Entropy Harvesting from Physical Sensors

Christine Hennebert
CEA/LETI
christine.hennebert@cea.fr

Hicham Hossayni
CEA/LETI
hicham.hossayni@cea.fr

Cédric Lauradoux
INRIA
cedric.lauradoux@inria.fr

ABSTRACT

Finding entropy sources is a major issue to design non-deterministic random generators for headless devices. Our goal is to evaluate a collection of sensors (*e.g.* thermometer, accelerometer, magnetometer) as potential sources of entropy. A challenge in the analysis of these sources is the estimation of min-entropy. We have followed the NIST recommendations to obtain pessimistic estimations from the dataset collected during our campaign of experiments. The most interesting sensors of our study are: the accelerometer, the magnetometer, the vibration sensor and the internal clock. Contrary to previous results, we observe far less entropy than it was expected before. Other sensors which measures phenomena with high inertia such as the temperature or air pressure provide very little entropy.

Categories and Subject Descriptors

G.3 [**Probability and statistics**]: Random number generation; K.6.5 [**Security and Protection**]: [Miscellaneous]

General Terms

Non-deterministic random bit generator

Keywords

Entropy sources, *min-entropy* estimators and sensors.

1. INTRODUCTION

Context – Non-deterministic random bit generators (NDRBGs) play an important role on security: key generation, nonces or masking to name a few. The most common failure in the design of an NDRGB is related to the entropy sources ([11, 13] for instance). The use of bad entropy sources have ruined the security of many systems [14]. All the standards including the RFC 4086 [8], the German BSI report AIS 20 and 31 [1], the NIST report SP800-133 [4] emphasize the need to analyze the sources and to have as many as possible. On headless devices, entropy sources are scarce. We evaluate the main available resources: physical sensors.

Contribution – We address two key-points in the design of an NDRBG. The first point is the evaluation of the physical sensors. We plug different sensors on a node and measure physical quantities (*e.g.* acceleration, temperature or air pressure) and on-MCU features (clocks desynchronization) during three experiments. We focused on three modes of operation to evaluate the influence of the environment: stability, dynamic and attack mode (saturation).

The second point addressed is the discussion on *min-entropy* evaluation. The performance of an entropy source is quantified by *min-entropy*. We consider that in many works the analysis of this quantity is not rigorous. In previous results, the *min-entropy* computed for an observation of the source can be overestimated because the data sequence is biaised. To handle this problem, we have followed the process recommended by the NIST. To our knowledge, we are the first to do so.

Outline – The paper is organized as follows. In Section 2, we provide the definitions related to (N)DRBGs and entropy. The related work of this paper is then given (Section 3). We describe the platforms in Section 4 and the experiments performed to carry out our research in Section 5. The analysis of these datasets is made in Section 6 for the NIST tools and in Section 7 for our results.

2. DEFINITIONS

Before going any further, we provide four definitions to remove any ambiguity in the acronyms or terms used in the rest of the paper. We have chosen to follow the notations used by the NIST in its latest recommendations [5, 6].

Definition 1 (Deterministic RBGs)
A deterministic random bit generator (DRBG) produces a pseudo-random sequence of bits from an initial secret value called a seed (and, perhaps additional input). A DRBG is often called a pseudo-random bit (or number) generator.

Definition 2 (Non-deterministic RBGs)
A non-deterministic random bit generator (NDRBG) produces (when working properly) outputs that have full entropy. Also called a true random bit (or number) generator in the literature.

Definition 3 (Entropy sources)
An entropy source provides random bitstring. There is no

WiSec'13, April 17–19, 2013, Budapest, Hungary.

assumption that the bitstring are output in accordance with the uniform distribution.

Two types of entropy sources are distinguished in practice: *dedicated and opportunistic sources.*

DEDICATED SOURCES – These sources are designed only to produce randomness and are often included in secure embedded systems (*e.g. smartcards*). Many hardware sources have been proposed: semiconductor thermal noise [15] or ring oscillators [12]. More examples can be found in the proceedings of CHES. These sources can be very efficient and benefit from tamper resistance features. However, dedicated sources can be very expensive in term of cost or integration time. *We consider the case in which adding dedicated hardware is not affordable.*

OPPORTUNISTIC SOURCES – Because dedicated sources are not always an option, designers have started to divert some peripherals from their primary purpose to seek entropy. The first proposition made was to exploit hard-drive timing [7]. Since then many other sources have been exploited: mouse, keyboard stroke [16], performance counters. These specific sources are found in commodity computers but they are not available on headless devices (which leads to attacks against the Linux random number generator [13]). *On these devices, physical sensors are a main option for entropy.*

To compare entropy sources, we need to have some metrics for security and efficiency. The Shannon entropy and the min-entropy are the main metrics for security.

Definition 4 (Entropy)
Let S be a source of entropy which produces values over $\mathcal{X}$. We associate to these outputs a random variable X and x denotes a value sampled from X. The Shannon entropy and min-entropy for the source are given by:

$$H(X) = -\sum_x \mathbf{Pr}[X = x] \log_2 \mathbf{Pr}[X = x], \quad (1)$$

$$H_\infty(X) = -\log_2(\max_x \mathbf{Pr}[X = x]). \quad (2)$$

The Shannon entropy (Equation 1) was the first obvious choice to study sources. However, the Shannon entropy fails to measure the capability of an adversary. In [2], the authors show that min-entropy (Equation 2) is a better metric. It is also the metric recommended by the NIST [6].

Remark 1 *Determining the min-entropy of a source requires the prior knowledge of the corresponding probability distribution. The term min-entropy of a source S is used in an ambiguous way in many papers. Let us define the random variable Y associated to an observation of S. This observation of S is defined as a set of n values sampled from S and belonging to the set $\mathcal{Y} \subseteq \mathcal{X}$. At a first sight, computing $H_\infty(Y)$ can not be considered as a good estimator of $H_\infty(X)$.*

There are two solutions to solve this deadlock. First, we are able to make a complete or unbiased observation. It means that $\mathcal{Y} = \mathcal{X}$ and n is large enough such that the probability distribution of X and Y are very closed. Both conditions seem hard to achieve. Second, we use an estimator of $H_\infty(X)$ based on the observation Y. We denote this value $\mathbf{Est}_{H_\infty}(Y)$ or $\mathbf{Est}(Y)$ for short. To our knowledge, we are the first to speak of this problem. We have chosen the second solution. The NIST has proposed several tools to evaluate $\mathbf{Est}(Y)$. They are described in Section 6.

In the rest of the paper, we use the notation $H_\infty(Y)$ to denote the biased min-entropy estimator, $\mathbf{Est}(Y)$ a better estimator and $H(Y)$ an estimator of Shannon entropy.

3. RELATED WORK

Many propositions of NDRBGs and entropy sources exist for systems having strong user interactions (Fortuna [9]). Unfortunately, they are not portable to headless devices. Three works are particularly close to our paper: TinyRNG from Francillon and Castelluccia [10], the evaluation of iPhone's sensor by Lauradoux, Ponge and Roeck [17] and the work of Voris, Saxena and T. Halevi [19] published at WiSec 2011.

TinyRNG [10] is a modification of Fortuna [9] suitable for sensor nodes. In their work, the authors of [10] have investigated the opportunity to use channel errors as a source of randomness. Their motivation was that bit errors during communications are difficult to observe/predict and manipulate if certain conditions are met on the signal quality. However, they did not estimate how much entropy can be extracted from this source. We have considered other sources of entropy and provide an estimation of min-entropy.

In [17], Lauradoux, Ponge and Roeck have proposed a new online estimator for Shannon entropy. The different estimators for Shannon entropy are compared on datasets produced by the different sensors (GPS, accelerometer and compass) available on an iPhone. Compared to our work, they focused on the health test needed to check the correctness of sources rather than the estimation of min-entropy.

Voris, Saxena and T. Halevi [19] have proposed a complete design of NDRBG for RFID. Their work implements a proposition of Barak and S. Halevi [3] at CHES 2003 using an accelerometer and a randomness extractor based on Toeplitz matrix. The authors of [19] have conducted an intensive campaign of experiments using a WISP node and a smartphone to evaluate the performance of an accelerometer as entropy source. Their work is very closed to ours but two criticisms can be made. First, they have computed $H_\infty(Y)$ which can lead to an overestimation of min-entropy without giving any information on their observation size and diversity. Despite working on different hardware than in [19], our results for $H_\infty(Y)$ are of the same order. However, the results obtained using $\mathbf{Est}(Y)$ are different (see Section 7).

Second, the authors of [19] have adapted a design built for dedicated sources to an opportunistic source. Their NDRBG follows the advices of Barak and S. Halevi [3] under which it is mandatory that the adversary cannot influence the entropy source. Unfortunately, Voris, Saxena and T. Halevi [19] were unable to prove that an adversary cannot influence an accelerometer. Our opportunistic sources do not need such a proof because we rely on the security model of Barak and S. Halevi [2]. They also discuss the merits (Figure 3 in [19]) of other sensors which are considered in our work. However, their discussions are only qualitative without experiments to support them.

4. PLATFORM DESCRIPTION

4.1 Hardware

Our experiments are based on two sensor nodes: eZ430-RF2500 and Zolertia Z1 platform. The main characteristics of our sensors are summarized in Table 1.

TEXAS INSTRUMENT EZ430-RF2500 – The eZ430-RF2500

is a development platform proposed by Texas Instrument. It uses the MSP430F22x4 processor paired with the CC2500 multi-channel RF transceiver designed for low-power wireless applications. The MSP430 integrates two independent clocks: VLO (very-low-frequency oscillator) and DCO (digitally controlled oscillator). The shift between these two clocks is exploited for entropy generation by TI in [20].

ZOLERTIA Z1 (ULTRA-LOW POWER 16-BIT MCU 16MHZ) – It is a low power wireless module compliant with IEEE 802.15.4 and Zigbee protocols. Its core architecture is based upon the MSP430 micro-controller and a TI CC2420 radio transceiver. Two sensors are embedded on the Z1: (a) a digital temperature sensor with $\pm 0.5°$C accuracy (in the range 25°C to 85°C), and (b) a digital accelerometer (3-Axis).

In addition to the internal thermometer and accelerometer, the Z1 node supports external sensors: two analog sensors (Phidget) and one digital sensor (Ziglet) can be used concurrently. We describe below the Phidget sensors used.

Vibration Sensor: it embeds a piezoelectric transducer. As the transducer shifts from the mechanical neutral axis, bending creates strain within the piezoelectric element and generates voltage.

Magnetic sensor: this is a ratiometric Hall-effect sensor which provides a voltage output that is proportional to the applied magnetic field.

Motion Sensor: it detects changes in infrared radiation that occurs when there is an object/person's movement with a different temperature from the surroundings. This sensor is also characterized by a narrow sensing area.

Gas pressure sensor: it measures absolute gas pressure from 20 to 250 kPa (2.9 to 36.3 psi) with a maximum error of $\pm 1.5\%$. It is suitable for measuring vacuum, or atmospheric pressure; it can also be used as a crude barometer.

Temperature & humidity sensor: it measures the relative humidity (RH) from 10% to 95% with a typical error of $\pm 2\%$RH at 55% RH. It also measures ambient temperature in the range of -30°C to +80°C with a typical error of $\pm 0.75°$C in the range 0°C to 80°C.

4.2 Software

For the Z1 nodes, we used the Contiki OS to develop the experiment programs. For the eZ430-RF2500, we used IAR Embedded Workbench Kickstart to build and debug embedded applications for MSP430.

5. EXPERIMENTS

We place the aforementioned sensors in different scenarios and collect data. Below, we describe the different experiments made. We used a sampling frequency of 50Hz.

5.1 Stability Mode

Our experiments have started with retrieving sensor data in a stable situation without any influence, which significantly reduces the abrupt changes in the measured phenomenas. In these experiments, we set the sensors in stable conditions and away of all disturbances for a day.

5.2 Saturation Mode

We try to put the sensors in extreme conditions. These experiments differ from one sensor to another, depending on the phenomenon being measured, and which can sometimes be very difficult to implement. We have not attempt to saturate the humidity and the gas pressure sensors because the conditions needed can damage the circuitry of the node.

ACCELEROMETER – To saturate the accelerometer, we have used a playground roundabout.

VIBRATION SENSOR – We used a Power Plate found in a gym club which can generate from 10 to 50 vibrations per second.

TEMPERATURE SENSOR – There are two ways to saturate a temperature sensor, the first one, by increasing the temperature, but in the absence of a precise temperature controller we may burn the circuit. The second solution is to lower the temperature. We used the second solution and put the temperature sensor in the freezer at -18°C.

MOTION SENSOR – The principle of this sensor is to detect temperature differences between an object in its environment. To saturate this sensor, we stick it on a human body in such way that it detects only the body heat.

MAGNETIC SENSOR – We put it between two magnets.

5.3 Dynamic mode

In addition to the stability and saturation mode, we made several measurements in other situations. For instance, we used all the sensors during a road trip. The road we took going through the top of some mountains up to 1176 meters above sea level, which can vary the atmospheric pressure and the temperature of the car. Curves and shift provided a good challenge for the accelerometers. And vibration made the old car a good ground for experimenting vibration sensor.

6. NIST METHODOLOGY

Recently, the NIST has released a first draft of recommendations in order to evaluate both the entropy source model and the meaning of entropy [6]. In [6], the NIST gives several definitions and tests applied to data in order to evaluate the amount of entropy provided by the sources. These tests are not to be mistaken with the NIST statistical tests suite [18] which are made to test if a bit-string is conformed or not to some characteristics of an independent and identically distributed (i.i.d.) sequence. Applying the NIST statistical tests suite to the data produced by our sensors is not particularly significant because it is quite obvious that sequences obtained are biased.

The methodology [6] described to estimate min-entropy is different if the probability distribution of S is known or not. If the source S is i.i.d., some specific tests are applied. Otherwise, five tests are used to compute different statistic on the digitized samples and to provide information about the structure of the data. Their application to non-i.i.d. data will produce an underestimation of the contained min-entropy. No information is assumed on the probability distribution associated to the output of S. Each test reveals information about the unknown distribution given a statistical measurement. The entropy is estimated by minimizing over this set of distribution. For the following tests, a confidence level of 95% is considered.

The tests are based on the property that the expectation of a random value in the probability domain is analogous to the mean of a statistical set in the statistical domain. So, finding the probability distribution that characterizes the data structure is to find the parameter that the probability is equal to the most pessimistic estimate of the mean of the observed series that is the low-bound of the confidence interval. There are five tests: the frequency test ($\mathbf{Est}_1(Y)$),

Sensor	Output type	Accuracy	Sensibility	Max. Consump.	Operating Temp. Min, Max
Vibration	Ratiometric	—	—	400 μA	-20, 70 °C
Magnetic	Ratiometric	± 0.5%	[0, 1000] (Gauss)	2 mA	-20, 85 °C
Motion	Ratiometric	—	5 meters	15 μA	-20, 85 °C
Gas pressure	Ratiometric	± 1.5 %	20,250 Kilopascals	5 mA	0, 85 °C
Temperature	Ratiometric	± 2 °C	$[-40, 100]$ °C	3.9 mA	-30, 80 °C
Humidity	Ratiometric	± 5 %	10%, 90% RH	3.9 mA	-30, 80 °C

Table 1: Summary of sensors characteristics.

the collision test ($\mathbf{Est}_2(Y)$), the partial collection ($\mathbf{Est}_3(Y)$), the compression test ($\mathbf{Est}_4(Y)$) and the Markov test.

The five previous tests operate on relatively small values. This limitation is clearly an issue for the analyst who needs to study sequence of large values. Due to this problem, we were unsuccessful with the Markov test because our data were too large to have a reasonable computation time.

7. ANALYSIS

During our campaign, we were unable to gather the same amount of data in all experiments. We succeed to gather large amount of data (near a million) in the stability mode but not in the dynamic one (near 50000). The explanation for this discrepancy is the complexity to maintain the sensors into the dynamic state or the necessity to preserve their integrity. It results that our confidence level is higher in the stability mode than in the saturation/dynamic mode.

Another important element of our analysis is the number of distinct states observed during the data collection. The maximum and minimum values are given for each sensor in Table 2. This table shows that our dataset are biased except for the TI's clocks desynchronization for which we observe almost all the possible states. For opportunistic sources, we believe that biased observations cannot be avoided. It emphasizes the need to have pessimistic min-entropy estimators.

	Distinct states	
Sensor	min.	max.
Accelerometer.X	1	489
Accelerometer.Y	1	1024
Accelerometer.Z	11	11
Internal thermometer	5	622
External thermometer	175	697
Vibration	44	1018
Magnetic	9	216
Gas pressure	48	92
Motion	227	1815
Humidity	229	238
TI's clocks	63843	63843

Table 2: The min-max number of states by sensors.

Comparison of min entropy tools – In Figure 1, we have compared the different measures of min-entropy and Shannon entropy for an experiment with the vibration sensor. The four estimators are more pessimist than $H_\infty(Y)$. The frequency estimation $\mathbf{Est}_1(Y)$ is the closest to the biased $H_\infty(Y)$ value due to its nature. The most pessimist tests are the partial collection test $\mathbf{Est}_3(Y)$ and the compression test $\mathbf{Est}_4(Y)$. These trends are verified for all the sensors.

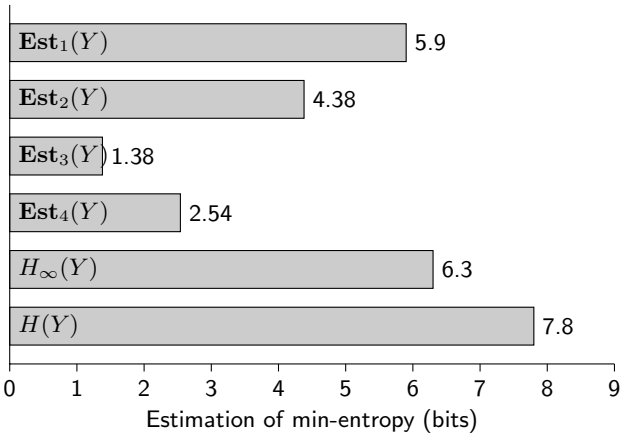

Figure 1: Comparison of the estimators for the vibration sensor on a power plate at 30Hz.

In Figure 2, we have validated the estimators on sensors for which min-entropy results can be (almost) predicted. As a witness, we have used the internal and external thermometers. Temperature is a physical phenomenon with high inertia. In the stable mode, both sensors have measured the lab temperature. Due to the precision limits of the sensors, there is only a few variation in the outputs of both sensors. Therefore, all the entropy can only come from the characteristics of the electronic components. The biased estimator $H_\infty(Y)$ evaluates that there is 3.55 bits of min-entropy for the internal sensor. This value looks excessive. The NIST estimators $\mathbf{Est}_3(Y)$ and $\mathbf{Est}_4(Y)$ considered that the min-entropy is very close to zero.

The case of the external thermometer is a bit different. The input signal of the analog-to-digital converter (ADC) is more noisy for the external thermometer than for the internal one. The wires and impedances connecting the sensor to the ADC are responsible for this noise. As a consequence, the estimation is higher with the external sensors.

To synthesize the results of all the estimators, we have considered the minimum of the four NIST estimators:

$$\mathbf{Est}(Y) = \min(\mathbf{Est}_1(Y), \mathbf{Est}_2(Y), \mathbf{Est}_3(Y), \mathbf{Est}_4(Y)).$$

General comments – The results obtained for all the sensors are given in Table 3. In all the modes, the sensors measuring physical phenomena with a high inertia have very little min-entropy. This is the case for the thermometer, the gas pressure and the humidity. The most interesting sources

Sensor	Stability mode		Dynamic mode		Saturation mode		Entropy behaviour under adversarial control
	$\mathbf{Est}(Y)$	$H_\infty(Y)$	$\mathbf{Est}(Y)$	$H_\infty(Y)$	$\mathbf{Est}(Y)$	$H_\infty(Y)$	
Internal thermometer	≈0	≈0	0	0	0.00	0.00	Nullify entropy
External thermometer	0.05	4.79			0.00	5.75	Nullify entropy
Humidity	0.16	6.21					Unknown
Gas Pressure	0.29	2.66					Unknown
Magnetic	0.62	2.98			0.02	0.94	Decrease entropy
Motion	0.11	5.5			0.65	5.51	Inconclusive
Vibration	0.17	2.87	0.48	6.12	3.02	7.85	Increase entropy
Accelerometer.X	0.22	1.38	1.34	4.90	0.00	0.00	Nullify entropy
Accelerometer.Y	0.42	1.05	2.39	7.75	0,00	0.00	Nullify entropy
Accelerometer.Z	0.36	1.72	2.83	6.15	0.33	0.70	Decrease entropy
TI's clocks	1	14,29					Unknown

Table 3: Summary of results.

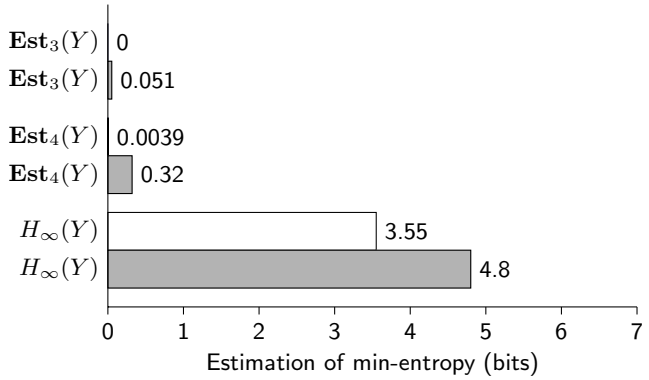

Figure 2: Observations for the internal (white) and external (grey) temperature sensor.

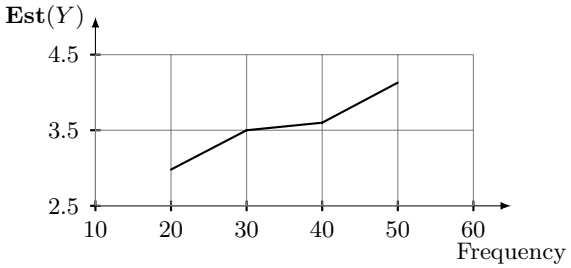

Figure 3: Effect of the power plate frequency on the vibration sensor.

of entropy are the vibration sensor, the accelerometer, the TI's clocks and the magnetic sensor. For these sensors, the mode has a significant impact on the estimation.

Magnetic sensor – In the stability mode, the best sources of min-entropy are the TI's clocks and the magnetic sensor. In our lab condition, it appears that the background noise is enough to create variation on the sensor.

Despite the small number of states in the output space, the magnetic sensor returns 0.62 bits of min-entropy. To saturate the magnetic sensor, we had put it between two magnets. At the first time, the magnets were in the same direction ($\mathbf{Est}(Y) = 0.37$), and at a second time we reversed the direction of one magnet. We were able to almost saturate the sensor in this second time ($\mathbf{Est}(Y) = 0.02$), by forcing it to return only nine values for all samples. This allows us to conclude that the magnetic sensor can be reliable as a source of entropy if we define a threshold value for the values output by the sensor. This health test can detect saturation attacks or Faraday cage attacks.

Vibration sensor – In the stability mode, the results for the vibration sensor are not very promising. They are better for the dynamic mode (see Table 3). Our attempt to saturate this sensor on a power plate was a failure. We never saturate the sensor and increase significantly its min-entropy. The Figure 3 shows the effect of the power plate working frequency on the min-entropy. This sensor seems particularly difficult to saturate and our power plate does not allow us to increase the frequency in order to find the saturation point. The best option for an adversary is to keep the sensor in the stability mode.

Accelerometer – This sensor provides little entropy in the stability mode, this is reflected by the relatively limited number of states observed. The dynamic mode is quite better but we obtain very good results on the power plate. The min-entropy is almost greater to one bit even if the frequency seems to affect the axis differently.

In the saturation mode, a playground roundabout was enough to saturate two axis (X and Y) of the accelerometer but, as it turns on a plan surface, it fails to saturate Z (0.7 bits of min-entropy). The results of this experiments are given in Table 3.

We have also manually manipulated the accelerometer with a lasso to generate an helical trajectory. The results with this last test are not included in these tables as the datasets collected did not fulfill the requirements to pass the NIST min-entropy tests. Surprisingly, the results show that we can completely saturate the X and Y axes, forcing them to return the maximum acceleration value (156), and because of irregularities in our movements, the Z axis returning few values with more dominant maximum value.

8. CONCLUSION

Our study has shown that the best candidates to produce entropy are the accelerometer, the vibration sensor and the magnetic sensor. On this point, our results agree with the work of Voris, Saxena and T. Halevi [19]. But we have

shown that the amount of min-entropy collected is clearly overestimated in [19].

Generating a cryptographic key requires to collect many samples from opportunistic sources. The study of these sources is very challenging as they have unknown probability distribution and we can only obtain biased observations of their realization. The NIST estimators have improved the situation but many open problems are still around. We would like to mention the need to have a conditional mean entropy estimator. Indeed, using different sources to feed an NDRBG requires to know the correlation between the sources. In most of the NDRBGs used in practice, the sources are assumed to be independent. However, this hypothesis seems incorrect and a conditional mean entropy estimator would be a precious tool to improve sources analysis.

Acknowledgement

This research work was supported by the ANR VERSO ARESA2 project and the European FP7 project BUTLER, under contract no. 287901.

9. REFERENCES

[1] Functionality classes and evaluation methodology for physical random number generators. Technical Report AIS 31, Bonn, Germany, September 2001. `http://tinyurl.com/9wv8dtj`.

[2] B. Barak and S. Halevi. A model and architecture for pseudo-random generation with applications to /dev/random. In *ACM Conference on Computer and Communications Security - CCS 2005*, pages 203–212, Alexandria, VA, USA, November 2005. ACM.

[3] B. Barak, R. Shaltiel, and E. Tromer. True Random Number Generators Secure in a Changing Environment. In *Cryptographic Hardware and Embedded Systems - CHES 2003*, Lecture Notes in Computer Science 2779, pages 166–180, Cologne, Germany, September 2003. Springer.

[4] E. Barker and A. Roginsky. Recommendation for Cryptographic Key Generation. NIST Special Publication 800-133, July 2011.

[5] E. Barker and A. Roginsky. Recommendation for Random Bit Generator (RBG) Constructions. Draft NIST Special Publication 800-90C, Aug. 2012.

[6] E. Barker and A. Roginsky. Recommendation for the Entropy Sources Used for Random Bit Generation. Draft NIST Special Publication 800-90B, Jan. 2012.

[7] D. Davis, R. Ihaka, and P. Fenstermacher. Cryptographic Randomness from Air Turbulence in Disk Drives. In *Advances in Cryptology - CRYPTO '94*, Lecture Notes in Computer Science 839, pages 114–120, Santa Barbara, CA, USA, August 1994. Springer.

[8] D. Eastlake, J. Schiller, and S. Crocker. Randomness Requirements for Security, June 2005. RFC 4086.

[9] N. Ferguson and B. Schneier. *Practical Cryptography*. 2003.

[10] A. Francillon and C. Castelluccia. TinyRNG, A Cryptographic Random Number Generator for Wireless Sensor Network Nodes. In *Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, IEEE WiOpt 2007*, Limassol, Cyprus, April 2007. IEEE.

[11] I. Goldberg and D. Wagner. Randomness and the Netscape browser. *Dr. Dobbs Journal*, Jannuary 1996.

[12] J. D. Golic. New Methods for Digital Generation and Postprocessing of Random Data. *IEEE Trans. Computers*, 55(10):1217–1229, 2006.

[13] Z. Gutterman, B. Pinkas, and T. Reinman. Analysis of the Linux Random Number Generator. In *IEEE Symposium on Security and Privacy - S&P 2006*, pages 371–385, Berkeley, CA, USA, May 2006. IEEE Computer Society.

[14] N. Heninger, Z. Durumeric, E. Wustrow, and J. A. Halderman. Mining Your Ps and Qs: Detection of WidespreadWeak Keys in Network Devices. In *USENIX Security Symposium*, pages 205–219, Bellevue, WA, USA, July 2012. USENIX.

[15] B. Jun and P. Kocher. The Intel random number generator. *Cryptography Research Inc. white paper*, 1999.

[16] J. Kelsey. Entropy sources. In *NIST RNG Workshop*, July 2004. `http://tinyurl.com/6bkbwbm`.

[17] C. Lauradoux, J. Ponge, and A. Roeck. Online Entropy Estimation for Non-Binary Sources and Applications on iPhone. Rapport de recherche RR-7663, INRIA, June 2011.

[18] A. Rukhin and al. A statistical test suite for random and pseudorandom number generators for cryptographic applications, 2001.

[19] J. Voris, N. Saxena, and T. Halevi. Accelerometers and randomness: perfect together. In *Fourth ACM Conference on Wireless Network Security - WISEC 2011*, pages 115–126, Hamburg, Germany, June 2011.

[20] L. Westlund. Random Number Generation Using the MSP430. Technical Report SLAA338, Texas Instruments, 2006.