Getting Started with Arrow Shield96 Trusted Board and EmSPARK Security Suite

Getting Started Guide

Date Dec 4, 2020 | Version 1.2

TABLE OF CONTENTS

1. INTRODUCTION

The **Getting Started Guide** provides an overview of the steps to use the Shield96 Trusted Platform and the EmSPARK Security Suite package. The following tasks will be performed:

- Install the filesystem on the board
- Download the CoreLockr Kit (SDK included with EmSPARK Security Suite Kit) with toolchain and examples and install it in a Linux development environment (not on the Shield96)

Short Product Description

The Shield96 Trusted Platform is a fused, locked SBC board preloaded with Sequitur Labs EmSPARK Security Suite 2.0. The board implements secure boot and a minimum, standard version of Linux preloaded to the QSPI present on the board. The board does not include a filesystem because it does not ship with an SD card. The Out of the Box experience includes the procedure to automatically download the Linux filesystem from the cloud. Once a SD Card is inserted and the board is connected to the Internet via Ethernet the board automatically connects to a Sandbox Platform provided by Sequitur Labs which downloads securely to the board the latest firmware and the filesystem. The procedure includes the registration of the board to AWS IoT Core since AWS IoT Core service is used for the firmware management. The Sandbox is built as a service on AWS. The Sandbox has further built-in functionality to provide various customers firmware update for the device lifecycle.

The EMSPARK Security Suite for the Shield96 Trusted Board provides secure boot, a preloaded TEE (Trusted Execution Environment) a number of Trusted Applications (TAs) and an SDK that includes the CoreLockr APIs abstracting all TA secure applications in easy to use C APIs in Linux.

Out of the box the board provides a secure enclave through the TrustZone\TEE and an immutable key pair that generates a CSR for the creation of a unique Device Certificates which acts as the device ID for communication with cloud applications – in this case with AWS IoT Core. The CoreLockr SDK provides APIs to generate more CSRs and thus to create other device IDs. The private keys cannot be extracted from the secure enclave.

1.1. Prerequisites

The following hardware and software are required to use the EmSPARK Suite:

- To install the filesystem on the Shield96
  - Shield96 Trusted Platform: Board rev: v1.4, Jumper for J3 must be in place.
  - 4GB or larger SD Card to install the filesystem (U1 and U3 cards should NOT be used due to HW Limitations)
  - Ethernet Network connection on the board
  - Linux Host computer connected to the board
- To develop and build applications using the CoreLockr APIs
  - CoreLockr Kit downloaded from Arrow Electronics
  - Linux development environment to extract the CoreLockr Kit

2. INSTALLATION PROCEDURE

This section describes steps to install the filesystem on the board, to use the CoreLockr APIs in a Linux development environment and to apply a firmware update. The **Troubleshooting** section describes how to resolve commonly seen problems during the filesystem installation.

2.1. Install Filesystem on the Board

To install the filesystem on the board, the process consists of these steps detailed in the following sections:

- Insert a blank SD Card on the board (No partitions defined)
- Connect the board to the host computer and start a serial console
- Power up the board
- Ensure the board has connectivity to the Internet

The board will register itself with AWS, configure the SD Card, and install the base filesystem when network connection is available and a SD Card is present.

2.1.1. Insert SD Card on the Board

The SD Card must be 4GB or larger. Insert a blank SD Card on the board. The installation process will partition, format and install the root filesystem on the card.

Important Notes:

- Inserting a SD Card that has a filesystem will prevent the installation process.
- U1 and U3 cards should NOT be used due to HW Limitations.

2.1.2. Connect the Board and Start the Console

To start the serial console which is the TEE console where occasionally the Secure World prints output messages:

- Connect a micro USB cable to J10 (debug) micro USB port on the board
- Connect the host machine to the serial port
    - 115200 bps
    - No parity
    - 8 bits
    - 1 stop bit
    - No flow control
- Connect a micro USB cable to the PC/power micro USB port on the board, if you would like to power the board separate from the console connection.

2.1.3. Power up the Board

When the board boots for the first time it will do the following:

- Boot to the Linux RAM FS
- Check for network connection
- Check for SD Card

The initial setup checks for network connection and SD Card. If the checks succeed, then the board will register itself to AWS and retrieve the Root filesystem.

This process is automated but can be observed from the console connection.

The device is now able to run the example applications and modify the root filesystem.

The serial terminal will print output such as the following:

```
Checking: mmcblk1
Getparts:  /dev/mmcblk1 status:  2
Create partitions
...
eth0 at:  192.168.x.xxx
...
Enroll device at AWS ...
...
Enroll customer/device at AWS IoT ...
...
Retrieve device rootfs information from AWS ...
...
Payload server:  screechowl.seqlabs.com
Service port:    2270
Root filesystem: Shield96RootFS.tar.gz
...
Extract rootfs ...
```

Finally:

```
Welcome to Sequitur Labs CoreTEE
root@seqlabs_coretee:~#
```

2.1.4. Secure Boot Mode - Starting and Using the System

To start using the system on the board, start the console. After the board starts up:

- Access `username:password = root:root`.
- The board is configured to acquire an IP address using DHCP

The board is ready for your configuration:

- Required configuration:
  - The date on the board must be current for certificate management. When the board is used offline, the date must be configured. When the board is configured for remote access, verify that the date is current.
- Optional configuration:
  - Configure additional user(s). In addition to `root`, users in the `coretee` group have access to the TEE clients and can execute applications using the CoreLockr APIs. If users are created to execute the example applications, then create the `coretee` group if not already created and add users to this group.
  - Configure the board for remote access. The board is configured to acquire an IP address using DHCP. SSH is set up in the filesystem.

The board set up is complete and ready to execute operations in the TEE and execute applications using the EmSPARK Suite CoreLockr APIs.

Note: `ssh` does not start automatically on boot. One way start `ssh` is to execute

```
    /etc/init.d/~S50sshd start
```

To enable it to start up with boot, rename `/etc/init.d/~S50sshd` to `/etc/init.d/S50sshd`.

2.2. Download the CoreLockr Kit and Extract the Contents in Linux Development Environment

Download and extract the contents of the CoreLockr Kit in Linux development environment (not on the Shield96).

The `CoreLockr Kit arrow_corelockr_package_YYYY-MM-DD_n.tar.gz` contains everything needed to build applications using the CoreLockr APIs:

- APIs and Normal World Assets
- CoreLockr APIs (C libraries)
- OpenSSL Crypto Engine
- Code Examples
- Toolchain and Client API

If building in a 64-bit Linux development environment, install libraries to enable 32-bit support:

- Platform agnostic library list to install the necessary libraries for the toolchain to run
  - 32bit glibc
  - 32bit libstdc++
  - 32bit zlib (compression library)
- Debian based systems
  - dpkg --add-architecture i386
  - apt-get update
  - apt-get install libc6:i386 libstdc++6:i386 zlib1g:i386
- Red Hat based systems
  - dnf install glibc.i686 libstdc++.i686 zlib.i686

2.3. Update Firmware

The EmSPARK Secure Boot is a fully redundant system on the QSPI. The board runs two boot stacks, initially the primary stack is Stack A, and the non-primary stack is Stack B. The non-primary stack is the failover.

The firmware update mechanisms allow updating components of the Secure Boot. For this Shield96 Trustboard, an update of CoreTEE is available and needs to be applied to enable additional features in the TEE. The `update_package` directory in the CoreLockr package contains the components needed to execute the update. Please see `update_package/README.txt` for instructions to apply the update. `README.txt` also describes an overview of the failover and update process.

3. TROUBLESHOOTING

This section describes issues that may occur during the installation of the filesystem on the SD Card inserted on board and how to resolve them. Issues generally occur because the SD Card type is not supported, the SD Card is not empty or the board is not connected to the Internet. The following points illustrate the errors.

Issue: The SD Card is inserted on the board, but the board does not recognize it

Messages on the board terminal are similar to these:

```
checking mmcblk1
>>> getparts status:  3
Please insert a micro-sd card
>>> getparts status [loop]:  3
Please insert a micro-sd card
```

Procedure: This issue may be caused by the SD Card type. Ensure that the SD Card is not U1 or U3. Such cards should not be used due to HW Limitations.

Issue: The board terminal is reporting a kernel panic and does not install the filesystem

Messages on the board terminal are similar to these:

```
Function entered at [<c0113df5>] from [<c01141d9>]
Function entered at [<c01141d9>] from [<c01141e5>]
---[ end Kernel panic - not syncing: Attempted to kill init!
exitcode=0x00000100
 ]---
```

Procedure: This issue is produced when the SD Card is not empty. Ensure that the SD Card has no partitions, even if the partitions have not content. To remove the partitions, use a tool such as fdisk to delete all the partitions on the SD Card.

Issue: The board terminal reports "leasefail: not found" and the filesystem does not install

The board terminal prints messages like these:

```
Setup mmcblk1p1
udhcpc: started, v1.28.4
udhcpc: sending discover
udhcpc: sending discover
udhcpc: sending discover
/usr/share/udhcpc/default.script: exec: line 7:
/usr/share/udhcpc/sample.leasefail: not found
```

Procedure: This errors occur when the board is not connected to the Internet. Ensure that the board is connected to the Internet.

CHANGE HISTORY

DATE	VERSION	RESPONSIBLE	DESCRIPTION
July 15, 2020	1.0	Julia Narvaez	Produced document for release.
October 9, 2020	1.1.	Julia Narvaez	Expanded Introduction. Added Troubleshooting section.
December 4, 2020	1.2	Julia Narvaez	Updated board revision