Harvesting Entropy from On-board Sensors of Constrained Devices for Hardening Security of IoT Communication Mechanisms

Marcin P. Pawlowski[1]*, Antonio J. Jara[1], and Maciej J. Ogorzalek[2]
[1]Institute of Information Systems, University of Applied Sciences,
Western Switzerland (HES-SO), Sierre, Switzerland
marcin.pawlowski@uj.edu.pl, jara@ieee.org
[2]Department of Information Technologies, Faculty of Physics,
Astronomy and Applied Computer Science, Jagiellonian University, Krakow, Poland
maciej.ogorzalek@uj.edu.pl

Abstract

One of the basic fundamentals for the current cryptosystems is high entropy random number generator. The cryptosystems that been based on the weak random number generators will not provide adequate level of security. The high entropy random number generators for the Internet of Things are very hard due to the fact of the inherited limitations of the highly constrained devices. For the purpose of the random number generation in the constrained environments, four entropy harvesting methods have been proposed and evaluated. As the potential source of the entropy the on-board integrated sensors (i.e. temperature, humidity and two light sensors) have been analysed. Additionally the costs (i.e. time and memory consumption) of the presented approach have been evaluated to assess the requirements for the Internet of Things applications. The results obtained from the first method using temperature and both light sensors have been characterized with high Shannon entropy of around 7.5. The fastest entropy harvesting methods were based on the light sensors. Presented results indicate that it is feasible to design high entropy true random number generator that have been based on the integrated on-board sensor that will be fast and require very few memory resources from the constrained device.

Keywords: Entropy, Security, Internet of Things, Sensors

1 Introduction

In the near future many people will benefit from the emerging technologies such as the Internet of Things (IoT). The IoT will consist of billions of highly constrained devices with limited computing capabilities and interconnected by the wireless communication mechanisms with the outer world. The web of newly connected machines will positively change our life standards but at the same time new security and privacy concerns will be introduced.

There have been many efforts from the research community addressing the security issues of the IoT networks. The most notable efforts came from the Internet Engineering Task Force *Datagram Transport Layer Security for the Internet of Things (DTLS-IoT)* working group that have been addressing the issues of usage of the Transport Layer Security protocol for protection of the end-to-end communication between the constrained devices in the IoT networks.[2] The recently created *Authentication and Authorization for Constrained Environments (ACE)* IETF working group have been devoted to address the

Research Briefs on Informaiton & Communication Technology Evolution (ReBICTE), Vol. 1, Article No. 3 (January 15, 2015)

*Corresponding author: HES-SO Valais Wallis TechnoArk 33960 Sierre Switzerland, phone: +48-696228393 need of highly secure and privacy oriented standards for the authorization and authentication in Internet of Things.[1]

The pivotal for the security of the network communication have been the entropy of the random number generators. The randomness have been widely used in many places of the communication stack and operating system especially responsible for the security. Having unpredictable source of the randomness might be the last line of defence against insider adversary trying to predict the outcome of random sequence to revile the biased seed that have been used and decipher the communication security. The threat of reviling sensitive information from system with biased random number generator could also be mitigated by employing high entropy random number generator. It is essential for the operation of most of the security mechanisms to relay on good source of randomness.[5] Example of good source of randomness have been the cryptographically secured pseudo random number generators (PRNG).[6] Such generators have been commonly used in many of the current security mechanisms (e.g. TLS, DTLS) and in particularly during the key negotiation phase of the authentication mechanism (e.g. EAP-TLS, EAP-PSK or EAP-MD5 [10]). Unfortunately the PRNG are required to be initiated with seed that consist of truly-random bytes to generate secure random sequences. The key to the security of the PRNG lies in the unpredictability of the seed. In highly constrained environments the problematic task have been the collection of truly-random data that could be used as the seed for the PRNG.

Main motivation behind this paper have been to evaluate entropy harvesting methods and analyse potential sources of entropy in the constrained hardware. Finding the secure, fast and lightweight ways to generate the high entropy data for the needs of the secure communication and authentication mechanisms have been also the prime motivation of this research.

As the potential sources of the entropy the on-board sensors of TelosB compatible mote have been investigated in conjunction of the proposed new way of harvesting entropy. The presented approaches have been tested and the entropy of its output have been measured with *Shannon* and *min-entropy* estimators. Additionally the time and memory requirements for ContikiOS operating system of the proposed approaches have been measured.

The remainder of this paper have been structured as follows, Section 2 have been devoted for the analysis of the related works, Section 3 introduces basic definitions used throughout this publication, Section 4 describes the platform and software used for the research, Section 5 have been devoted for the description of the methods designed and evaluated for the publication, Section 6 describes the experiments that have been prepared and executed during the research, Section 7 presents the analysis of the results, Section 8 discusses the differences between this results and the previous research efforts and Section 9 concludes the research. The paper ends with Section 10 describing future work efforts and Section 11 with acknowledgements.

2 Related work

The work of the Hennebert, Hossayni and Lauradox in [9] have been the most relevant to this work. The authors have evaluated the many on-board and additional sensors (e.i. temperature, humidity, magnetic, gas pressure, motion, vibration and acceleration sensors) as the potential source of the entropy. The evaluation of the sensors have been executed in three modes of operation: stability, dynamic and saturation. Additionally the authors as the first have used the NIST recommended *min-entropy* to measure the quality of the source of entropy. However, the authors have not measured the time required to gather particular amount of the entropy and they also have not presented the results of memory consumption of the proposed solutions.

In [7] the Francillon and Castelluccia have showed that the transmission bit errors on wireless sensor network have been very good source of randomness. Authors showed that usage of transmission bit errors as the source of entropy is feasible and have designed and implemented practical random number generator for constrained devices. They have measured the time and memory requirements of presented solution. Unfortunately the authors have not evaluated the quality of the entropy of the source of randomness they have been using.

Both related works have presented very interesting contribution for the problems of the entropy harvesting in the constrained environments. But they have been also missing the complementary information (i.e. time and memory measurements for the [9] and entropy evaluation for the [7]). The contribution of this paper will present the analysis of the new solutions with whole information suite.

3 Definitions

Following the notations from the NIST recommendations [6] and [4] the definitions presented below have been used throughout this paper.
The source of randomness have been evaluated by two entropy estimators, the most known Shannon entropy (equation 1) and NIST recommended min-entropy (equation 2).

$$H(X) = -\sum_{x} \mathscr{P}[X=x] \log_2 \mathscr{P}[X=x] \tag{1}$$

$$H_\infty(X) = -\log_2(\max_x \mathscr{P}[X=x]) \tag{2}$$

Random number generators or random bit generators could be differentiated in to two categories the *deterministic* and the *non-deterministic*.

The *Deterministic Random Bit Generator (DRBG)* produces pseudo-random sequence of bits and requires to be initialized with seed on which secrecy the security of the DRBG lies. Two identical generator initiated with the same seed will produce the same output. Such DRBG are very useful in the communication security but they will no further analysed in this research. Throughout this paper the DRBG will be addressed as the Pseudo Random Number Generators (PRNG).

Second category of the random number generators have been *Non-deterministic Random Bit Generators (NDRBG)*. NDRBG produce output with full (or at least very high) entropy. That means that the output sequence of the NDRBG is unpredictable. In this research such random number generators are pursuit. Throughout this paper the NDRBG will be addressed as True Random Number Generators (TRNG).

4 Platform description

The main platform for used during the research have been TelosB compatible mote with MSP430 MCU and 10 KB of RAM and 50 KB of Flash memory. All developed applications have been designed to run under control of ContikiOS 3.x. During the research as the source of the entropy the temperature, humidity, visible light and separate visible light with infra-red on-board sensors have been used. Technical details of the analysed sensory devices have been presented on the Table 1.

Additionally the *entropy* [8] software packet for the *R programming language and software environment for statistical computing and graphics* [3] have been used for the estimation of the entropies and to process and graphical represent the collected data.

Sensor	**Part**	**Properties**
Light(1)	Hamamatsu® S1087 Series	Visible Range (560 nm peak sensitivity wavelength)
Light(2)	Hamamatsu® S1087 Series	Visible & Infra-red Range (960 nm peak sensitivity wavelength)
Temperature	Sensirion® SHT11	Range: -40 ∼ 123.8 °C Resolution: : ± 0.01(typical) Accuracy: ± 0.4 °C (typical)
Humidity	Sensirion® SHT11	Range: 0 ∼ 100% RH Resolution: 0.05 (typical) Accuracy: ± 3 %RH (typical)

Table 1: Technical specification of the analysed on-board sensors.

5 Methods

The entropy harvesting methods proposed in the paper have been designed to determine the quality of the entropy of the bytes that have been concatenated least significant bits of the outputs of the analysed sensory devices.

Four entropy harvesting methods have been designed for the purpose of this research. All of them are based upon observation of the instability (unpredictability) of the least significant bits of the returned data from the on-board sensor devices. The instability could be related to various factors such as Analog-Digital Converter precision, electronic noise, measurement accuracy, external environment instability like small temperature or light variations that unpredictable.

The most straightforward approach exploiting that observation have been represented by the following algorithm:

```
Require: $1 \leq M \leq 4$
  $e \leftarrow 0$
  $N \leftarrow \lfloor 8 \div M \rfloor$
  for $i \leftarrow 0$ to $N-1$ do
    $s \leftarrow read\_sensor()$
    $t \leftarrow least\_significant\_bits(s, M)$
    $e \leftarrow e$ or $binary\_shift\_left(t, (i * M)$
  end for
  if $M = 3$ then
    $s \leftarrow read\_sensor()$
    $t \leftarrow least\_significant\_bits(s, 2)$
    $e \leftarrow e$ or $binary\_shift\_left(t, 6)$
  end if
  return $e$
```

Where M is the number of the method, $read\_sensor()$ returns the output data from the particular sensor, $least\_significant\_bits(a, b)$ returns b least significant bits from the input a and $binary\_shift\_left(a, b)$ returns value which is equal to value a left shifted by b bits.

First method ($M = 1$) gets only one least significant bit from every data returned form the sensor device and concatenates the gathered bits until they form an output byte - the sensor needs to be read

Sensor	Method 1	Method 2	Method 3	Method 4
Temperature	1698 ms	850 ms	636 ms	424 ms
Humidity	512 ms	253 ms	188 ms	127 ms
Light(1)	1.19 ms	0.85 ms	0.70 ms	0.67 ms
Light(2)	1.22 ms	0.85 ms	0.76 ms	0.70 ms

Table 2: Time required to collect 1 byte of the entropy from analysed sensors with particular harvesting methods.

eight times to build one byte of random data. *Second method* ($M = 2$) takes two least significant bits from every value returned from the sensor device - it requires only to read four times the sensory data to build one byte of the entropy. *Third method* ($M = 3$) reads twice the sensor data and takes the three least significant bits from every sensor value returned, then reads the sensor last time and takes two least significant bits - it is required to access the sensor three times to build one byte of the entropy. *Fourth method* ($M = 4$) takes four least significant bits from every sensor data request and build the output random byte - it is required to read the sensor data only twice to build one random byte.

6 Experiments

The experiments have been designed to determine what conjunction of the proposed method and sensory devices give best randomness. This have been achieved by evaluating the Shannon entropy and min-entropy of the gathered random data. Additionally alongside the RAM and Flash memory usage the time required to collect one byte of the entropy have been measured.

All of the experiments have been performed in stable conditions, home-office environment without introducing any extreme factors during the measurements. The simulation environment have been devised in such manner to reflect the conditions of the large quantities of the IoT devices deployed in the natural home-office environments.

7 Analysis

In this section the experimental results have been presented and analysed. At first, the time related results have been discussed then the memory consumption results have been presented and at the end of the section the entropy measurement results have been analysed.

7.1 Times

First experiments have been designed to determine the time required to collect one byte of the entropy from analysed sensors. It has been noted that the longest entropy collecting time has been measured for temperature sensor. The temperature sensor based harvesting method one takes almost 1.7 second to collect one byte of the entropy. The second longest entropy gathering time has been observed for the humidity sensor and it has been around 0.5 second. Both sensors have been based on the same on-board electronics *Sensirion® SHT11* which have been the main factor of the delay. The entropy collection time for temperature and humidity sensors have been reduced around four times for the method fourth. Clearly the main factor contributing to the time reduction have been of course the fact of the cut of the number of reads of sensor device.

Sensor	RAM	Flash			
		Method 1	Method 2	Method 3	Method 4
Temperature	1	966	988	998	976
Humidity	1	966	988	998	976
Light(1)	4	488	510	520	498
Light(2)	4	488	510	520	498

Table 3: Flash and RAM memory byte requirements for gathering entropy with particular harvesting methods.

The fastest entropy collecting times have been recorded for the light sensors and it has been around 1.2 ms when first harvesting method has been used. Using fourth entropy collecting method halved the random byte gathering time. Both light sensor devices have been based on the same electronics *Hamamatsu® S1087 Series*.

Full processing time comparison for different entropy harvesting methods has been presented on the table 2.

7.2 Memory

Second experiment has been designed to measure the memory consumption of entropy gathering function in comparison to the sensory device and method in use. Similarly to the time experimental results the differentiation between *Sensirion® SHT11* and *Hamamatsu® S1087 Series* based sensory devices has been observed. The entropy harvesting methods using SHT11 based sensory devices (which have been the humidity and temperature sensors) have flash memory requirements of around 1 kilobyte and RAM requirements of just 1 byte. The entropy gathering methods that have been using the S1087 Series based devices (light sensors) have 50% lower flash memory requirements of around 500 bytes but have higher RAM usage of 4 bytes.

Full comparison of Flash and RAM memory consumption of different entropy harvesting methods has been presented on the table 3.

7.3 Entropy

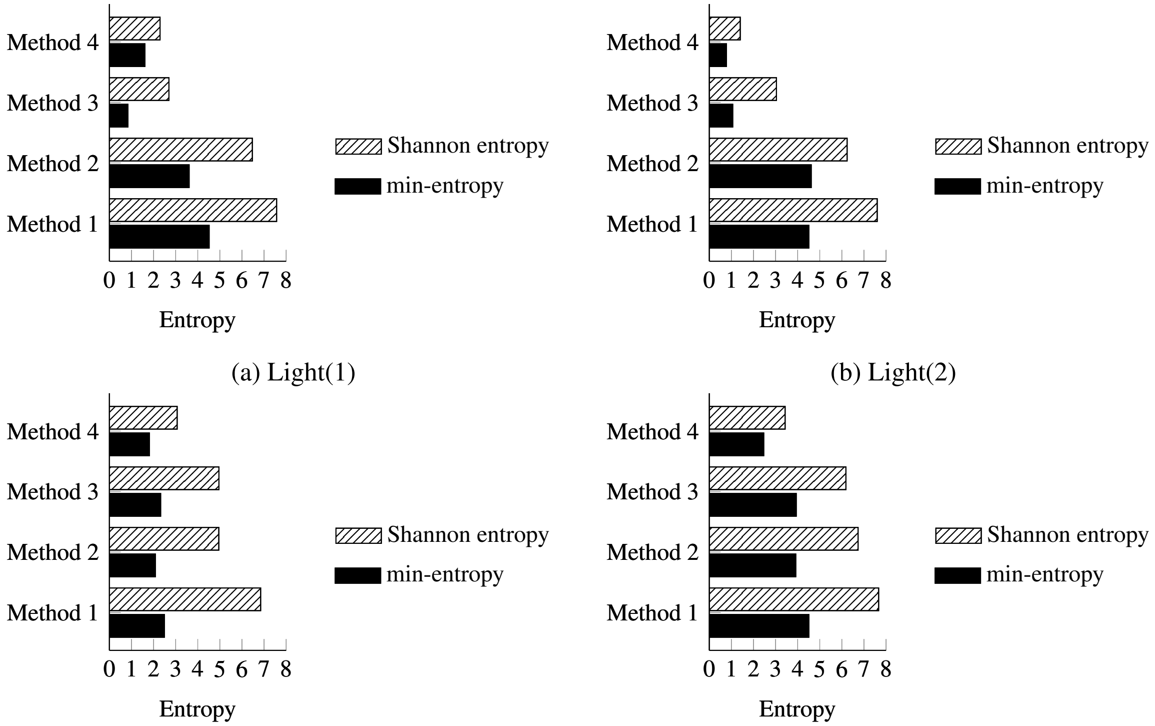
Last experiment have been devoted to estimate the values of Shannon entropy and min-entropy of the gathered random bytes.

7.3.1 By methods

The highest Shannon-entropy has been observed for the method one. The min-entropy has also the highest values for the first method, with the exception of the second method for light(2) sensor (this might be related to the insufficient number of the samples). The differences between the values of estimated entropies between other methods have been less significant. Interesting observation has been made while analysing entropy of the data collected from the light sensors, there has been very prominent differentiation between first two methods and last two methods for both min-entropy and Shannon entropy.

7.3.2 By sensors

The highest Shannon-entropy has been observed for the temperature sensor. The min-entropy has been high for the temperature sensor but also for both light sensors, with the exception of the third and fourth



(a) Light(1)

(b) Light(2)

(c) Humidity

(d) Temperature

Figure 1: Shannon entropy and min-entropy estimation of random bytes from analysed sensors by particular entropy harvesting methods.

methods. The interesting observation has been made for the results of random data from humidity sensor where the differences between the Shannon entropy and the min-entropy have been most significant.

Full Shannon entropy and min-entropy comparison for different entropy harvesting methods has been presented on the figure 1.

8 Discussion

The min-entropy results that have been obtained during this research have been mostly in parallel to the results presented by the Hennebert, Hossayni and Lauradox in [9]. Measured min-entropy of the temperature sensors in the [9] has been estimated to 3.55 for the internal sensor and 4.8 for the external sensor. The on-board temperature sensor that has been analysed during this research has achieved 4.5 of min-entropy for the first method. That has been much higher entropy value in comparison to the internal temperature sensor results from [9] but also has been a bit lower result in comparison to the external sensor.

The highest Shannon entropy measured in the [9] has been for the vibration sensor and has been estimated for 7.8. The highest measured Shannon entropy of all of the on-board sensor devices analysed during this research has been observed for temperature sensor and it has been 7.66.

The differences in the values of the entropy between best performing methods have not been very substantial. Because all of the estimated entropy results have not been presented in the [9] the impact of the methods presented in this paper on the entropy results has not been conclusive. Additional metrics that could be compared with the results of this research have not been presented in the [9].

Due to the fact that in the [7] the authors have been using very different development environment (i.e. operating system and constrained devices processors) the comparisons presented below has been more symbolic than conclusive.

The memory requirement that have been presented in this research have been significantly lower then the results presented by the Francillon and Castelluccia in [7].

The ROM memory requirements for the solution presented in [7] have been measured as 11086 bytes for MICA2 platform and 9832 bytes for MICAz platform. The highest ROM memory requirements for the solutions analysed during this research have been only 998 bytes for the temperature and humidity based solution using the most memory consuming method (i.e. third method). The lowest ROM memory requirement have been measured for the first method with light sensor and has been 488 bytes. The solutions presented in this research achieved ROM memory savings of around 90% for the humidity/temperature based methods and of around 95% for the light based methods. The ROM memory usage that has been very important factor for the very constrained devices and its applications.

The RAM memory requirements presented in the [7] have been measured as 471 bytes for the MICA2 platform and 416 bytes for the MICAz platform. The RAM requirements for the solutions presented in this research have been evaluated for 1 byte for the temperature/humidity sensor based methods and 4 bytes for the light sensor based methods. It should be mentioned that the solutions presented in this research have been designed to generate only one byte of the random data in contrast to the solution presented in [7] where it generates 8 bytes in one execution.

The processing times presented in the [7] have been estimated as around 146.2 ms (with EEPROM operations) or 4.5 ms (without EEPROM operations) - calculated as a sum of random number generation, initialization and entropy accumulation for getting 1 byte of the entropy. On the one hand, the results presented by the authors of the [7] have been significantly smaller then the time results for the temperature/humidity based methods (1.6 second for the method one). On the other hand, the processing time results for the light sensor based methods have been significantly shorter (1.22 ms for the method one).

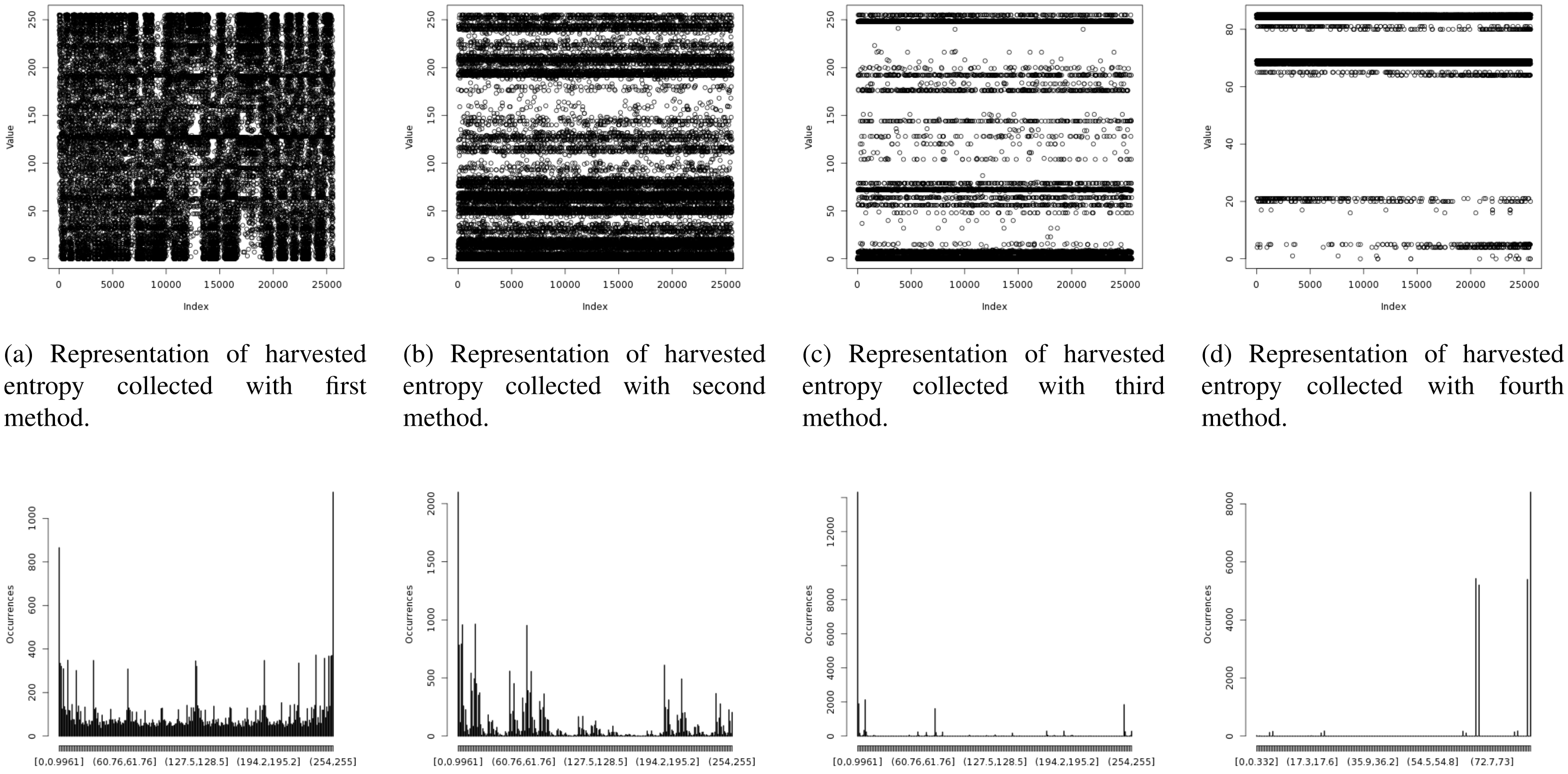
The authors of [7] have not measured the quality of the entropy of its solution.

9 Conclusion

Received results have been consistent with the obvious assumption that the most entropy could be harvested using method that concatenates only the least significant bit of the data read from the sensor device. The least significant bits of the sensory measurement are prone to various kinds of disorders which outcome has been hard to predict. The precision of the measurement, unpredictability of the external environment and internal electronic noise have been the factors that compose to highest unpredictability of the least significant bit of sensor devices.

The best source of the entropy has came from the conjunction of the first method and temperature sensor. The time required to execute this solution have been estimated as 1.7 second. Such delay to receive only one bit of the entropy has been unacceptable especially for the usage in the communication stack. The same timing issues have been observed for the humidity sensor based methods with lower but still unacceptable 0.5 second per one byte of entropy.

The most efficient source of the entropy with high values of min-entropy and Shannon entropy and exceptionally quick entropy harvesting times of around 1.2 ms per byte of entropy have been the light sensors based methods. In addition to the excellent randomness source the light sensors based method have 50% less flash memory requirements than temperature/humidity sensory devices. That makes the light sensors in conjunction with the method one the most efficient solution for entropy harvesting for the constrained devices.



(a) Representation of harvested entropy collected with first method.

(b) Representation of harvested entropy collected with second method.

(c) Representation of harvested entropy collected with third method.

(d) Representation of harvested entropy collected with fourth method.

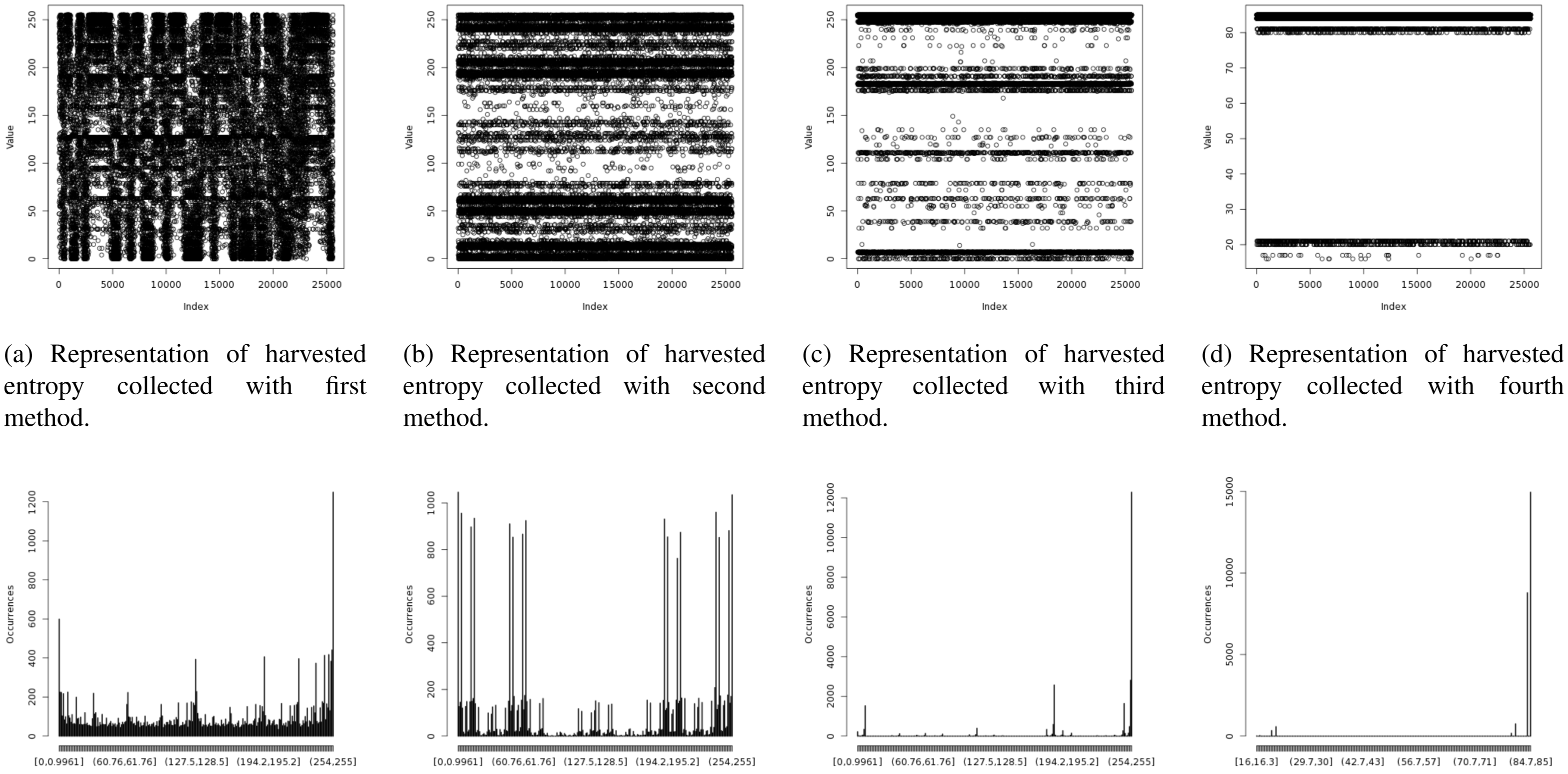
(e) Occurrence distribution of harvested entropy collected with first method.

(f) Occurrence distribution of harvested entropy collected with second method.

(g) Occurrence distribution of harvested entropy collected with third method.

(h) Occurrence distribution of harvested entropy collected with fourth method.

Figure 2: Entropy harvested from light(1) sensor with particular harvesting methods.



(a) Representation of harvested entropy collected with first method.

(b) Representation of harvested entropy collected with second method.

(c) Representation of harvested entropy collected with third method.

(d) Representation of harvested entropy collected with fourth method.

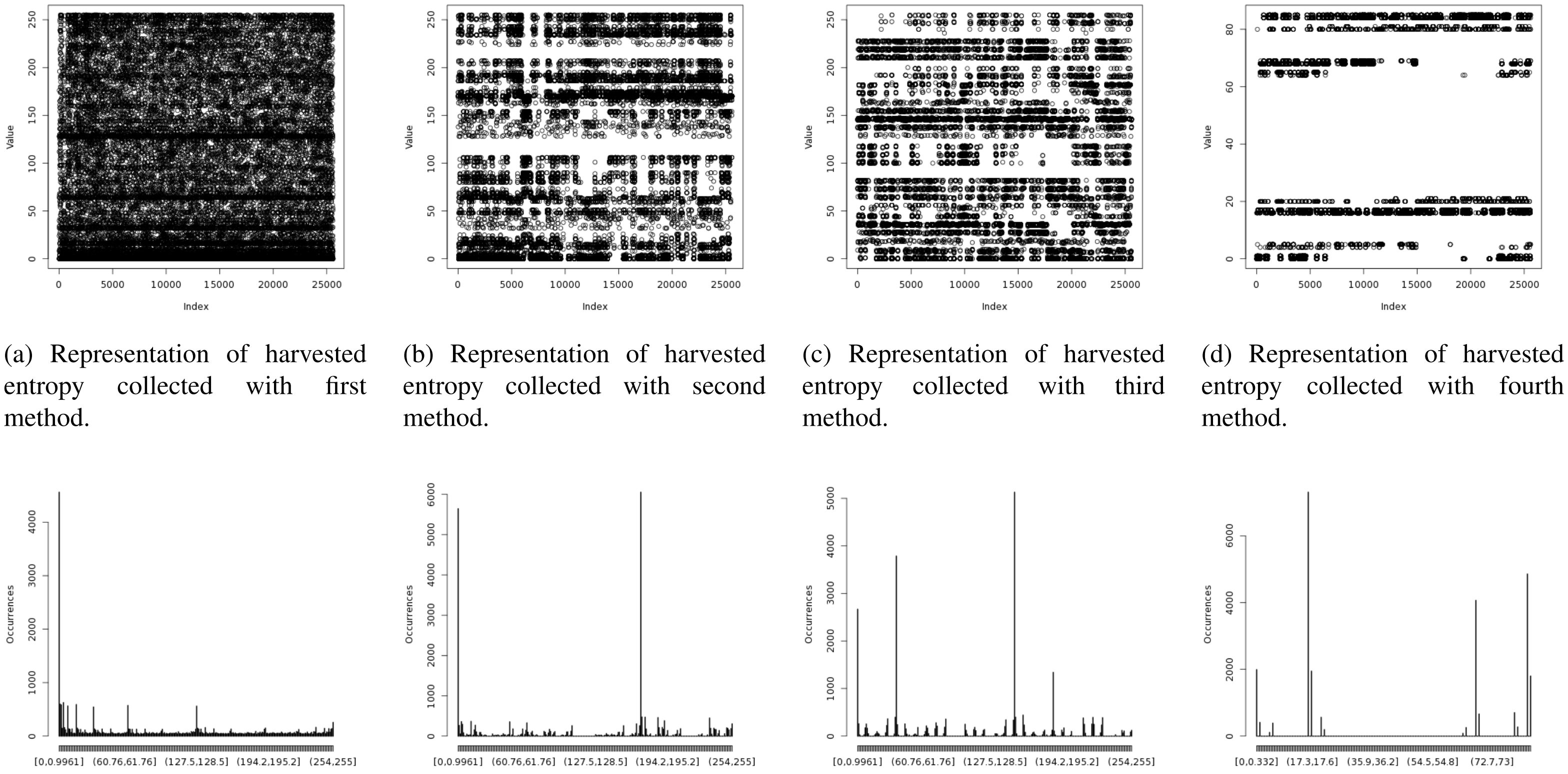
(e) Occurrence distribution of harvested entropy collected with first method.

(f) Occurrence distribution of harvested entropy collected with second method.

(g) Occurrence distribution of harvested entropy collected with third method.

(h) Occurrence distribution of harvested entropy collected with fourth method.

Figure 3: Entropy harvested from light(2) sensor with particular harvesting methods.



(a) Representation of harvested entropy collected with first method.

(b) Representation of harvested entropy collected with second method.

(c) Representation of harvested entropy collected with third method.

(d) Representation of harvested entropy collected with fourth method.

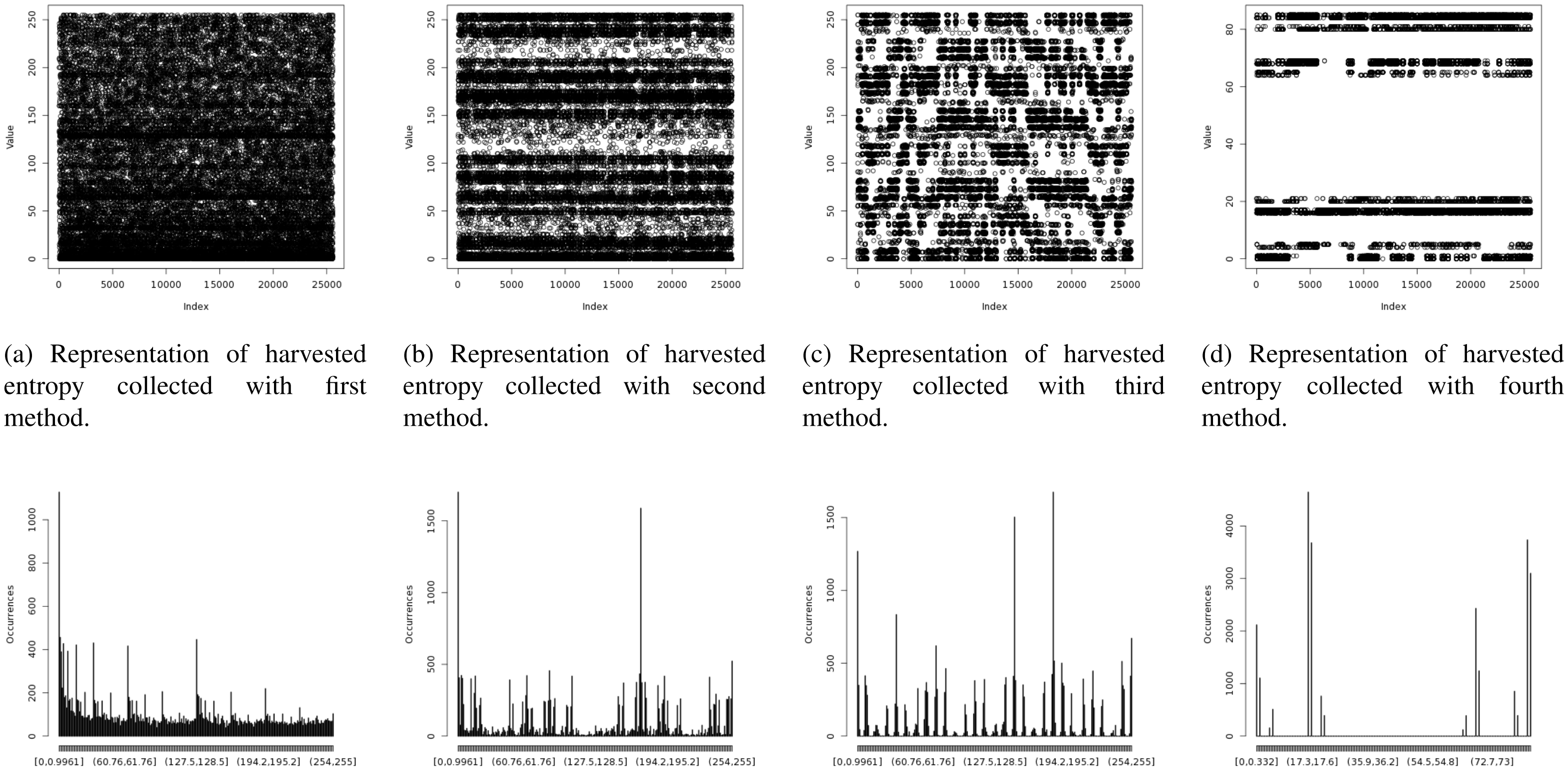
(e) Occurrence distribution of harvested entropy collected with first method.

(f) Occurrence distribution of harvested entropy collected with second method.

(g) Occurrence distribution of harvested entropy collected with third method.

(h) Occurrence distribution of harvested entropy collected with fourth method.

Figure 4: Entropy harvested from humidity sensor with particular harvesting methods.



(a) Representation of harvested entropy collected with first method.

(b) Representation of harvested entropy collected with second method.

(c) Representation of harvested entropy collected with third method.

(d) Representation of harvested entropy collected with fourth method.

(e) Occurrence distribution of harvested entropy collected with first method.

(f) Occurrence distribution of harvested entropy collected with second method.

(g) Occurrence distribution of harvested entropy collected with third method.

(h) Occurrence distribution of harvested entropy collected with fourth method.

Figure 5: Entropy harvested from temperature sensor with particular harvesting methods.

10 Future Work

The future development of the potential sources of the entropy for the constrained devices will be devoted to analyse different types of the sensory devices in the conjunction of different IoT capable devices. One of the most important sensors that will be analysed will be *Sensirion® SHT21* which is part of the Bluetooth Smart One development board. The correlation of the internal electronic noise influences and dependences will also be pursuit.

Identical methodology will be applied for the analysis of the entropy gathered from an analogue microphone. Such device is also an on-board device of the Bluetooth Smart One development board.

Additionally the currently obtained results will be extended to find statistical biased values and correct the distribution to have better Shannon entropy and min-entropy estimations.

11 Acknowledgments

The authors would like to thank to the HES-SO and the Institute of Information Systems funding and support. This project has been supported by the Swiss national government through the Sciex-NMSch (Scientific Exchange Programme between Switzerland and the New Member States of the EU) with the project code 13.121, named BASTION ”Bootstrapping, Authentication, Security and Trust for the Internet of Things Networks”.

References

[1] IETF Authentication and Authorization for Constrained Environments (ACE) Working Group. `https://datatracker.ietf.org/wg/ace/charter/`.

[2] IETF Datagram Transport Layer Security for the Internet of Things (DTLS-IoT) Working Group. `https://datatracker.ietf.org/wg/dice/charter/`.

[3] The R Project for Statistical Computing. http://www.r-project.org/.

[4] Barker, Elaine and Kelsey, John. Recommendation for the Entropy Sources Used for Random Bit Generation. *Draft NIST Special Publication*, 2012.

[5] D. Eastlake, J. Schiller, and S. Crocker. Randomness Requirements for Security. IETF RFC 4086, June 2005. http://tools.ietf.org/html/rfc4086.

[6] Elaine B Barker and John Michael Kelsey. *Recommendation for Random Bit Generator (RBG) Constructions*. US Department of Commerce, National Institute of Standards and Technology, 2012.

[7] A. Francillon and C. Castelluccia. Tinyrng: A cryptographic random number generator for wireless sensors network nodes. In *Proc. of the 5th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops (WiOpt'07), Limassol, Ciprus*, pages 1–7. IEEE, April 2007.

[8] Hausser, J., and K. Strimmer. R entropy package. http://strimmerlab.org/software/entropy/.

[9] C. Hennebert, H. Hossayni, and C. Lauradoux. Entropy harvesting from physical sensors. In *Proc. of the 6th ACM conference on Security and Privacy in Wireless and Mobile Networks (WiSec'13), Budapest, Hungary*, pages 149–154. ACM Press, April 2013.

[10] Marcin Piotr Pawlowski, Antonio J. Jara and Maciej J. Ogorzalek. Extending Extensible Authentication Protocol over IEEE 802.15.4 networks. In *Proc of the 8th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS'14), Birmingham, UK*, pages 340–345. IEEE, July 2014.

Author Biography

Marcin Piotr Pawlowski received M. Sc. degree in computer science from Jagiellonian University in Krakow, Poland, and B. Sc. degree in electronics and telecommunication from AGH University of Science and Technology in Krakow, Poland, in 2010 and 2013, respectively. He is pursuing Ph. D. degree in computer science at Jagiellonian University in Krakow, Poland. Recently He have received SCIEX Fellowship for realization of the project BASTION - Analysis, design and evaluation of Bootstrapping, Authentication, Security and Trust for the Internet of Things Networks, and is working at the Institute of Information System of the University of Applied Sciences-Western Switzerland. He is interested in the network protocols, and security mechanisms, and in particularly for wireless applications such as Internet of Things.

Antonio J. Jara Assistant Prof. PostDoc at University of Applied Sciences Western Switzerland (HES-SO) from Switzerland , vice-chair of the IEEE Communications Society Internet of Things Technical Committee, and founder of the Wearable Computing and Personal Area Networks company HOP Ubiquitous S.L., He did his PhD (Cum Laude) at the Intelligent Systems and Telematics Research Group of the University of Murcia (UMU) from Spain. He received two M.S. (Hons. - valedictorian) degrees. Since 2007, he has been working on several projects related to IPv6, WSNs. and RFID applications in building automation and healthcare. He is especially focused on the design and development of new protocols for security and mobility for Future Internet of things, which was the topic of his Ph.D. Nowadays, he continues working on IPv6 technologies for the Internet of Things in projects such as IoT6, and also Big Data and Knowledge Engineering for Smart Cities and eHealth. He has also carried out a Master in Business Administration (MBA). He has published over 100 international papers, As well, he holds one patent. Finally, he participates in several Projects about the IPv6, Internet of Things, Smart Cities, and mobile healthcare.

Maciej J. Ogorzalek Professor and Head of the Department of Information Technologies, Jagiellonian University Krakow, Poland. He held visiting positions in Denmark, Switzerland, Germany, Spain, US, Japan, Hong Kong. In 2000 he worked at the National Microelectronic Center, Seville, Spain. In 2001 he was visiting professor at Kyoto University, in 2005 Hertie Foundation guest professor at The Goethe University Frankfurt-am-Main. 2006-2009 he held the Chair of Bio-signals and Systems, Hong Kong Polytechnic University under the Distinguished Scholars Scheme. Author of over 280 technical papers published in journals and conference proceedings, and the book Chaos and Complexity in Nonlinear Electronic Circuits (World Scientific, 1997). Receipient of many awards including IEEE-CAS Golden Jubilee Award and the CASS Guillemin-Cauer Award, 2002, CAS Distinguished Service Award 2012, Education Medal (Poland). He served as Editor-in-Chief of IEEE Circuits and Systems Magazine (2004-2007), Associate Editor of Proceedings of the IEEE (20014-2009), IEEE Transactions on Circuits and Systems (several times), Journal of the Franklin Institute (1997-present), International Journal of Bifurcation and Chaos (2004-present), International Journal of Circuit Theory and Applications (2000-present). He was the Vice-President of the Sniadecki Science Foundation (until 2005). He served IEEE CAS Society in various capacities including 2008 President. In 2012 elected Member of Academia Europaea. In 2014 became IEEE Division 1 Director-elect.