

M2 QRNG ACCEL GEN 2 – Usage Manual

Table of Contents

- 1. Scope 2
 - 1.1. Overview 2
- 2. Referenced Documents 2
- 3. Document History 2
- 4. M.2 QRNG Acceleration BOARD Setup Steps 3
 - 4.1. Overview 3
 - 4.2. Diagnostics Test 5

1. Scope

1.1. Overview

This document provides guidance on installing the EigenQ M2 QRNG Acceleration Board and validating random number generation.

2. Referenced Documents

None

3. Document History

DATE	VERSION	AUTHOR	DESCRIPTION
11-Jun-26	1.0	Pedro	Initial Draft

4. M.2 QRNG Acceleration BOARD Setup Steps

Guidance for a technician to setup, install and use the M2 QRNG Acceleration Board. To complete this task the following is required:

- Ubuntu Server 24.x
- M2 QRNG Acceleration Board **GEN 2**
- [eigenq-eqdma-dkms.deb](#) file downloaded and present on the O.S.
- [qrng-xdma-accel.tar](#) file downloaded and present on the O.S.

4.1. Overview

This section guides you through setting up and validating the EigenQ M2 QRNG Accel Board on Ubuntu 24.x: confirm prerequisites, install the hardware, deploy EigenQ services, build and load the driver, and verify service health and entropy generation.

1. Verify that the target machine is running Ubuntu 24.x.
2. Install the M2 QRNG ACCEL Board into the machine (turn off the machine, insert the device, and turn it on).
3. Confirm that the M2 QRNG ACCEL Board was recognized by the OS. Execute the following command in a terminal:

```
lspci | grep -i accelerators
```

Expected result:

```
86:00.0 Processing accelerators: EigenQ, Inc. QMA Board PCIe Gen2
```

4. If the command above does not return the EigenQ device description, run the command below to update the PCI IDs database.

```
sudo update-pciids
```

Run the `lspci` command again and confirm that EigenQ is listed.

5. Install the dependencies needed to install the EigenQ EQDMA driver.

```
sudo apt-get update && sudo apt-get install -y dkms make gcc tpm2-tools libtss2-dev
```

6. Install the EigenQ EQDMA Driver.

```
sudo dpkg -i ./eigenq-eqdma-dkms.deb
```

You may be asked for a password to enroll the MOK key. Please remember this password for Step 9, as you will need it after rebooting the machine.

7. Unpack and install the EigenQ Accelerator Service.

```
tar -xf qrng-xdma-accel.tar
```

```
cd qrng-xdma-accel
```

```
chmod +x ./install.sh
```

```
sudo ./install.sh
```

8. Check the loading status of the driver.

```
sudo dkms status
```

Expected result:

```
eigenq-eqdma/1.0, 6.8.0-84-generic, x86_64: installed
```

9. Reboot.

```
sudo reboot
```

You may be asked to enroll MOK, please select the option to enroll MOK and use the password you may have set on step 6.

10. Confirm the driver was properly loaded.

```
ls /dev/eq*
```

Expected result:

```
/dev/eqdma0_c2h_0 /dev/eqdma0_events_ .... (more devices will be listed, if you  
have eqdma0 devices driver was loaded successfully)
```

If no devices are returned, reboot the machine and run the command again.

11. Now if the driver is loaded, let's start the service.

```
sudo systemctl start gq-accel-qrng
```

12. Check if the service is working.

```
sudo systemctl status gq-accel-qrng
```

13. Verifying logs.

```
sudo journalctl -u gq-accel-qrng.service -f
```

4.2. Diagnostics Test

- Please download the script [pqu_diag.sh](#)

You can use the EigenQ [pqu_diag.sh](#) script to run diagnostics on the M2 QRNG Acceleration Board.

Different tests will be run there and it may take a while. To do that you can download the diagnostics script and then run it as root.

```
unzip pqu_diag.zip
```

```
./pqu_diag.sh
```

With the execution of this script, OpenSSL 3.6.1 will be installed on the O.S and you may use it in case you want to use this OpenSSL version that is also connected to our M2 Board to generate keys, certificates, among others. To do that after running the `pqu_diag.sh` script, you need only to load the version for your current user and start using it:

```
export PATH=/app/openssl_bin/bin:$PATH
```

```
export LD_LIBRARY_PATH=/app/openssl_bin/lib64:$LD_LIBRARY_PATH
```

```
openssl version -a
```

Confirm it will return the 3.6.1 version and you can use the Post Quantum Ciphers together with our EigenQ QRNG Acceleration Board.

Command example to generate mDSA keys:

```
openssl genpkey -algorithm ML-DSA-87 -out mdsa87_private.pem
```

```
openssl pkey -in mdsa87_private.pem -pubout -out mdsa87_public.pem
```