PCIe QRNG ACCEL GEN 3 - Usage Manual

Table of Contents

1. Scope 2
   1.1. Overview 2
2. Referenced Documents 2
3. Document History 2
4. PCIe QRNG Acceleration Board Setup Steps 3
   4.1. Overview 3
   4.2. Diagnostics Test 4
5. Extras 4
   5.1. OpenSSL 3.6.1 Integrated with EigenQ QMA 4

© 2026 EigenQ, Inc

1. Scope

1.1. Overview

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

2. Referenced Documents

None

3. Document History

DATE	VERSION	AUTHOR	DESCRIPTION
27-Aug-26	1.1	Pedro	Editorial and procedural corrections
28-Aug-26	1.2	Pedro	Updates on OpenSSL commands and fixes to log path.

© 2026 EigenQ, Inc

4. PCIe QRNG Acceleration Board Setup Steps

Guidance for a technician to set up, install, and use the PCIe QRNG Acceleration Board. The following are required:

- Ubuntu Server 24.x
- PCIe QRNG Acceleration Board with Gen 3 firmware
- eigenq-firm-gen03-packages.zip downloaded and available on the OS

4.1. Overview

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

1. Verify that the target machine is running Ubuntu 24.x.
2. Install the PCIe QRNG Acceleration Board in the machine. Shut down the machine, install the board, and then power the machine on.
3. Confirm that the PCIe QRNG Acceleration Board is recognized by the OS. Run the following command in a terminal:

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

Example expected result (PCI bus address may vary):

```
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 following command to update the PCI ID database:

```
sudo update-pciids
```

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

5. Install the EigenQ EQDMA drivers.

```
unzip eigenq-firm-gen03-packages.zip
sudo apt-get update && sudo apt-get install -y ./*.deb
```

You may be asked to set a password for MOK enrollment. Keep this password for Step 7, where it may be required after the machine reboots.

6. Check the driver installation status.

```
sudo dkms status
```

   Example expected result (kernel version may vary):

```
eigenq-eq_qma/3.0, 6.8.0-138-generic, x86_64: installed
```


© 2026 EigenQ, Inc

7. Reboot.

   ```
   sudo reboot
   ```

   If prompted during startup, select the option to enroll the MOK key and enter the password set in Step 5.

8. Confirm that the driver loaded successfully.

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

   Expected result:

   ```
   /dev/eq_qma0
   ```

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

9. If the driver is loaded, start the EigenQ service.

   Check whether the service is running:

   ```
   sudo eq-qes status
   ```

   If the service is not running, initialize it:

   ```
   sudo eq-qes init
   ```

10. Verify that the service is operating correctly.

    ```
    sudo eq-qes monitor
    ```

Press Ctrl+C to stop monitoring.

4.2. Diagnostics Test

Use the EigenQ diagnostics script to run diagnostics on the PCIe QRNG Acceleration Board. First create a directory for the eigenq logs to be stored. For now let's go with /home/eigenq/logs.

```
mkdir -p /home/eigenq/logs
```

Then we can start the diagnostics script, run:

```
bash /usr/lib/eq-diagnostics.sh
```

5. Extras

5.1. OpenSSL 3.6.1 Integrated with EigenQ QMA

The following steps install an OpenSSL 3.6.1 build integrated with the EigenQ Board. This build uses /dev/hwrng as its primary entropy source, where the EigenQ service provides entropy. Please download openssl_bin_ub24.tar and transfer it to the tmp folder. You can use wget if you would like.

© 2026 EigenQ, Inc

```
tar -xf openssl_bin_ub24.tar
sudo mkdir /app
sudo mv openssl_bin /app/openssl_bin
sudo chmod 644 /dev/hwrng
```

To use this OpenSSL version in the current shell session, run:

Become the root user:

```
sudo su
```

Set the openssl on the linux path:

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

Confirm that the OpenSSL version is 3.6.1:

```
openssl version
```

Verify that OpenSSL is integrated with the EigenQ Board.

Stop the EigenQ service:

```
sudo eq-qes stop
```

Clear the hardware RNG buffer:

```
timeout 5s cat /dev/hwrng | pv > /dev/null
```

Run the following test:

```
timeout 5s openssl rand -hex 32
echo $?
```

Expected result: exit code 124. This indicates that the command did not complete within the 5-second timeout while the EigenQ service was stopped.

Initialize the EigenQ service:

```
sudo eq-qes init
```

Run the following test:

```
timeout 5s openssl rand -hex 32
echo $?
```

Expected result: the command returns the requested random data and exit code 0.
Re-run the diagnostics script to include the OpenSSL benchmark test:

```
bash /usr/lib/eq-diagnostics.sh
```


© 2026 EigenQ, Inc