-2019-13120: "Amazon FreeRTOS up to and including v1.4.8 lacks length checking in prvProcessReceivedPublish, resulting in untargetable leakage of arbitrary memory contents on a device to an attacker. If an attacker has the authorization to send a malformed MQTT publish packet to an Amazon IoT Thing, which interacts with an associated vulnerable MQTT message in the application, specific circumstances could trigger this vulnerability."
  - CVE-2018-16524: "Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component allow information disclosure during parsing of TCP options in prvCheckOptions."
  - CVE-2018-16525: "Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component allow remote attackers to execute arbitrary code or leak information because of a Buffer Overflow during parsing of DNSSLLMNR packets in prvParseDNSReply."
  - CVE-2018-16526: "Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component allow remote attackers to leak information or execute arbitrary code because of a Buffer Overflow during generation of a protocol checksum in usGenerateProtocolChecksum and prvProcessIPPacket."
  - CVE-2018-16527: "Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component allow information disclosure during parsing of ICMP packets in prvProcessICMPPacket."
  - CVE-2018-16598: "An issue was discovered in Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component. In xProcessReceivedUDPQueue, any received DNS response is accepted, without confirming it matches a sent DNS request."
  - CVE-2018-16599: "An issue was discovered in Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component. Out of bounds memory access during parsing of NBNS packets in prvTreatNBNS can be used for information disclosure."
  - CVE-2018-16600: "An issue was discovered in Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component. Out of bounds memory access during parsing of ARP packets in eARPPacket can be used for information disclosure."
  - CVE-2018-16601: "An issue was discovered in Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component. A crafted IP header triggers a full memory space copy in prvProcessIPPacket, leading to denial of service and possibly remote code execution."
  - CVE-2018-16602: "An issue was discovered in Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component. Out of bounds memory access during parsing of DHCP responses in prvProcessDHCPReplies can be used for information disclosure."
  - CVE-2018-16603: "An issue was discovered in Amazon Web Services (AWS) FreeRTOS through 1.3.1, FreeRTOS up to V10.0.1 (with FreeRTOS+TCP), and WITTENSTEIN WHIS Connect middleware TCP/IP component. Out of bounds access to TCP source and destination port fields in xProcessReceivedTCPQueue can leak data back to an attacker."
  - CVE-2019-13120: "Amazon FreeRTOS up to and including v1.4.8 lacks length checking in prvProcessReceivedPublish, resulting in untargetable leakage of arbitrary memory contents on a device to an attacker. If an attacker has the authorization to send a malformed MQTT publish packet to an Amazon IoT Thing, which interacts with an associated vulnerable MQTT message in the application, specific circumstances could trigger this vulnerability."

- Added Unity Test framework to Gecko SDK delivery to enable Amazon FreeRTOS.
- Added bare-metal (no RTOS) support to SystemView.
- Added FreeRTOS support to SystemView.
12.2 Improvements
None

12.3 Fixed Issues
Fixed in release 3.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>690494</td>
<td>Added Linker section in GCC linker script for Segger RTT in order to support the block/buffer auto-detection.</td>
</tr>
<tr>
<td>653931</td>
<td>Fixed tiny-printf library compilation issue with GCC9.</td>
</tr>
<tr>
<td>668857</td>
<td>Fixed issue when creating a project for Series 0/1 device on IAR 8.5 and a corrupt file error would happen.</td>
</tr>
</tbody>
</table>

12.4 Known Issues in the Current Release
None

12.5 Deprecated Items
None

12.6 Removed Items
None
13 RAIL Library

13.1 New Items

Added in release 3.2.0.0

- Added RAIL_StartScheduledCcaCsmaTx and RAIL_StartScheduledCcaLbtTx APIs to allow applications to schedule a CSMA/LBT transmit.
- Added support for a new RAIL_EVENT_ZWAVE_LR_ACK_REQUEST_COMMAND, triggered on the reception of a Z-Wave Long range packet with acknowledgment request bit set, following which the application must call RAIL_ZWAVE_SetLrAckData to populate the fields of the Z-Wave Long range acknowledgement packet.
- Added RAIL_TX_OPTION_RESEND to allow re-transmitting the packet most recently loaded into the Transmit FIFO. This can be used in combination with RAIL_SetNextTxRepeat() on supported platforms to repeatedly transmit the same packet.
- Added new RAIL_SetNextTxRepeat() API and RAIL_TxRepeatConfig_t to allow configuration of repeated transmits triggered by an initial transmit.
- Added ability to configure TxToTx state transition time via RAIL_StateTiming_t::txToTx and RAIL_SetStateTiming(). This time is generally used between an autoACK transmit and a user transmit that was pending. It is also used during repeated transmits by default, but can be overridden by the RAIL_TxRepeatConfig_t::delayOrHop or RAIL_BLE_TxRepeatConfig_t::delayOrHop configuration.
- Added ability to hop channels during repeated transmits in RAIL_TxRepeatConfig_t using RAIL_TX_REPEAT_OPTION_HOP.
- Added BLE-specific hooks for repeated transmits with channel hopping in RAIL_BLE_SetNextTxRepeat() and RAIL_BLE_TxRepeatConfig_t.
- The “RAIL Utility, Callbacks” component can now be configured to not provide the RAILCb_AssertFailed() function for situations where the application wants to provide its own implementation.

13.2 Improvements

Changed in release 3.2.0.0

- Added support for the RAIL_TX_POWER_MODE_*_HIGHEST options in the RAIL_GetTxPowerCurve() function.
- Changed when RAIL_EVENT_IEEE802154_DATA_REQUEST_COMMAND event is issued to better facilitate support for 802.15.4E-2012 Enhanced ACKing and reduce time spent in the event handler. The event is now issued *after* receiving the Auxiliary Security Header (if present) in the MAC Header of the incoming frame, and for MAC Command frames, after receiving the MAC Command byte (which may be encrypted). This change is *not* backwards-compatible with prior releases for Enhanced ACK requesting frames, but is for Immediate ACK requesting frames. Use of RAIL_IEEE802154_EnableEarlyFramePending() is no longer required to support Enhanced ACKing; the notion of early frame pending notification has also shifted to after the Auxiliary Security Header for Enhanced ACK requesting frames.
- Changed RAIL Timer Synchronization over sleep on the EFR32xG21 to use the RTCC instead of the PRORTC to reduce current consumption in EM2.
- The posting of RAIL_EVENT_TX_ABORTED now occurs before the PA ramps down, at the same time other transmit completion events get posted.
- Allow RAIL_ZWAVE_ReceiveBeam to be run on US LR regions instead of always returning an error.

13.3 Fixed Issues

Fixed in release 3.2.0.0

<table>
<thead>
<tr>
<th>ID #</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>456701</td>
<td>Fixed an issue on EFR32xG1x parts where calling RAIL_Init() with the MSC-&gt;CTRL_CLKDISFAULTEN bit set would cause a bus fault.</td>
</tr>
<tr>
<td>654600</td>
<td>On EFR32XG21, a watchdog has been added to terminate an RSSI averaging operation in case the RAIL_EVENT_RSSI_AVERAGE_DONE event does not occur.</td>
</tr>
<tr>
<td>ID #</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>655541</td>
<td>Fixed an issue on EFR32XG22 and later where packet filtering might be incomplete in FEC-enabled radio configurations causing good packets to be improperly dropped. Note that if packet filtering fails close to the end of an otherwise successfully received packet, the packet may be flagged RAIL_RX_PACKET_READY_CRC_ERROR rather than RAIL_RX_PACKET_ABORT_FILTERED. Note this issue is still present on earlier chips.</td>
</tr>
<tr>
<td>665705</td>
<td>Fixed an issue where a transmit with RAIL_TX_OPTION_SYNC_WORD_ID 1 to use SYNC2 would improperly indicate SYNC1 was used in the packet trace appended information.</td>
</tr>
<tr>
<td>666275</td>
<td>Fixed potential delays when using RAIL's channel hopping or duty cycling features in EM1P mode on the EFR32xG22 and newer parts.</td>
</tr>
<tr>
<td>667103</td>
<td>Fixed RAIL_ReadRxFifo() to behave as documented when passed NULL dataPtr: the data is thrown away rather than copied out.</td>
</tr>
<tr>
<td>671817</td>
<td>Fixed an issue when switching between certain radio configurations (for example, Z-Wave) where use of RAIL_TX_OPTION_REMOVE_CRC can become permanently stuck.</td>
</tr>
<tr>
<td>673333</td>
<td>Fixed an issue with RAIL_TX_OPTION_WAIT_FOR_ACK transmits where an RX overflow during the ACK wait period would silently abort the ACK timer resulting in no RAIL_EVENT_RX_ACK_TIMEOUT being generated.</td>
</tr>
<tr>
<td>682032</td>
<td>Fixed an issue where setting RAIL_SCHEDULED_TX_DURING_RX_ABORT_TX for a scheduled transmit could cause subsequent non-scheduled transmits to be blocked.</td>
</tr>
<tr>
<td>696665</td>
<td>Fixed an issue in RAIL_ConvertRawToDbm for PAs that use piecewise-linear line segment fit, where the minimum raw power level was incorrectly compared to the minimum deci-dBm value.</td>
</tr>
<tr>
<td>697097</td>
<td>Fixed a rare situation where a premature RAIL_EVENT_TX_BLOCKED event might occur when auto-ACK is enabled and a scheduled transmit using RAIL_SCHEDULED_TX_DURING_RX_ABORT_TX is pending when an erroneous packet is received.</td>
</tr>
<tr>
<td>699890</td>
<td>Fixed missing C++ compatibility in the “RAIL Utility, Init” component’s generated header file.</td>
</tr>
<tr>
<td>701604</td>
<td>Fixed an issue where using Rx_Channel_Hopping with channels in different frequency bands would cause RAIL to assert.</td>
</tr>
<tr>
<td>703788</td>
<td>Fixed an issue on EFR32xG2x devices where RAIL would not allow you to initialize the radio with voltage scaling enabled even though this is supported on these devices.</td>
</tr>
<tr>
<td>705595</td>
<td>Fixed an issue where custom PA curves provided through the PA Module were not respected when building Silicon Labs Zigbee applications.</td>
</tr>
<tr>
<td>708511</td>
<td>Fixed possible RAIL_ASSERT_FAILED_UNEXPECTED_STATE_TX_FIFO when RAIL_SetTxFifo() is called after the transmit FIFO had been filled enough to cause its write offset to wrap.</td>
</tr>
</tbody>
</table>

### 13.4 Known Issues in the Current Release

None

### 13.5 Deprecated Items

None

### 13.6 Removed Items

None
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.

Note: This content may contain offensive terminology that is now obsolete. Silicon Labs is replacing these terms with inclusive language wherever possible. For more information, visit www.silabs.com/about-us/inclusive-lexicon-project

Trademark Information
Silicon Laboratories Inc.*, Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR, Ember®, Energy Micro, Energy Micro logo and combinations thereof, “the world’s most energy friendly microcontrollers”, Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, Gecko OS, Gecko OS Studio, ISOmodem®, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, 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.