Seasoning Effect Based Side Channel Attacks to AES Implementation with Phase Change Memory

Lei Xu
University of Houston
Houston, Texas 77204
xuleimath@gmail.com

Weidong Shi
University of Houston
Houston, Texas 77204
larryshi@cs.uh.edu

Nicholas Desalvo
University of Houston
Houston, Texas 77204
nick14822@sbcglobal.net

ABSTRACT

Side channel attacks have received much attention as of recent. Compared with cryptanalysis that focus on the algorithms themselves, side channel attacks are usually much more practical and realistic. Side channel attack is closely related to hardware and implementations. Many cryptographic algorithms that are secure by themselves become insecure when they are poorly implemented or are running on devices with special properties. In this work, we examine the probability of side channel attacks on phase change memory which is envisioned as a candidate of universal memory to replace DRAM or SRAM. The attacks exploit the seasoning effect of phase change memory, a phenomenon of PCM cell behavior change as a function of operative cycles. We conducted detailed experiments using PCM modeling tools that accurately simulate the seasoning effect of PCM cells. The results show that for pipelined AES cipher that is not properly implemented when using PCM as the storage device, side channel attacks may lead to serious security risks. We also provide suggestions for secure implementation of AES to prevent such attacks.

Categories and Subject Descriptors

B.7.1 [**INTEGRATED CIRCUITS**]: Types and Design Styles—*Memory technologies*; E.3 [**Data Encryption**]: Standards

Keywords

PCM, side channel attack, AES

1. INTRODUCTION

Researchers have and continue to extensively design and analyze cryptography schemes under the black box assumption. This assumption states that attackers can access only the inputs and outputs of cryptography schemes. For further clarification, the attackers have no access to any data or knowledge related to intermediate results. Contrary to this assumption, real world attackers usually are able to acquire more knowledge than only the inputs and outputs. For instance attackers may acquire information related to the key used in the scheme by examining energy consumption [7,8], radiation [5], time [15], or even sound information [6]. These types of attacks are formally referred to as side channel attacks. During these types of attacks, a cryptography scheme is more akin to a gray box than a black box, i.e. in addition to learning the inputs/outputs of a cryptography scheme, attackers may also learn something about the internal states of the scheme. Besides simple measurement, many sophisticated analyses methods are also developed, e.g. higher order analysis [13], differential analysis [18], and correlation analysis [3]. Compared to the traditional theoretical analyses, side channel attacks continue to become increasingly more threatening as the attackers continue to learn about the cryptography schemes employed. Moreover, many experiments have shown that side channel information can be derived relatively easily.

Due to the nature of side channel attacks, the effectiveness of such attacks depends heavily upon the hardware and software implementations. New types of devices with different physical characteristics may lead to new side channel attacks. The vulnerability may exist within the new type of random access memory: Phase change memory (PCM) [14]. Compared to the traditional memories such as DDR, the seasoning effect of phase change memory is potentially detrimental to security, i.e., when a bit saved in a PCM cell is flipped repeatedly, the physical properties of the cell will change and lead to alternations of that cell behaviors. The seasoning effect may be exploited for side-channel information leakage. Presently, most research work related to PCM focuses on problems such as material and life extending. Yet very little attention has been spent on PCM characteristics that may facilitate side channel attacks.

We propose a side channel attack against AES implementation using PCM as its memory. The attack can recover all the round keys. Compared with other side-channel exploits against AES, the attack has the following advantages.

- The attack is simple, yet effective. The attacker only needs to force the encryption algorithm to run multiple times and measure the result seasoning effects of the PCM cells;
- The attack can be conducted off-line. Unlike other side channel attacks that measure real time information such as energy consumption, or radiation, or try to inject faults when the program is running, we only need to measure the seasoning effect of the PCM cells

HASP '14, June 15 2014, Minneapolis, MN, USA

http://dx.doi.org/10.1145/2611765.2611770.

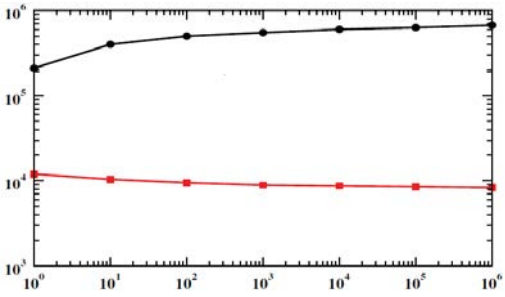

Figure 1: Seasoning effects both on SET and RESET states of a Phase Change Memory with respect to number of cycles performed. Plotted figures are based measured results of [17].

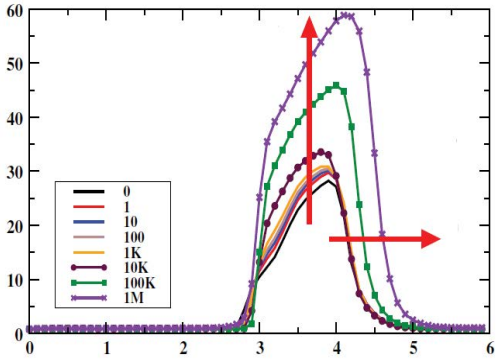

Figure 2: Equivalent $R-I$ characteristic of the PCM array with respect to program cycle numbers. The result figure is based on [20].

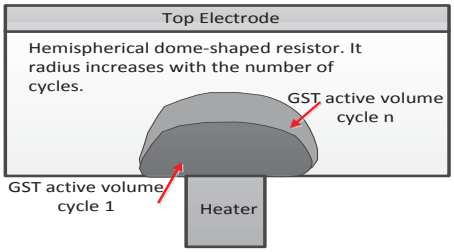

Figure 3: Physical model of expanding active volume with operational cycles of PCM cell. Reset increases with the radius of an expanding hemispherical resistor of GST material [17].

after completion of the encryption process. The attack can be repeated to eliminate measurement errors.

The rest of the paper is organized as follows: In Section 2 we shortly review the technique of PCM and the seasoning effect associated with programming cycles. Then we describe the evaluation environment in Section 3, followed by details of the basic key recovery attack in Section 4. We provide simulation and analyses in Section 5. Section 7 discusses related side channel attack works. In Section 8, we conclude the paper.

2. PROPERTIES OF PHASE-CHANGE MEMORY

Phase change memory (PCM) exploits the large resistance contrast between the amorphous and crystalline states in so-called phase change materials [14]. Although the principle of applying phase change materials to electronic memory was first demonstrated in the 1960s [11], it was only in the past decades that advances in materials and device technology have made PCMs that rival incumbent technologies for dynamic random access memory. A survey on the technology status and features of PCM points to its potential as a possible replacement for on-chip SRAM and off-chip universal DRAM memories (e.g., [14]).

Phase change memory relies on the phase change transition from crystalline into amorphous and vice versa of an active chalcogenide material such as the Ge2Sb2Te5 (GST). The active material phase change between amorphous (a-GST) and crystalline state (x-GST) principally is achieved through Joule heating. The phase change can be controlled by a careful design of writing waveform shape and parameters. The amorphous phase is associated with the RESET state of the memory. It can be realized by a program operation (RESET) which melts and amorphizes the material. The crystalline phase (x-GST) is associated with the SET state of the memory. It can be achieved by an erase operation (SET) where a crystalline shunt grows into the amorphous dome of a memory cell until saturation. The phase transition is fully reversible. However, it has been observed that an alteration of the properties of the active material may occur under SET/RESET cycling stress [12, 17, 20]. It is named as seasoning effect in phase change memory. Under this effect, when a phase change memory array is subjected to repeated SET/RESET cycling, one can observe R_{RESET} increase and R_{SET} decrease as demonstrated in Fig. 1.

The seasoning effect can be observed as well by an increase of the minimum programming voltage and on a modification of the IâĂŞV characteristics of the PCM array as shown in Figure 2(B). The process of obtaining R_{GST} is the follows. When a read voltage V_{read} is applied, the read current I_{read} drained by the SET state can be sensed. The measured I_{read} current values are converted into resistance R_{GST} by taking into account the structure of the PCM cell and its connection on the array using an equivalent circuit model depicted in Figure 2(A).

Modeling and plausible interpretations of the seasoning effect of phase change memory can be found in the recent literature [21]. In [21], the authors present a detailed computational model for that can accurately reproduce the SET seasoning effect of a phase change memory array using computer simulation that takes into account the erase kinetic process. The model is based on erase models described in [19] and [4].

In [17] and [21], the authors present a possible interpretation for the RESET seasoning effect. According to [17], every cycle of melting of the GST with the associated compositional change may result in an increasing proportion of material in the active volume with a lower crystalline-state resistivity. Consequently, with each cycle, there will be an increasingly larger volume of GST with altered composition.

The compositionally altered GST is expected to lower the melting point, which leads to the seasoning effect. As illustrated in Fig. 3, every cycle of melting may result in an increasing radius of the active volume. The active volume can be modeled as a hemispherical dome-shaped resistor. As presented in [17], the analytical dependence of the Ohmic resistance change R_{reset} of a hemispherical dome-shaped resistor with increasing radius r can be expressed as,

$$\triangle R_{reset} = \frac{\rho}{2\pi} \times \frac{r_i - r_1}{r_i r_1} \quad (1)$$

where r1 is the initial radii of the hemispherical dome-shaped resistor, r_i is the radii of the hemispherical dome-shaped resistor at program cycle i, and ρ is the resistivity of the comprising material. For interpretation of the SET seasoning effect, according to [21], during the erase operation, a crystalline shunt inclusion on the hemispherical amorphous dome must be created in order to have a current percolation path. As the amorphous cap thickness alters, the crystalline shunt geometry consequently has to be changed. This geometric alteration hypothesis of active material volume during program cycling is congruent with the measurement results where there is an increase of the time required for creating a percolation path into the programmable region of the memory cell with the number of program cycles [21].

According to the analytical seasoning model in [21] which is based on empirical electrical characterization data extracted from actual phase change memory array (e.g., [20]), the impact of SET seasoning can be evaluated by the following equation:

$$\triangle R_{set}(i) = R_{set}(i) - R_{set}(i-1) \quad (2)$$

where i is the number of SET/RESET cycling iteration. Experimental results show that $\triangle R_{set}$ variability is negligible with the cell to cell SET seasoning variability of only few tens of ohms from the statistical mean. Furthermore, there is no observed dependency or influence of the memory array topology on the SET seasoning effect.

Using empirical measurement results, the equation can be further elaborated as,

$$\triangle R_{set}(i) = \frac{l(i)\rho_c(i)}{A_c(i)} - \frac{l(i-1)\rho_c(i-1)}{A_c(i-1)} \quad (3)$$

where ρ_c(i) and l(i) in cycling can be modeled with the following linear expressions:

$$\rho_c(i) = a * i + \rho_{c0} \quad (4)$$

$$l(i) = b * i + l_0 \quad (5)$$

where ρ_{c0} is the resistivity in the crystalline state before seasoning, and l_0 is the length of the crystalline conduction path in the amorphous dome before seasoning. Coefficient a and b are extracted from the measured results of a 512 Kb phase change memory array.

A computer simulator can be developed using the SET seasoning model. As demonstrated in [21], the simulation results can match closely with the experimental seasoning data from electrical characterization and reproduce the seasoning phenomenon. In our evaluation, we developed our own modeling software based on the SET seasoning model described in [21] and the analytical erase model presented in [4]. Both models are validated by empirical experimental results using actual phase change memory array test chip in 180nm technology [20, 21].

3. EVALUATION ENVIRONMENT

To accurately model the seasoning effect of phase change memory array, we developed a simulation environment based on the analytical models of phase change memory operations and seasoning effects described in [4, 21]. The analytical models have been validated to agree closely with the experimental results collected from actual phase change memory arrays fabricated in 180nm technology with BJT selector [4, 21]. The experimental setup involves applying SET/RESET cycles to the phase change memory test chip using a dedicated Automated Test Equipment capable of generating arbitrary waveforms. The test environment can extract electrical characterization data from the test chip. The process and some of the extracted parameters can be found in [20] and [4]. As demonstrated in [20], the computational model can reproduce SET seasoning results that agree with the measured electrical characterization data.

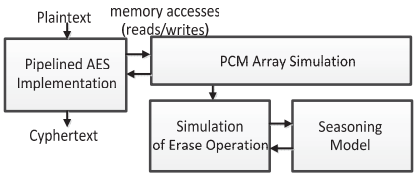

Figure 4: Simulation environment. Modeling of the phase change memory seasoning effect is based on the computational approach presented in [21] and the analytical model of erase operation in [4]. The simulation models are validated to match with the experimental results collected from actual phase change memory array in 180nm technology, see references of [4, 21].

Fig. 4 shows the simulation environment with the phase change memory model integrated with the simulation of AES hardware implementation. The AES implementation is a fully pipelined implementation of the AES cipher with 128 bits key. The implementation is in VHDL and tested on a Virtex-5 FPGA. Simulation of the AES cipher is integrated with the phase change memory modeling tools. The implementation stores temporary results on a RAM array of phase change memory. The modeled phase change memory array uses the following parameters: L is 30nm (GST thickness), A is $800nm^2$ (heater area), ρ_{c0} is 20 mΩ·cm (resistive constant of the crystalline phase at initial time before seasoning), ρ_a is 50 Ω·cm (resistive constant of the amorphous phase at initial time before seasoning), l_0 is 30nm (the length of the resistivity in the crystalline state before seasoning), R_h is 1KΩ (heater resistance), VPP is 4.8V, and V_{SEL} is 3.9V. The PCM is in 180nm technology with BJT selector.

We consider an offline attack scenario in which an attacker can physically access the device that contains the AES cipher and the phase change memory array after the device has been used for encrypting data. However, we assume that the AES key has been permanently deleted from the memory. Otherwise the attack becomes trivial. R_{GST} values of phase change memory cells can be measured using

```
1: procedure AES(input, key[])
2:     state ← input
3:     AddRoundKey (state, key[0])
4:     for i = 1 to 9 do
5:         SubBytes(state)
6:         ShiftRows(state)
7:         MixColumns(state)
8:         AddRoundKey (state, key[i])
9:     end for
10:    SubBytes(state)
11:    ShiftRows(state)
12:    AddRoundKey (state, key[10])
13:    Return state
14: end procedure
```

Figure 5: 128 bits AES encryption process.

test equipment such as the one presented in [4], which first collects I_{read} value for each memory cell and then convert I_{read} into R_{GST} using the equations in [4].

4. KEY RECOVERY ATTACK FOR AES

Within this section we give a detailed description of key recovery attack against AES when PCM has been utilized as memory.

4.1 Review of AES

Since the announcement of the AES standard [1], much work has been completed to analyze the security thereof. Up until now, all research results support AES as a secure encryption scheme without any weakness. It is believed that it is very challenging, if not impossible, to break AES with key length of 128 bits or longer. AES encryption consists of 11 rounds and for each round the round key is XORed with the intermediate result. Note that here we omit the key scheduling procedure and suppose all the 11 round keys are stored in a key array $key[11]$. The first round and the last round of the encryption process are different from other rounds. However, the round keys are used in the same way, i.e., XORed with the *state*. Fig. 5 describes the AES encryption process.

Here we consider the AES encryption process with a key size of 128 bits. Our method can be applied to the decryption process of AES and AES with longer keys as well.

4.2 Assumption of the Implementation

As we have mentioned before, side channel attacks depend heavily upon the characteristics of the hardware and software implementations. In this section, we clarify the assumptions of our method. We assume that the AES implementation is pipelined and it stores intermediate results in PCM memory array. In addition, we assume static memory locations are used for storing different intermediate results, i.e., in the process of encryption, different rounds employ different but static memory locations for storage. The assumption is reasonable for hardware based and pipelined implementation of AES. The AES encryption key is unknown to the attacker. For a given plain-text, the attacker can control or know the number of running times of the encryption. However, it is not required that the attacker knows the plain-text or be able to choose the plain-text. The attack can be conducted off-line by examining the PCM memory cells after the system encrypts a given plain-text. The attacker can use the measurement setup described in [4, 20] to probe the PCM memory cells and obtain R_{GST} values.

4.3 Key Recovery Process

Our aim is to recover all the 11 round keys. The overall idea is to check the seasoning effect of PCM cells to reveal information about the key, round by round. Specifically, bit changing is related to key pattern and seasoning effect is related to bit changing. So we can predict key value by examining seasoning effect of PCM cells.

First round key recovery. For the first round, suppose *state* is written to the specified memory area and then the XORed result of *state* and first round key is written to the same place. This operation may lead to seasoning effect of PCM cells:

- If the *state* bit is 1, and corresponding round key bit is 1, there is a strong seasoning effect;
- If the *state* bit is 0, and corresponding round key bit is 0, there is a weaker seasoning effect.

Because an attacker knows the value of *state* (it equals to the plain-text for the first round), he can predict the first round key bits if he also knows the seasoning effect level information. For the simplicity of this description, denote the set of bits that are changed as S_{change} and the set of bits that are not changed as S_{static}.

One of the troublesome problems is that the seasoning effect is hard to detect if a particular cell is flipped only several times more than the other. So we must find a way to magnify the seasoning effect. According to our assumption, the encryption process works in a pipelined manner and uses static memory locations for intermediate results, so we force the algorithm to encrypt the same plain-text multiple times to magnify the seasoning effect. Specially, *state* is written in the same memory location, and then the same round key is XORed with the *state*. In fact, this process equals two XOR operations with the round key, and bits in S_{change} are changed two times while bits in S_{static} are still unchanged. We can repeat this process until the seasoning effect is obvious enough to distinguish PCM cells that are changed a great number of times.

Table 1 gives an example of what we can learn from the seasoning effect information after enough repetitions. The first row stores the *state* value bit by bit (as it is the first round of AES encryption, these values are equal to the plain-text bits), which the attacker knows. The second row is for the first round key, which is unknown. The third row represents the seasoning effect information, h means this cell is changed a lot and ℓ means this cell is seldom changed. The attacker can predict the first round key in the following way:

- If the state is 1 and the seasoning effect level is h, then the correspond round key bit is 1;
- If the state is 1 and the seasoning effect level is ℓ, then the corresponding round key bit is 0;
- If the state is 0 and the seasoning effect level is h, then the correspond round key bit is 1;
- If the state is 0 and the seasoning effect level is ℓ, then the correspond round key bit is 0.

In this way, we can recover the first round key k_0, and reveal the result of the first round.

Table 1: Information from round 1

state	1	0	1	1	0	0	0	1
round key	x	x	x	x	x	x	x	x
cell status	h	h	h	ℓ	ℓ	ℓ	h	ℓ

Other round key recovery. Because we succeed in recovering the first round key and we know the plain-text, we can calculate the input *state* to the second round without much effort. Before `XOR`ed with the second round key, three operations (`SubBytes`, `ShiftRows`, `MixColumns`) are applied to *state*. These three operations are not related to any secret information and we can learn the treated *state* that to be `XOR`ed with second round key.

Now the situation becomes the same as the first round: we know the input and we can repeat the encryption process to magnify the seasoning effect. Armed with this information, an attacker can recover the second round key.

After recovering the second round key, the same method can be applied to recover the third round key. In this way, we can recover all the 11 round keys.

In practice, we do not need the encryption process to run round to round, i.e., running the first round several times before running the second round. We can repeat the encryption process as a whole to magnify the seasoning effect of cells and then do the measurement and analyses off-line.

4.4 Key Recovery Attack with Arbitrary Plain-text

Above key recovery attack assumes that the attacker knows the plain-text and allows the algorithm to encrypt the plain-text many times. In fact these requirements are dispensable.

The approach that a round key changes the *state* is through `XOR` operation. In this process, our concern is the flipping information, i.e., which bits of the *state* are changed. When a round key is applied to the *state*, the flips are completely determined by the round key itself. In other words, the round key determines the seasoning effect of PCM cells in the processing of encryption.

If different plain-texts are encrypted, another factor needs to be taken into consideration. Suppose a plain-text block m_1 is encrypted, and the intermediate results of the 11 rounds are denoted as $t_{1,0}, t_{1,1}, \ldots, t_{1,10}$. When a second plain-text block m_2 is coming, $t_{1,i}$ will be overwritten. In this case, some bits will change their values. However, the bit changes caused by overwritten are random, as it depends on both the previous intermediate result and the new plain-text block. So we can still learn the bit change patterns related to round keys. Section 5 provides more details of overwritten effects.

4.5 Eliminating Previous Usage Effects

As we have mentioned before, bit flips always result in seasoning effect of cells. Before starting our side channel analyses, the PCM may be used by other programs quite a bit which results in seasoning effect of cells. In this case, we need to eliminate these usage effects. To achieve this goal, we measure the seasoning effect of all cells that will be used for intermediate results. After completion of the analysis, we measure the seasoning effect of these cells again. As the seasoning process is known, we can calculate the difference of these two measurements and get the information needed for the round key recovery.

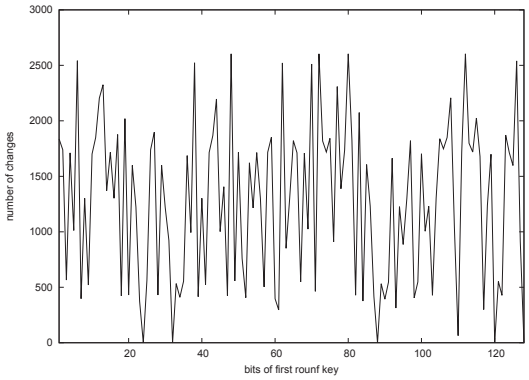

Figure 6: Bits changing information for the first round of AES encryption.

With this elimination method, our analysis is independent of seasoning effect information from PCM that are used by other applications.

4.6 AES with Other Key Sizes and Different Operation Modes

In the above discussion of key recovery attack, we assume that the key length is 128 bits. Our method can be easily extended to attack AES with longer key sizes such as AES 192 or AES 256. The only difference being that for each round it is required to measure the seasoning effect of additional cells.

Another issue that must be considered is operation modes. For real world application, AES is always used in conjunction with a certain operation mode [9]. For different operation modes, the input of the AES encryption function and treatment of plain-texts may vary. For our attack we only need the seasoning effect information, which depends on round keys. So different operation modes such as CBC (Cipher-Block Chaining), CFB (Cipher Feedback), and C-TR (Counter) will not affect and our attacking method is still valid.

5. SIMULATION AND EVALUATION

In this section, we do some simulation work on the side channel key recovery attack and evaluate the effectiveness. We use the 128 bits key given in [1] and a text file downloading from Internet (`www.textfiles.com/stories/100west.txt`) for the bit changing simulation. The file size is 20839 bytes and use the first 20832 bytes (1302 blocks) for the simulation. Fig. 6 shows the simulation result for the first round, where the round key is

`2b7e151628aed2a6abf7158809cf4f3c`.

In this figure, each peak point corresponds to bit value 1 and each valley point corresponds to bit value 0. The simulation result reflects the round key value accurately. Fig. 7 shows the simulation result for another round key

`cf4f3c8809bf7152a6a8aed51622b7e1`.

It can be seen that the round key totally determines the bits changing pattern.

Fig. 8 illustrates bits changing information for the other 10 rounds. Like in the first round, one can predict the

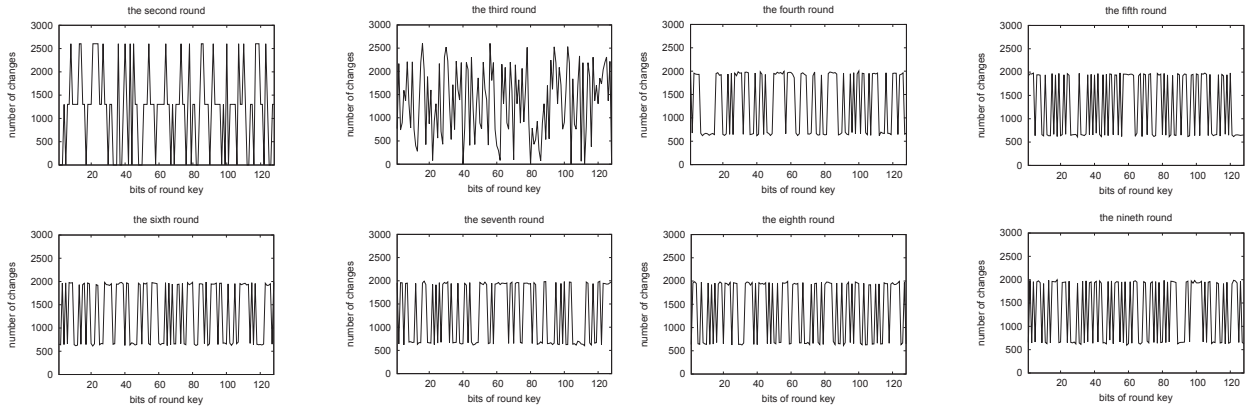

Figure 8: Bits changing information for round 1 to round 9.

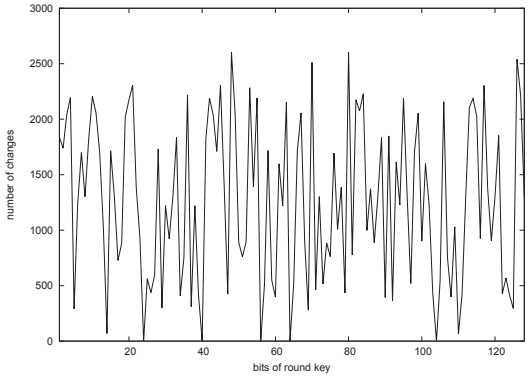

Figure 7: Bits changing information for the first round of AES encryption with another key.

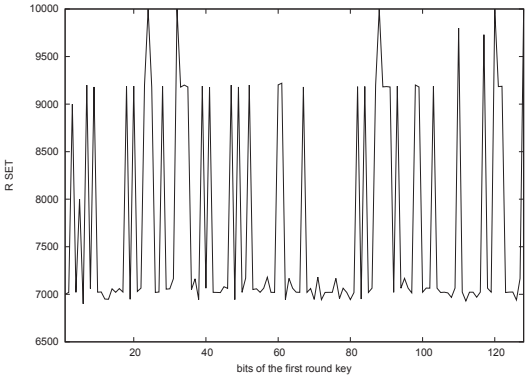

Figure 9: Relationship between seasoning effect and bit changing.

round key value easily from the shape of the curve. Although the intermediate results become more random for the later rounds, the bit changing patterns do not change as they are determined by the value of the round keys. Note that from the fourth round, nearly all the bits are flipped at least 651 times. These changes are due to rewriting of the intermediate results and reflects the fact that AES encryption process can make the distribution of intermediate results evenly with several rounds.

In practice, we can infer the differences from seasoning effect of cells. Data from [21], Fig. 9 demonstrates the seasoning effect of cells caused by bits changing. One can learn the value of the first round key from the shape of the curve reflecting seasoning effect. The pattern for the other rounds is similar.

6. COUNTERMEASURES TO OUR KEY RECOVERY ATTACK

Side channel attacks are highly tailored depending on the software and hardware utilized. According to our key recovery attack method, several countermeasures can be exercised to protect the round keys. The fundamental principle for this attack is that some usage patterns are left in the memory even if all the data have been erased. To prevent this side channel attack, one needs only to eliminate such patterns. Several techniques can be applied to achieve this goal and we discuss two of them that can be easily deployed.

- Memory location obfuscation. With memory location obfuscation, each operation in the process of encryption may save data in a random area of the memory, and it is hard for the attacker to repeat the process to magnify the seasoning effect and detect the seasoning effect pattern.
- Memory re-use. Memory re-use means the encryption program saves the intermediate result of every round in the same place, and in this way the seasoning effect of memory cells tends to be more random and it is harder to learn useful information about the round keys.

7. RELATED WORK

There has been little work that investigates how electrical characteristics of the emerging universal RAM such as phase change memory can facilitate certain side-channel attacks to the standard encryption schemes. There has been a great deal of research on other types of side channel exploits that target AES implementation such as cache-timing based attack (e.g., [2, 10]). Some of the cache-timing based at-

tacks could result in the entire AES key to be compromised. However, most of these attacks require real-time measurement of the timing information. Some even require the ability by the attacker to perform chosen plaintext attacks. Other side-channel attacks on AES involve real-time fault injection to a victim system or rely on differential analysis (e.g., [16]). Different from these prior researches on side-channel exploits of AES implementation, our study focuses primarily on the seasoning effect of phase change memory and evaluates its effectiveness as a side-channel of AES implementation that uses the emerging phase change memory. The issue becomes important as phase change memory is considered as a potential candidate of universal RAM that may replace SRAM and DRAM in the future. Commercial products of phase change memory have started to appear in the market. In addition, our described side-channel attack to AES is off-line based. Unlike most of the cache-timing based and fault injection based attacks, our attack based on the seasoning effect of phase change memory doesn't require real-time or online monitoring of a victim system. Different from these intrusive attacks such as fault injection, our attack doesn't require the capability of intervening normal operations of an AES based system. The attack can be conducted off-line by probing the seasoning effect levels of phase change memory array.

8. CONCLUSION

Within this work, we propose a side channel attack on AES when PCM based memory array is utilized to store intermediate results of encryption/decryption. We discuss how to use this attack to recover round keys for several different operation modes and show how effective the attack can be if the system is not carefully designed and implemented. We developed a simulator of phase change memory array that employs state-of-the-art analytical model of the seasoning effect. Furthermore, we conducted experiments to show the feasibility of applying the attack in realistic settings. The experimental study demonstrates effectiveness of the attack. Finally, we provide some suggestions for prevention of such key recovery attacks when PCM is used as the memory of AES implementation.

Acknowledgment

We thank all the reviewers for their valuable comments.

9. REFERENCES

[1] Fips pub 197: Advanced encryption standard, November 2001.

[2] D. J. Bernstein. Cache-timing attacks on AES, 2004. URL: `http://cr.yp.to/papers.html#cachetiming`.

[3] E. Brier, C. Clavier, and F. Olivier. Correlation power analysis with a leakage model. In M. Joye and J.-J. Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004*, volume 3156 of *LNCS*, pages 16 – 29. Springer, 2004.

[4] A. Chimenton, C. Zambelli, and P. Olivo. A new analytical model of the erasing operation in phase-change memories. *Electron Device Letters, IEEE*, 31(3):198–200, 2010.

[5] K. Gandolfi, C. Mourtel, and F. Olivier. Electromagnetic analysis: Concrete results. In Çetin K. Koç, D. Naccache, and C. Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2001*, volume 2162 of *LNCS*, pages 251 – 261. Springer, 2001.

[6] D. Genkin, A. Shamir, and E. Tromer. Rsa key extraction via low-bandwidth acoustic cryptanalysis. Cryptology ePrint Archive, Report 2013/857, 2013. `http://eprint.iacr.org/`.

[7] T. S. Messerges, E. A. Dabbish, and R. H. Sloan. Power analysis attacks of modular exponentiation in smartcards. In Çetin K. Koç and C. Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 1999*, volume 1717 of *LNCS*, pages 144 – 157. Springer, 1999.

[8] T. S. Messerges, E. A. Dabbish, and R. H. Sloan. Examining smart-card security under the threat of power analysis attacks. *IEEE Transactions on Computers*, 51(1):541 – 552, January 2002.

[9] M. orris Dworkin. NIST SP 800-38A Recommendation for Block Cipher Modes of Operation, 2011.

[10] D. Osvik, A. Shamir, and E. Tromer. Cache attacks and countermeasures: The case of AES. In D. Pointcheval, editor, *Topics in Cryptology CT-RSA 2006*, volume 3860 of *LNCS*, pages 1–20. Springer, 2006.

[11] S. R. Ovshinsky. Reversible electrical switching phenomena in disordered structures. *Physical Review Letters*, 21:1450 – 1453, 1968.

[12] J.-B. Park, G.-S. Park, H.-S. Baik, J.-H. Lee, H. Jeong, and K. Kim. Phase-change behavior of stoichiometric ge2sb2te 5 in phase-change random access memory. *Journal of the Electrochemical Society*, 154(3):139 – 141, 2007.

[13] E. Peeters, F.-X. Standaert, N. Donckers, and J.-J. Quisquater. Improved higher-order side-channel attacks with FPGA experiments. In J. R. Rao and B. Sunar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2005*, volume 3659 of *LNCS*, pages 309 – 323. Springer, 2005.

[14] S. Raoux, G. W. Burr, M. J. Breitwisch, C. T. Rettner, Y.-C. Chen, R. M. Shelby, M. Salinga, D. Krebs, S.-H. C. H.-L. Lung, and C. H. Lam. Phase-change random access memory: A scalable technology. *IBM Journal of Research and Development*, 52(4/5):465 – 479, 2008.

[15] M. Renauld, F.-X. Standaert, and N. Veyrat-Charvillon. Algebraic side-channel attacks on the AES: Why time also matters in DPA. In C. Clavier and K. Gaj, editors, *Cryptographic Hardware and Embedded Systems - CHES 2009*, volume 5747 of *LNCS*, pages 97 – 111. Springer, 2009.

[16] D. Saha, D. Mukhopadhyay, and D. RoyChowdhury. A diagonal fault attack on the advanced encryption standard. Cryptology ePrint Archive, Report 2009/581, November 2009.

[17] J. Sarkar and B. Gleixner. Evolution of phase change memory characteristics with operating cycles: Electrical characterization and physical modeling. *Applied Physics Letters*, 91(23):233506–233506–3, 2007.

[18] W. Schindler, K. Lemke, and C. Paar. A stochastic model for differential side channel cryptanalysis. In

J. R. Rao and B. Sunar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2005*, volume 3659 of *LNCS*, pages 30 – 46. Springer, 2005.

[19] D. Ventrice, P. Fantini, A. Redaelli, A. Pirovano, A. Benvenuti, and F. Pellizzer. A phase change memory compact model for multilevel applications. *Electron Device Letters, IEEE*, 28(11):973–975, 2007.

[20] C. Zambelli, A. Chimenton, and P. Olivo. Empirical investigation of SET seasoning effects in phase change memory arrays. *Solid-State Electronics*, 58(1):23 – 27, 2011.

[21] C. Zambelli, A. Chimenton, and P. Olivo. Modeling of SET seasoning effects in phase change memory arrays. *Microelectronics Reliability*, 52(6):1060 – 1064, 2012.