

Mira version 2.9.0

Release Notes



Table of Contents

2.9.0	1
Bugfixes.....	1
Features	1
Zephyr and nRF Connect SDK support.....	1
Manual FOTA triggering.....	1
Frontend configuration getter	1
User configurable RNG for MiraMesh.....	1
Updates and improvements.....	1
Examples are open source	1
Flash API example.....	2
Border Gateway UDP-forwarding always on.....	2
Border Gateway takes configuration file as CLI argument	2
“CERTIFICATE” is renamed to “FACTORY_CONFIG”	2
Memory usage prints.....	2
“make miradocs” starts a web server	2
Known issues.....	2
USB radio stick bootloader doesn’t enter DFU mode after update	2
Using gcc option -B results in infinite loop	2
mira_flash_write() hangs if writing to page 0.....	3
Higher current consumption when running with 2.6.2 root.....	3

2.9.0

Bugfixes

Features

Zephyr and nRF Connect SDK support

MiraMesh can now be used with the embedded operating system Zephyr. To enable this, MiraMesh uses the nRF connect SDK from Nordic Semiconductor. As in previous versions, multiprotocol & BLE is supported, but now with the use of the SoftDevice Controller (SDC) from nRF connect SDK.

How to use MiraMesh together with Zephyr in the nRF Connect SDK is described in [this example](#). The integration code is available [here](#).

MiraMesh support for nRF connect SDK and the SDC is exclusive to Zephyr; MiraOS and FreeRTOS + MiraMesh do not support the new SDK. However, they have continued support for the nRF5-SDK and the associated SoftDevice.

Zephyr + MiraMesh is not supported on the NXP mkw41z chipset.

Manual FOTA triggering

The Mira FOTA API has been extended to include a function to send a FOTA request message: `mira_fota_force_request()`. This will make the node immediately ask its parent if it has a new FOTA image to download. Mira sends these request messages internally on a five minute timer, so this new function can be used to make FOTA clients more responsive to new images being present in the parent. This can be useful reducing test times, for example. Keep in mind that a FOTA request will result in packets being sent to the parent, which will cost energy and add load to the parent link.

Frontend configuration getter

A new function has been introduced in the Mira diagnostics API: `mira_diag_get_active_frontend_cfg()`. This function returns the current active frontend configuration.

User configurable RNG for MiraMesh

`miramesh_init`'s configuration now has a new field for providing a different cryptographically secure random number generator. This is mostly useful for integrating with an RTOS' APIs to the RNG hardware, otherwise MiraMesh will have exclusive access to it.

Updates and improvements

Examples are open source

All example applications are now open source and their repo is [here](#) at GitHub. They're released with the MIT license.

Flash API example

A new MiraOS example has been added: `flash_write`. This example uses the Mira flash API to write some data into the FLASH memory area. It uses Mira processes to write asynchronously and verifies the resulting flash contents.

Border Gateway UDP-forwarding always on

The Border Gateway will now always forward UDP messages. The command line option to enable it is deprecated and does nothing.

Border Gateway takes configuration file as CLI argument

The border gateway executable can now take a configuration file as an argument. This enables protection of sensitive parameters, such as encryption key and PAN ID. These parameters should now be specified in a file, with appropriate access rights, and specified to the border gateway executable. The CLI options `--key` and `--panid` are still present, but are deprecated and should not be used as they are considered insecure.

“CERTIFICATE” is renamed to “FACTORY_CONFIG”

The linker section called “CERTIFICATE” is now “FACTORY_CONFIG”. The old name can still be used for backwards compatibility. Similarly, the MiraMesh configuration field “certificate” has been renamed to “factory_config”. The old name is available for backwards compatibility.

Memory usage prints

The build system now prints memory usage on compilation.

“make miradocs” starts a web server

To make the search function work when reading the Mira documentation, the “miradocs” make target will start a python web server and then try to launch the browser at its pages. It will run until its process is stopped with CTRL-C in the command line.

Known issues

USB radio stick bootloader doesn't enter DFU mode after update

When starting the gateway with an older radio stick using USB it may need to update the USB DFU bootloader. The update will complete, but it will not enter the DFU mode. It needs to be in DFU mode to be able to also load the gateway radio firmware, so this hangs the gateway host program. A workaround is to manually set the USB radio stick into DFU mode by running the script `reset_to_dfu.py`, located in the tools folder.

Using gcc option -B results in infinite loop

When compiling an application with `libmira.a` using gcc option `-B`, compilation gets stuck in an infinite loop, compiling the application over and over again.

mira_flash_write() hangs if writing to page 0

Instead of throwing an error, the API hangs indefinitely if one tries to write to the invalid block 0.

Higher current consumption when running with 2.6.2 root

During compatibility testing it was observed that 2.9.0 nodes draw slightly more current when having a 2.6.2 root node. In the test case an increase of 1-2 uA was observed for devices normally drawing around 18 uA. It is therefore recommended to run the same Mira version on all devices in the network.