Accelerometer Based Random Number Generation on RFID Tags

Nitesh Saxena
Polytechnic Institute of New York University
Six MetroTech Center, Brooklyn, NY 11201
nsaxena@poly.edu

Jonathan Voris
Polytechnic Institute of New York University
Six MetroTech Center, Brooklyn, NY 11201
jvoris@isis.poly.edu

1. INTRODUCTION

There has been much work on the development of security and privacy mechanisms that attempt to take the limited capabilities of RFID tags into account. Most of these schemes rely on the presence of random number generation. Generating random numbers of sufficient quality for cryptographic applications is not a trivial task, even for traditional computers. Consequently, mechanisms must be designed for introducing randomness from an external source. While modern general purpose computers have several techniques available for the generation of high quality random numbers, this requirement is beyond the capacity of today's average RFID tag.

In this abstract, the viability of using onboard sensors to collect ambient noise of different forms for use as a source of randomness is considered. Specifically, accelerometers are focused on as an example of a typical low cost sensor. This work aims to analyze ways in which entropy sources such as these can be utilized to efficiently produce the amount of randomness necessary to support various cryptographic protocols aimed at low cost tags. Results from cryptographic literature will be applied to produce a random number generation procedure with provable security guarantees that were lacking in previous work.

2. RELATED WORK

Since many of the random number generation techniques available to traditional computers are out of the reach of RFID tags, alternative approaches to the creation of random values must be considered. A recent example of such a proposal is Fingerprint Extraction and Random Number Generation in SRAM (FERNS) [4, 5]. This technique utilizes onboard RAM as the source of true randomness. This technique is quite promising as any device, regardless of its constraints, will contain some amount of onboard memory from which randomness can be drawn.

Unfortunately, previous work has illustrated that practical considerations prevent the FERNS approach to random number generation from reaching its full theoretical potential [6]. Since FERNS relies on preexisting memory circuitry as a source of entropy, it must compete with other system functionalities for use of this shared resource. Furthermore, RAM is subject to a phenomenon known as data remanence. Such memory retains its contents while receiving power and for a duration of several seconds afterwards. Thus, there is a time period after losing power during which stored data remains intact in memory. After a portion of memory has been used for entropy collection once, it will require a relatively extended period of time without power before it can again be used in this capacity. This may lead to unacceptably high delays.

Since random number generation based on existing general-purpose memory circuitry appears insufficient for the needs of RFID authentication protocols, entropy collection techniques which rely on onboard sensors are turned to instead. While not as general purpose as RAM, sensors have many uses outside of security and privacy applications. As such, they may already be present on an RFID device. While the naturally occurring phenomena these sensors capture are unpredictable, they necessarily contain some bias rather than being distributed uniformly. From the perspective of a cryptographic application expecting high quality randomness, this bias is unacceptable because it could potentially be exploited by an adversary to extract information about the cryptosystem's internal state.

Extraction functions have been created to bridge the gap between the expectations of cryptographic designers and the realities of the availability of random numbers. An extractor is a function that takes a string of unpredictable but biased, or "weakly" random, bits as input and returns a string of close to uniform, or "strongly" random, bits as output. One example of such an extractor is the "independent sources" extraction of [1], which simply works by multiplying two independent values and adding the result to a third in a recursive fashion. Along the same lines, a second type of extractor was described in [2]. This extraction technique utilizes a Toeplitz matrix as a seed, which is multiplied against the column matrix containing the input to the hash function.

3. PROPOSAL AND EXPERIMENTS

To investigate the viability of generating cryptographic quality random numbers on RFID tags, several experiments were performed. First, it was necessary to approximate the min-entropy of the accelerometer samples that were intended for extraction. To do so, it was first established under what circumstances the min-entropy of the accelerometer samples was minimized. Accelerometer samples were taken over a 10 minute interval while a variety of different movements were performed with the tags. Out of all these patterns, the stationary option yielded the lowest min-entropy with a value of 3.43. Thus it was concluded that this is the min-entropy level that should be assumed for accelerometer outputs, since it is unknown how much motion, and therefore how much additional min-entropy, would be captured by the samples at any given time.

Having established that the min-entropy of the RFID tag's accelerometer samples was at its lowest when the tag was still, next a sizable sample was needed to ensure an accurate estimate of the sensor value's min-entropy. To achieve this, a tag was programmed to transmit its raw accelerometer values upon receipt of a query from a reader. The reader was left to query the tag overnight. The 1,231,095 samples collected yielded a min-entropy of 3.46, confirming that the original min-entropy estimate was not due to a fluke in the smaller sample.

Next, the extractor from [1] was implemented on a WISP tag. In order to determine to what extent this "independent sources" extractor increased the quality of randomness beyond what was initially present in the raw accelerometer samples, a tag programmed to perform this extraction on its accelerometer values and return the result was also left to be queried by a reader overnight. The min-entropy of the 1,237,066 result samples was found to be 3.90. While this is quite far from the desired level of uniformity, it is still a 12.71% increase in min-entropy over the raw samples. More beneficially, however, this hashing process reduces the sample size from 30 to 10 bits. Since more entropy is being derived from fewer bits, this is a promising result for using the hashed accelerometer samples for further processing.

To take advantage of this increased entropy per bit, the extractor from [2] was implemented on a laptop computer. The "independent sources" extracted samples were then fed into this extractor. Unlike the previous results, which were still far from uniform, there were no repeated values among the 24,741 resultant twice-hashed values. This indicated that the goal of generating uniformly distributed random values from accelerometer sensor values had been achieved.

The Toeplitz matrix extraction technique is in the process of being implemented and evaluated on WISP tags, along with several other potential randomness extraction techniques. Several changes had to be made in order to optimize the extractor to work within the constraints of a low cost RFID tag. Only the top row and leftmost column of the Toeplitz matrix seed are permanently stored on the tag. When performing matrix multiplication operations, each row of the matrix is instead generated as needed in order to minimize the amount of memory needed to store the seed. Furthermore, all binary values are stored in byte arrays rather than arrays of boolean values. While this adds complexity to the manipulation of individual bits, it was also done to save on storage space. As a final example, rather than buffering accelerometer samples and applying the extraction function to them once enough had been received, the matrix operations were done on a piecemeal, sample-by-sample basis, saving both memory as well as computation.

4. CONCLUSIONS

This abstract introduced the problem of random number generation in the context of RFID technology. A hitherto unexplored technique for addressing this issue, sensor based entropy collection, was proposed as well. These preliminary results indicate that, when affordable, sensors are a viable option for obtaining random values on RFID tags. This method is not just limited to RFID devices, however. It extends easily to other devices like mobile phones and traditional computing devices such as laptops. The authors of [3] mention the possibility of developing games for entropy generation based on motion that would be monitored through an accelerometer. A drawback to that approach is that it involves direct participation on the part of the device's user. In contrast, this work investigates random number generation techniques that do not require any explicit user involvement.

5. REFERENCES

[1] B. Barak, R. Impagliazzo, and A. Wigderson. Extracting Randomness Using Few Independent Sources. *SIAM Journal on Computing*, 2006.

[2] B. Barak, R. Shaltiel, and E. Tromer. True Random Number Generators Secure in a Changing Environment. In *Cryptographic Hardware and Embedded Systems*, 2003.

[3] R. Halprin and M. Naor. Games for Extracting Randomness. In *Symposium On Usable Privacy and Security*, 2009.

[4] D. Holcomb, W. Burleson, and K. Fu. Initial SRAM State as a Fingerprint and Source of True Random Numbers for RFID Tags. In *Conference on RFID Security*, 2007.

[5] D. E. Holcomb, W. P. Burleson, and K. Fu. Power-up SRAM State as an Identifying Fingerprint and Source of True Random Numbers. *IEEE Transactions on Computers*, 2009. to appear.

[6] N. Saxena and J. Voris. We Can Remember It for You Wholesale: Implications of Data Remanence on the Use of RAM for True Random Number Generation on RFID Tags. In *Conference on RFID Security*, 2009.