SP 800-90B Non-Proprietary Public Use Document

Nokia Jitter Entropy (JENT)

Document Version 1.3

Firmware Version 1.0

Nokia Corporation

600-700 Mountain Avenue
Murray Hill, New Jersey 07974-0636

USA

February 6, 2023

Revision History

Version	Change
1.0	Initial Draft
1.1	Updated diagram and other edits
1.2	Added cutoff values for RCT, APT and lag prediction. Changed software to firmware.
1.3	Added OS

Table of Contents

Description

The Nokia JENT 1.0 is a non-physical (NP) entropy source based on the open-source CPU Jitter RNG version 3.4.0 firmware binary. It was tested on the platforms listed in **Table *1*** running on the Marvel Armada XP MV78460 processor with the Wind River Linux LTS 21 operating system. The Non-IID track was used to calculate min-entropy.

Table 1 Tested Platforms

Platform	Processor	Operating System
PSS-24x	Marvel Armada XP MV78460	Wind River Linux LTS 21
PSS-16II	Marvel Armada XP MV78460	Wind River Linux LTS 21
PSS-32	Marvel Armada XP MV78460	Wind River Linux LTS 21
PSIM	Marvel Armada XP MV78460	Wind River Linux LTS 21
PSS-8	Marvel Armada XP MV78460	Wind River Linux LTS 21
PSI-4L	Marvel Armada XP MV78460	Wind River Linux LTS 21
PSI-8L	Marvel Armada XP MV78460	Wind River Linux LTS 21
PSS-8x	Marvel Armada XP MV78460	Wind River Linux LTS 21
PSS-12x	Marvel Armada XP MV78460	Wind River Linux LTS 21

Security Boundary

The Nokia JENT entropy source model is shown in **Figure *1***. The noise source is based on time delta variations in execution of hashing and memory access operations. The security boundary of the Nokia JENT entropy source includes the Jitter RNG firmware binary that is compiled from the firmware library (Jitterentropy-library-3.4.0). The output of the Nokia JENT entropy source is used to seed an SP 800-90A compliant DRBG (outside the entropy source security boundary).

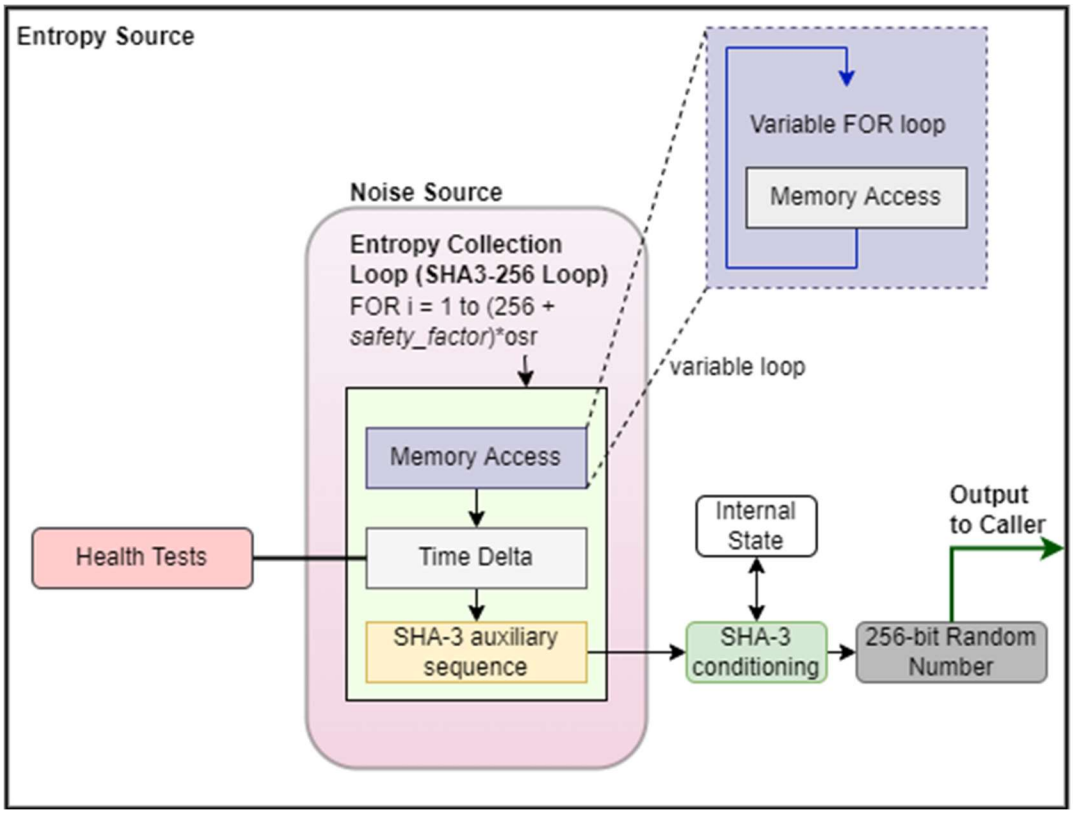

Figure 1 JENT Entropy Source Model

Operating Conditions

The operating conditions for the Nokia JENT entropy source are listed in Table 2.

Table 2 Operating Conditions

Platform	Processor	Processor Clock Speed (MHz)	Processor Temperature	Processor Voltage
PSS-24x	Marvel Armada XP MV78460	1600	0 – 105°C	0.85 – 0.95 V core voltage
PSS-16II	Marvel Armada XP MV78460	1600	0 – 105°C	0.85 – 0.95 V core voltage
PSS-32	Marvel Armada XP MV78460	1600	0 – 105°C	0.85 – 0.95 V core voltage
PSIM	Marvel Armada XP MV78460	1600	0 – 105°C	0.85 – 0.95 V core voltage
PSS-8	Marvel Armada XP MV78460	1333	0 – 105°C	0.85 – 0.95 V core voltage
PSI-4L	Marvel Armada XP MV78460	1600	0 – 105°C	0.85 – 0.95 V core voltage
PSI-8L	Marvel Armada XP MV78460	1600	0 – 105°C	0.85 – 0.95 V core voltage
PSS-8x	Marvel Armada XP MV78460	1600	0 – 105°C	0.85 – 0.95 V core voltage
PSS-12x	Marvel Armada XP MV78460	1600	0 – 105°C	0.85 – 0.95 V core voltage

Configuration Settings

The configuration settings for the Nokia JENT entropy source are shown in Table 3.

Table 3 Configuration Settings

Parameter	Value	Description
JENT_RANDOM_MEMACCESS	True	Enabled
JENT_MEMORY_BITS	17	Determines the memory block size
JENT_CONF_ENABLE_INTERNAL_TIMER	Disabled	Disables internal timer
JENT_CONF_DISABLE_LOOP_SHUFFLE	Enabled	Disables the pseudo-random hash and memory access loop count

Physical Security Mechanisms

The physical security for this entropy source is provided by the underlying hardware platforms, as this entropy source is of non-physical nature.

Conceptual Interfaces

The Nokia JENT entropy source provides the jent_read_entropy function, which returns a requested amount of entropy to the the caller. This function corresponds to the GetEntropy interface from SP 800-90B. The Nokia JENT entropy source also provides scripts to obtain raw, digitized outputs from the noise source for use in validation testing. As is compliant with SP 800-90B, on-demand health tests may be initiated by reallocating a new CPU Jitter RNG handle.

Min-Entropy Rate

The H_submitter is 1/2 bit per 64-bits of noise sample. The min-entropy rate at the output of the Nokia JENT entropy source is 256 bits of entropy per 256-bit data block, or 1 bit/bit.

Health Tests

The Nokia JENT entropy source implements the health tests complying with SP 800-90B section 4.4, including both continuous and start-up tests.

The Nokia JENT entropy source implements the following continuous health tests:

- Stuck Test
- Repetition Count Test (RCT)
- Adaptive Proportion Test (APT)
- Lag Predictor Test

The stuck test calculates the first, second and third discrete derivative of the time to be processed by the hash. Only if all three values are non-zero, the received time delta is considered to be non-stuck.

The RCT and APT are described in SP 800-90B section 4.4. These tests are applied with $\alpha = 2^{-30}$ and corresponding allowed cutoff values.

- Repetition Count Test conforming to section 4.4.1
    - $H = 1/osr$
    - $\alpha = 2^{-30}$
    - $osr = 2$
    - Cutoff value $C = 60$
- Adaptive Proportion test conforming to SP 800-90B section 4.4.2
    - $W = 512$
    - $H = 1/osr$ bit of entropy per 8-bit sample
    - alpha value of $\alpha = 2^{-30}$
    - Cutoff value $C = 422$

The lag predictor test is a vendor-defined conditional test based on SP 800-90B section 6.3.8 that is designed to detect a known failure mode where the result becomes mostly deterministic. The lag predictor test is configured in this entropy source with the following parameters:

- $\alpha = 2^{-22}$
- Window size: $window\_size = 131072$
- Lag history size: $lag\_history\_size = 8$
- Global Cutoff $= InverseBinomialCDF = CRITBINOM\left(n = window_{size} - lag_{histo\_size}; p = 2^{-\frac{1}{osr}}; 1 - \alpha\right) = 93504$
- Local cutoff = 75

When a health test fails, the API call to generate random numbers jent_read_entropy(3) informs the caller about the failure with the following error codes:

- -2 = RCT failure
- -3 = APT failure
- -5 = Lag predictor test failure

The Nokia JENT entropy source applies a start-up health test that runs the continuous RCT and APT health tests on 1,024 noise source samples. The collected noise source samples are not reused for the generation of random numbers.

The Nokia JENT entropy source also supports on-demand testing of the noise source output via reallocation of a new CPU Jitter RNG handle, which re-executes the start-up health tests.

Maintenance

There are no specific maintenance requirements for the Nokia JENT entropy source.

Required Testing

These steps are performed to validate the Nokia JENT entropy source:

- Obtain 1,000,000 samples of raw noise data by running the invoke_testing.sh script included in the jitterentropy-library-3.4.0 package. Process the data by the SP 800-90B tool and ensure that the entropy rate is at least as high as the H-submitter.
- Obtain the restart noise data by running the same invoke_testing.sh script and process it by the SP800-90B tool to verify:
    - the sanity test to apply to the noise restart data passes, and
    - the minimum of the row-wise and column-wise entropy rate shall not be less than half of the initial entropy estimate of the noise source (per section 3.1.3 of SP 800-90B). The initial entropy estimate of the noise source is calculated as H_I = min($H_{original}$, $n \times H_{bitstring}$, $H_{submitter}$), where $H_{original}$ and $nxH_{bitstring}$ are obtained from the calculation in 1 above.