Leakage Resilient of Timing Side Channel Attack for Key Exchange Security Model

Clement Chan Zheng Wei, Chuah Chai Wen

Abstract: ***In leakage resilient cryptography, leakage resilient key exchange (KE) protocols are constructed to resist side channel leakage attack. Side channels attacks take place during the execution of the cryptographic schemes or protocols. For a KE protocols to stay secure, counter measures must be employed on the cryptographic primitives instantiated during the protocol building to counter side channel leakage attacks. This work propose a leakage resilient key derivation function (KDF) primitive to be instantiated on KE protocol. Then, this work proceed to define a security model to show the leakage resilient (KDF) is provably secure using indistinguishability game-hopping technique. Lastly, this work revisit the KE protocol proposed by Alawatugoda and construct an improved leakage resilient KE protocol by instantiating our proposed leakage resilient KDF.***

Index Terms: Key Exchange Protocol, Leakage Resilient Cryptography, Security Models, Timing Analysis Attack;

I. INTRODUCTION

Leakage resilient cryptography is an approach on resisting side channel attacks (SCAs) against a cryptographic scheme or protocol. SCAs place during the execution of the cryptographic schemes or protocols, where an adversaries can obtain information of parameters leakage involved in the computations. These leakage information about the primitives leaked to the adversary goes beyond the security prediction by the protocol designer which allow the adversary to break an cryptographic protocol laterally. Leakage resilient designing approach takes on side channel point of view to provide secure implementation of a protocol under the presence of SCAs. Examples of SCAs are memory attack [1], power analysis attack [2], electromagnetic attack [3], faulty bug attacks [4] and timing analysis attack [5]–[7]. Such SCA needed to be counter measured, the methods are either (1) a software-based approach [8] by masking the leaking timing information and stopping the information leakage, or (2) a hardware-based approach [9] which work around the design of hardware to minimize the information leakage amount. Leakage resilient cryptography also applied on a KE protocol to address strong leakage attacks, thereby exist the notion of leakage resilient KE protocols which are resilient to SCAs. Examples of leakage resilient KE protocols can referred to [10]–[13].

Revised Manuscript Received on May 22, 2019.

Clement Chan Zheng Wei, Information Security Interest Group(ISIG), Faculty of Computer Science and Information Technology, University Tun Hussein Onn Malaysia, Malaysia

Chuah Chai Wen, Information Security Interest Group(ISIG), Faculty of Computer Science and Information Technology, University Tun Hussein Onn Malaysia, Malaysia

The building and design of a KE protocol involves multiple cryptographic primitives and such primitive example are key derivation function (KDF).

Timing analysis attack is one of the SCA where the adversary gets side information by analysing the execution time acquire from a cryptographic protocol computation. Here the adversary knows a set of protocol parameters as well as the corresponding running time of the cryptographic protocol needs to process these parameter inputs. With the pattern analysis involving both the relationship between the execution time and parameter inputs, the adversary is able to derive useful side channel information about the secret key.

The work [7] by ChaiWenChuah et al. study extensively the actual practical deployment of timing analysis attack in real-world cryptographic application. They model timing attack against KDF based on stream ciphers, hash functions and block ciphers. The execution time taken to generate a cryptographic key by the KDF is measured with different parameter inputs. Due to the design of KDF primitive, certain types of KDF are input length dependent execution which allow the adversary to derive notable pattern about a KDF. Adversary is able to distinguish whether a session key or secret key is generated by which specific type of KDF through timing analysis of side information leakage. This allows the attacker to go beyond the intended security prediction of a protocol if leakage resilient notion is not consider during the protocol building.

This paper proposes to extends the results from previous works on modelling leakage resilient KDF security model [14] by ChaiWenChuah et al. and improve a weakness of KDF primitive in leakage resilient key exchange(KE) protocol by Alawatugoda[11]. This work discuss about (1) the insecurity of protocol π[11]by Alawatugoda incorporated from the vulnerability of timing analysis exploitation in KDF execution, (2) the theoretical software-based approach of modification on KDF primitive to thwart timing analysis attack, (3) the construction on extending the result of KDF security model by ChaiWenChuah to address timing analysis attack, (4) the comparison on security properties of this proposed work with previous existing works.

II. LEAKAGE RESILIENT KEY DERIVATION FUNCTION

KDF is a common cryptographic primitive used in the protocol building of a KE protocol. In the KE protocol, KDF accepts cryptographic inputs and transform it into an arbitrary bit length of pseudorandom string that can be directly used as a cryptographic key or in

further cryptographic execution. Definition of secure and efficient KDF by Hugo Krawczyk [15] on HMAC [16] is reviewed.

In this paper, to define a leakage resilient KDF, this work proposed a theoretical software-based approach [8] by masking the leaking timing information of a KDF to countermeasure timing analysis attack. The approach here is to make the execution time independent of the secret key. The execution time of deriving a cryptographic key by the KDF is randomized (through an efficient and invertible randomized transformation) to mask the pattern of relationship between the execution time and parameter inputs involved.

A definition of KDF in term of its input parameters and outputs is given. Later, the notion of leakage resilient of side channel information for a KDF is introduced, then only define what is meant for a KDF to be side channel leakage resilient. Thereby, a formalization of leakage resilient KDF.

Definition 2.0.1 (KDF) [15] A $KDF$is define as: $K \leftarrow KDF(PrivS, s, ctx, n)$, where K is the n bit length derived cryptographic key of the KDF, PrivS is a private seed, s is a public salt, ctx is a public context string and n is a positive integer that indicate the bit length of the cryptographic key, whereby $|PSPACE|$ is the distribution of private seed space, $|SSPACE|$ is the distribution salt space and $|CSPACE|$ is the distribution context space.

Definition 2.0.2 (Side channel leakage – timing analysis) Let $scr_i \xleftarrow{R} \sum K_i$ denoted as the leakage distribution for the randomized execution time of KDF to compute a cryptographic key, in condition where '$\xleftarrow{R}$' denote randomized execution time, where for each i^{th} execution, a randomized execution time scr_i is output from the leakage distribution $\sum K_i$ for a KDF to compute a cryptographic key K_i.

Definition 2.0.3 (Security of KDF with respect to side channel leakage) For a KDF be a leakage resilient KDF, we define ε as the advantage $Adv_{KDF}(A)$ for PPT adversary A to win the following distinguishing game with probability not significantly greater than $\left(\frac{1}{2} + \varepsilon\right)$:

1. The algorithm accepts a $PrivS$ from the distribution of $|PSPACE|$.
2. For $i = 1, \ldots, q' \leq q$: Adversary A chooses arbitrary values of s_i from $|SSPACE|$ and ctx_i from $|CSPACE|$ and receive the value of $KDF(PrivS, s, ctx, n)$ with the randomized execution time of KDF to compute a cryptographic key, $scr_i \xleftarrow{R} \sum K_i$. (These learning queries are adaptive)
3. Adversary A choses values for s from $|SSPACE|$ and ctx from$|CSPACE|$ in such$\{s, ctx\} \notin \{s_i, ctx_i\}, \ldots, \{s_q', ctx_q'\}$.
4. A bit $b \xleftarrow{R} \{0,1\}$ is chosen at random, if $b = 0$, adversary A receives outputs of $KDF(PrivS, s, ctx, n)$ with the randomized execution time from $scr_i \xleftarrow{R} \sum K_i$, else A is given a random string of n length.
5. Step (2) is repeated up to $q - q'$ queries such that $\{s, ctx\} \notin \{s_i, ctx_i\}, \ldots, \{s_q', ctx_q'\}$.
6. Adversary A outputs a bit $b' \in \{0,1\}$. A wins if $b' = b$.

Remarks: It is important for the applicability of (definition 2.0.1 – 2.0.3) that the adversary is given the execution time of the cryptographic key by the KDF. This definition is achieved when the KDF can remain secure even when the side information scr(execution time of computing a cryptographic key) with s salt and ctx context information are known to the attacker. Since with the notion where the execution time is randomized through the proposed theoretical software-based approach [8] by masking the leaking timing information of a KDF to countermeasure timing analysis attack, the KDF should shows no pattern interpretation for an adversary to learn side information and break the security of KDF trivially (leak no useful information).

III. FRAMEWORK OF TIMING SIDE CHANNEL LEAKAGE RESILIENT SECURITY MODEL FOR KDF

To prove the security of cryptographic primitive such as KDF in the KE protocol, formal security models to analyse KDF have been introduced by Hugo Krawczyk [15], Yao & Yin [17] and ChaiWenChuah et al. [14]. However, these existing frameworks (e.g., CPM model by ChaiWenChuah et al. [14]) on KDF do not address adversary capabilities with timing analysis attack.

In this paper, this work proposed a security model called 'CPM-R'. For a KDF to declare CPM-R secure, the adversary can choose for a salt value with a context information as inputs for the KDF and is given a side channel randomness which is the execution time of a cryptographic key by the KDF, additionally the adversary cannot win the indistinguishability game with the probability higher than guessing probability.

Definition 2.1 {CPM-R secure}: The KDF is (t, q, ε) CPM-R secure if for all adversary A running in polynomial time t and making at most $q < |SSPACE| \times |CSPACE|$ queries to the KDF with chosen salt value and chosen context information and the execution time of KDF for deriving the cryptographic key as side channel information win the following indistinguishability game with probability not higher than$\left(\frac{1}{2} + \varepsilon\right)$.

Table. 1 Security model CPM-R

Learning stage	1. C chooses $PrivS \leftarrow PSPACE$. 2. For $i = 1, \ldots, q' \leq q$,	(2.1) A chooses $s_i \leftarrow SSPACE$ and $ctx_i \leftarrow CSPACE$. (2.2) C computes $K_i = KDF(PrivS, s_i, ctx_i)_n$ (2.3) A is provided the derived cryptographic key, K_i with the $scr_i \leftarrow \sum K_i$.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Challenge stage	1. A chooses $s \leftarrow SSPACE$ and $ctx \leftarrow CSPACE$. (subject to restriction$\{s, ctx\} \notin \{s_i, ctx_i\}, \ldots, \{s_q', ctx_q'\}$) 2. C chooses $b \xleftarrow{R} \{0,1\}$. 3. C sends K' with $scr \leftarrow \sum K$ to$A$.	(2.1) If $b = 0$, C outputs $K' = KDF(PrivS, s, ctx)_n$ (2.2) else C outputs $K' \xleftarrow{R} \{0,1\}^n$.
Adaptive stage	1. Step 2 in Learning stage is repeated for up to $q - q'$queries (subject to restriction $\{s_i, ctx_i\} \neq \{s, ctx\}$). 2. A outputs $b' = 0$, if A believes that $K'$is cryptographic key, else outputs $b' = 1$.	
A wins the game if $b' = b$.		

The 3 major stages in CPM-R security framework are:

Learning Stage – In this stage, a PrivS is chosen from the $|PSPACE|$, meanwhile the adversary is able to choose for s from $|SSPACE|$ and ctx from $|CSPACE|$. The adversary is able to make query count up to $q_i < |SSPACE| \times |CSPACE|$. For each query made by adversary, the adversary is given a cryptographic key $K_i = F_R(\text{PrivS}, s_i, ctx_i)_n$ derived by KDF using the associated private seed, salt and context string chosen follow with a side channel randomness 'scr_i' which is the execution time of KDF to compute the cryptographic key.

Challenge Stage – The adversary is able to choose for the salt and context string as public inputs for the KDF in challenge stage as long as the adversary follow the restriction where the chosen public inputs must not be the same set of input chose before in the learning stage. In challenge stage, a random bit $b \xleftarrow{R} \{0,1\}$ is randomly generated. If the random bit output is $b = 0$, then a real cryptographic key $K' = F_R(\text{PrivS}, s_i, ctx_i)_n$ is derived, otherwise a random bit string $K' \xleftarrow{R} \{0,1\}^n$ of n bit length is generated. The output with the execution time of KDF 'scr' is the given to the adversary.

Adaptive Stage – The adaptive stage consists of the repetitive iteration of learning stage where the adversary is able to learn more information before answering whether the cryptographic key 'K'' is the real session key derived from the KDF or a random bit string. The adversary is required to follow a restriction where the adaptive learning query must not ask for any public inputs directly related to the inputs set in the challenge stage.

The final step of the game is the adversary guesses whether the cryptographic key 'K'' is the real session key or a random bit string. Adversary will submit $b' = 0$ if the adversary think that the K' is the real session key from KDF otherwise the adversary will submit $b' = 1$. The adversary wins the game if $b' = b$. We want the probability that the adversary win the guessing game not higher than $\Pr[b' = b] < 12 \pm \varepsilon$, where ε is negligible, thereby achieve the claim of timing analysis side channel leakage resilient.

IV. SECURITY PROOF FOR KDF SECURITY MODEL CPM-R

This section shows the relationship of proposed model CPM-R with existing model, where CPM-R should be covering wider range of attacks specifically side channel timing analysis attack. This indicate CPM-R is a stronger security model than existing model during the occurrence of SCA. This section establishes a more precise relations between our proposed CPM-R security model and existing KDF security model, the relationships include the implications and non-implications between security models.

A. Implications between Security Models

Security proof for the implication relationships of CPM-R between other security models can be proven by comparing existing CPM model [14] because CPM possesses the strongest adversary capability/notion.

Lemma 1: $CPM\text{-}R \Rightarrow CPM$

Proof: Assume that a KDF is CPM-R secure but not CPM secure. There exists an adversary A who able to win the CPM game with probability greater than $(\frac{1}{2} + \varepsilon)$. An assumption is made where exists an adversary B who plays the CPM-R game with C. Adversary B will take advantage of capability A, whereby adversary A is playing CPM game with B meanwhile B is playing the CPM-R game with C. C sends K' with $scr \leftarrow \sum K$ to A.

Based on the following game in table 2, the probability of B wins the CPM-R game is equal to the probability of A wins the CPM game. The assumption is that the KDF is not CPM secure, thereby the probability of A wins the CPM game is greater than $\frac{1}{2} + \varepsilon$. Hence, B is able to wins the CPM-R game with probability greater than $\frac{1}{2} + \varepsilon$.

The $A \overset{CPM}{\Longrightarrow} B \overset{CPM-R}{\Longrightarrow} C$ game is conducted as following:

Table. 2 Security game played between adversary *A* with *B* and *B* with *C*

Learning stage	1. C chooses $PrivS \leftarrow PSPACE$. 2. For $i = 1, \ldots, q' \leq q$,	(2.1) A chooses $s_i \leftarrow SSPACE$ and $ctx_i \leftarrow CSPACE$ and sends it over to B. (2.2) Bforwards s_i and ctx_i to C. (2.3) C computes $K_i = KDF(PrivS, s_i, ctx_i)_n$. (2.4) B is provided the derived cryptographic key, K_i with the $scr_i \leftarrow \sum K_i$. (2.5) B forwards K_i with the $scr_i \leftarrow \sum K_i$ to A.

Challenge stage	1. A chooses $s \leftarrow SSPACE$ and $ctx \leftarrow CSPACE$ and sends it over to B. (subject to restriction$\{s, ctx\} \notin \{s_i, ctx_i\}, \ldots, \{s_q', ctx_q'\}$) 2. B forwards s and ctx to C. 3. C chooses $b \overset{R}{\leftarrow} \{0,1\}$. 4. C sends K' with $scr \leftarrow \sum K$ to B. 5. B forwards K' with $scr \leftarrow \sum K$ to A.	(3.1) If $b = 0$, C outputs $K' = KDF(PrivS, s, ctx)_n$ (3.2) else C outputs $K' \overset{R}{\leftarrow} \{0,1\}^n$.
Adaptive stage	1. Step 2 in Learning stage is repeated for up to $q - q'$queries (subject to restriction $\{s_i, ctx_i\} \neq \{s, ctx\}$). 2. A outputs $b' = 0$, if A believes that $K'$is cryptographic key, else outputs $b' = 1$. 3. A sends b' to B then B forwards b' to C.	
B wins the game if $b'_A = b_C$.		

The result of indistinguishability game shows that the KDF is not CPM-R secure.

- Proven that (lemma 1) CPM-R ⇒ CPM, if the KDF is not CPM secure, then it is not CPM-R secure.

B. Non-Implications between Security Models

To prove the non-implication between the security models (corollary 1), two KDFs had been analysed based on table 3. Of these two KDFs, KDF1 is proposals by Hugo Krawczyk [15], and KDF2 is proposals by this work on (Definition 2.0.3).

Table. 3 Summary of security analysis of KDF proposals based on the proposed security framework for KDF

Theorem	KDF proposals	CPM	CPM-R
1	$KDF1(PrivS, s, ctx)_n = F(F(PrivS, s), ctx)$[15]	✓	×
2	$KDF2(PrivS, s, ctx)_n = KDF_R(Privs, s, ctx)_n \wedge scr_i \overset{R}{\leftarrow} \sum K_i$ (Definition 2.0.3)	✓	✓

KDF1: $KDF1(PrivS, s, ctx)_n = F(F(PrivS, s), ctx)$[15]
Theorem 1: KDF1 is secure respect to CPM model but not secure in CPM-R model, under timing analysis attack.
Proof: Security of KDF1 is already secure proven in CPM model by ChaiWenChuah et.al [14]. Next, KDF1 is shown not secure in CPM-R model during timing analysis attack. Adversary A can distinguish the secret key K' is the actual key generated using KDF1 or a random string of the same length during the occurrence of timing analysis attack, if only if:

- In CPM-R model, for some queries $i = 1, \ldots, q' \leq q$, where $scr_i \leftarrow \sum K_i$ the execution time of KDF to compute a cryptographic key is given to the adversary, meaning that$KDF1(PrivS, s, ctx)_n = F(F(PrivS, s), ctx)$ with $scr \leftarrow \sum K$. Adversary A can distinguish K' is one of the key generated by the KDF through pattern analysis of execution time by SCA.

The probability that A winning KDF1 indistinguishability game is always:$\Pr[A\text{wins}] \geq \frac{1}{2}$

Adversary A able to distinguish the challenge output with probability higher than guessing due to the occurrence of timing analysis attack, whereby the adversary can successfully determine whether the challenge output is generated by the KDF through pattern analysis of execution time. Therefore, KDF1 is CPM secure but not secure in CPM-R model under timing analysis attack.

- Proven that (corollary 1) CPM ↛ CPM-R from immediate result of theorem 1, if the KDF is CPM secure, it may not be CPM-R secure under SCA.

KDF2:$KDF2(PrivS, s, ctx)_n = KDF_R(Privs, s, ctx)_n \wedge scr_i \overset{R}{\leftarrow} \sum K_i$ (Definition 2.0.3)
Theorem 2: KDF2 is secure respect to both CPM and CPM-R security model, under timing analysis attack.
Proof: Firstly, KDF2 is shown to be CPM-R secure under timing analysis SCA. During the learning stage of the security game, C chooses $PrivS \leftarrow PSPACE$. C computes $K_i = KDF2(PrivS, s_i, ctx_i)_n$, where value of s_i and ctx_i is chosen by adversary A from $SSPACE$ and $CSPACE$. A is provided the derived cryptographic key, K_i with the corresponding execution time by KDF to compute the key, $scr_i \leftarrow \sum K_i$. During the challenge stage of the security game, the challenge key is computed as $K' = KDF2(PrivS, s, ctx)_n$, where the s and ctx are chosen by adversary A. C sends K' to A, adversary A can continues to learn the cryptographic key by adaptively repeating the learning stage up to $q - q'$ queries. Adversary A can only distinguish K' is the actual key generated using KDF2 or a random string of the same length, if only if:

- For some queries $i = 1, \ldots, q' \leq q$, where $\{s_i, ctx_i\} = \{s, ctx\}$, meaning that $KDF2(PrivS, s, ctx)_n = KDF2(PrivS, s_i, ctx_i)_n$. Adversary A can easily distinguish K' is one of the key generated before in learning stage because the salt and context variable used is the same on both stage. However, based on Definition 2.0.3, the pair of $\{s_i, ctx_i\}$ chosen in learning stage is restricted not to be similar in challenge stage. Thus, the probability of $\Pr[\{s_i, ctx_i\} = \{s, ctx\}] = 0$.
- Adversary can guess the $PrivS$ used in the computation by KDF2 such that the guessing $PrivS'$ is the same as the $PrivS$ used in the actual computation, $KDF2(PrivS', s, ctx)_n = KDF2(PrivS, s_i, ctx_i)_n$. Thus, adversary can win with probability of $\Pr[PrivS' = PrivS] \leq \frac{q_{PrivS}}{|PSPACE|}$.
- For some queries $i = 1, \ldots, q' \leq q$, where $scr_i \leftarrow \sum K_i$ the execution time of KDF to compute a cryptographic key is given to the adversary, meaning that $KDF2(Privs, s, ctx)_n \wedge scr_i \leftarrow \sum K_i$.

Adversary can try to distinguish K' is one of the key generated by the KDF through pattern analysis of timing by SCA However, based on Definition 2.0.2, KDF2 is a leakage resilient KDF where the leaked execution time is randomized '$\overset{R}{\leftarrow}$'. Thus, the adversary cannot learn any useful analysis of pattern from timing leakage attack.

The probability of adversary winning the indistinguishability game is:

$$\begin{aligned}\Pr[A\text{wins}] &= \Pr[A\text{wins}|PrivS' = PrivS]\Pr[PrivS' = PrivS] \\ &\quad + \Pr[A\text{wins}|PrivS' \neq PrivS]\Pr[PrivS' \neq PrivS] \\ &\leq 1\left(\frac{q_{PrivS}}{|PSPACE|}\right) + \frac{1}{2}\left(1 - \frac{q_{PrivS}}{|PSPACE|}\right) \\ &\leq \frac{1}{2} + \frac{q_{PrivS}}{2|PSPACE|}, \text{ where } \epsilon \\ &= \frac{q_{PrivS}}{2|PSPACE|} \text{ is negligible.}\end{aligned}$$

Adversary A can only distinguish the challenge output key with negligible probability. Therefore, KDF2 is CPM-R secure under timing analysis attack. Hence, KDF2 is secure under CPM-R and CPM by Lemma 1.

- Proven definition 2.0.3 leakage resilient KDF is secure under CPM-R and CPM model even under timing analysis attack.

V. INSECURITY OF KEY EXCHANGE PROTOCOL π[11]

In this section, this work highlight the KE protocol 'π' by Alawatugoda [11] which proven to be secure under weaken bounded after-the-fact leakage eCK model(wBAFL-eCK) [11]. The building of protocol π involves a leakage resilient public key encryption scheme(PKE), leakage resilient signature scheme(SIG) meanwhile only mention a secure KDF is required without further probing extensively into the security of KDF. The detail security primitives definition of PKE, SIG and KDF of protocol π can be found in [11] and is omitted to show in this work. But, the protocol π shows a security weakness in its protocol building, more specifically the KDF primitive implementation in protocol π is vulnerable to timing analysis SCA. Its KDF does not provide any protection measure against timing analysis attack.

A. Timing Analysis Resilient Protocolπ

Table. 4 Modified timing analysis resilient protocol π[11]

A (Initiator)		B (Responder)
$sk_A, vk_A \xleftarrow{\$} KG(1^k)$ $s_A, p_A \xleftarrow{\$} KeyGen(1^k)$	**Initial setup**	$sk_B, vk_B \xleftarrow{\$} KG(1^k)$ $s_B, p_B \xleftarrow{\$} KeyGen(1^k)$
$r_A \xleftarrow{\$} \hat{C}$ $\widetilde{r_A} \leftarrow \underline{Dec}(s_A, r_A)$ $X_A \leftarrow g^{\widetilde{r_A}}$ $\sigma_A \xleftarrow{\$} \underline{Sign}(sk_A, (A, B, X_A))$	**Protocol execution** $\xrightarrow{A,B,X_A,\sigma_A}$ $\xleftarrow{B,A,X_B,\sigma_B}$	If Vfy(vk_A, X_A, σ_A) = "true" { $r_B \xleftarrow{\$} \hat{C}$ $\widetilde{r_B} \leftarrow \underline{Dec}(s_B, r_B)$ $X_B \leftarrow g^{\widetilde{r_B}}$ $\sigma_B \xleftarrow{\$} \underline{Sign}(sk_B, (B, A, X_B))$

If Vfy$(vk_B, (B, A, X_B), \sigma_B)$ = "true" {
$\widetilde{r_A} \leftarrow \underline{Dec}(s_A, r_A)$
$\boldsymbol{ms} \leftarrow \boldsymbol{KDF_R}\left(\boldsymbol{X_B^{\widetilde{r_A}}}, \perp, \boldsymbol{k}, \perp\right)$
$K \leftarrow PRF(ms, A\|X_A\|\sigma_A\|B\|X_B\|$
}
	K is the session key.	$\boldsymbol{ms} \leftarrow \boldsymbol{KDF_R}\left(\boldsymbol{X_A^{\widetilde{r_B}}}, \perp, \boldsymbol{k}, \perp\right)$
		$K \leftarrow PRF(ms, A\|X_A\|\sigma_A\|B\|X_B\|$
		}	

Table 4 shows a modified protocol π of Alawatugoda [11]. Protocol π is a Diffie-Hellman type [18] key exchange protocol. The modification made on the protocol execution part where a timing analysis leakage resilient KDF based on (Section II -Definition 2.0.3) is used. After both party exchanging the public values, leakage resilient KDF_R is used to generate a shared secret key using the Diffie-Hellman type shared secret value computed earlier. Through leakage resilient KDF, the execution time for the KDF to generate the cryptographic secret output 'ms' is masked. Any adversary should not be able to deduce any useful pattern of analysis to determine the cryptographic secret output by the KDF in the distinguishing game. Thus, adversary cannot break the security game trivially with probability higher than guessing probability.

B. Security Proof for Timing Analysis Resilient Protocol π

Proof: Let A be an adversary that can adaptively issue queries at most q times in the security game which played by KDF challenger C and adversary A. The advantage for adversary A to break the security of Timing Analysis Resilient Protocol π is bounded by the success probability of adversary guessing whether the 'ms' value is computed using the KDF or randomly chosen. Note that the security of the protocol is tested specifically under timing analysis attack towards KDF scheme, the proofs of all games in the original protocol π are similar presented in [11, Appendix B] except the game involving the KDF challenger.

KDF game:

1. The KDF challenger C randomly chooses a bit $b \xleftarrow{R} \{0,1\}$.
2. If $b = 0$, C sends the real ms value computed by the KDF to adversary A, else A is given a random string of same length.
3. A then uses the received ms value to compute the session key.
4. A distinguish whether the ms value is computed using the KDF or randomly chosen by outputs a bit $b' \in \{0,1\}$. A wins if $b' = b$.

Remarks: The adversary should not have probability greater than $\frac{1}{2}$ to successfully distinguish whether the ms value is computed using the KDF or randomly chosen, the advantage of adversary win the distinguishing game with probability not greater than $Adv_{KDF}(A) \leq \frac{1}{2}$.

The KDF used is a timing analysis leakage resilient KDF based on (Section II -Definition 2.0.3) thereby there is no pattern analysis to be deduced to determine the ms value is computed by the KDF in the protocol.

VI. COMPARISON OF SECURITY MODELS AND PROTOCOLS

In this work, a security model namely CPM-R for KDF is presented, addressing the side channel leakage specifically timing analysis attack during the execution of protocol. Further, a leakage resilient key exchange protocol is presented, a modification on the KE protocol primitive building specifically KDF primitive is made into leakage resilient KDF primitive.

A. Comparison of KDF Security Models

Table. 5 Comparison of KDF security models

Security Model	**CPM** [10]	**CPM-R**
Type of adversary	Active	Active
KDF inputs not allowed to query by adversary:	Private seed	Private seed
KDF inputs allowed to query by adversary:	Salt, Context Information	Salt, Context Information
Side channel information given to adversary:	None	Execution time of KDF for computing a cryptographic output
Relationship between models:	1. *Lemma 1*: CPM-R ⇒ CPM, if a KDF is not CPM secure, then it is not CPM-R secure. 2. *Corollary 1*: $CPM\text{-}R \nrightarrow CPM$, if a KDF is CPM-R secure, it may not be CPM secure under timing analysis side channel leakage.	

Table 5 summarises the adversary power of existing KDF security models and the proposed CPM-R model. Both security model allow active adversary to query for the KDF parameter inputs. Both security models does not allow adversary to choose for the private seed input to the KDF, but both also allow the adversary to choose for public salt values and public context information values. The distinct difference between both security models is where the proposed CPM-R allow adversary to query for side channel information such as execution time of a KDF to compute a cryptographic output. This side channel information given to the adversary relates to actual real world side channel timing analysis attack in which exploit the pattern analysis derived from the execution time leakage by a KDF during the protocol execution. Based on the relationships result between CPM-R and CPM model, through lemma 1 and corollary 1, proposed CPM-R model is a stronger model compare to existing CPM model.

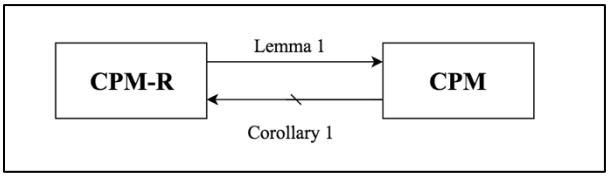

Fig. 1 The relationship between CPM-R with CPM

Figure 1 shows the result of proven relationships between the proposed CPM-R security model and existing CPM security model. From Lemma 1, the result shows if a KDF is proven not to be secure in CPM model, then the particular KDF is not secure in CPM-R security model. From corollary 1, the result shows that if a KDF is proven to be secure in CPM-R but it might not be necessary secure in CPM security model under timing analysis SCA. This concludes that CPM-R is a stronger security model which covers timing analysis attack compare to existing CPM model.

B. Comparison of KE Protocol

Table. 6 Comparison of key exchange protocol

KE protocol	**Protocol π**[11]	**Modified Protocol π**
Protocol construction	Generic	Generic
Primitives assumption	CPLA2-Secure Public Key Cryptosystems, UFCMLA-Secure Signature Schemes, DDH assumption, Key Derivation Function	CPLA2-Secure Public Key Cryptosystems, UFCMLA-Secure Signature Schemes, DDH assumption, **Leakage Resilient Key Derivation Function**
Side channel attack resilient	Cold-Boot Attack, Malware Attacks	Cold-Boot Attack, Malware Attacks, **Timing Analysis Attack**

Table 6 shows the difference between key exchange protocol π[11] and the proposed modified protocol π. Both protocols are generic construction that can be instantiated with suitable cryptographic primitives. The distinct difference between both protocol is that the modified protocol π instantiated use timing analysis leakage resilient KDF based on (Section II -Definition 2.0.3). In other words, the modified protocol π now addresses an extra side channel attack which is timing analysis attack. Noted that our proposed leakage resilient KDF on (Section II -Definition 2.0.3) is proven secure in our proposed CPM-R security model through (Section IV -Theorem 2). The leakage resilient KDF is provably secure even under timing analysis SCA where the execution time for the KDF to compute the cryptographic key is randomized thereby no useful information can be learnt by the adversary. The suggestion of implementing the leakage resilient KDF during the protocol building for protocol π[11] can results the KE protocol to be secure under timing analysis SCA as whole because the cryptographic primitives instantiated are leakage resilience.

VII. CONCLUSION

In this paper, a leakage resilient KDF is proposed to resist timing analysis SCA. The proposed idea is to mask the actual execution time for the KDF to compute a cryptographic key so adversary deduce no useful information to break the protocol scheme. This work also proposed a KDF security model namely CPM-R to prove the security claim of leakage resilient KDF under timing analysis attack. The proposed leakage resilient KDF is suggested to be implement into protocol π by Alawatugoda to achieve leakage resilient key exchange protocol as whole. A security proof is shown to prove the modified protocol π is leakage resilient.

ACKNOWLEDGMENT

The authors would like to express the sincerest gratitude and appreciation for all helpful feedbacks and comments given throughout the research works. This work is supported and funded by the Tier H082, Research Management Centre (RMC), of Universiti Tun Hussein Onn Malaysia (UTHM).

REFERENCES

1. J. A. Halderman et al., "Lest We Remember: Cold Boot Attacks on Encryption Keys," in Proceedings of USENIX Security, 2008.
2. P. Kocher, J. Jaffe, B. Jun, and P. Rohatgi, "Introduction to differential power analysis," J. Cryptogr. Eng., vol. 1, no. 1, pp. 5–27, 2011.
3. J. Longo, E. De Mulder, D. Page, and M. Tunst All, "SoC it to EM: Electromagnetic side-channel attacks on a complex system-on-chip," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2015, vol. 9293, pp. 620–640.
4. D. Boneh, R. A. Demillo, and R. J. Lipton, "On the importance of checking cryptographic protocols for faults," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 1997, vol. 1233, pp. 37–51.
5. P. C. Kocher, "Cryptanalysis of Diffie-Hellman, RSA, DSS, and other systems using timing attacks (Extended Abstract)," in Advances in Cryptology CRYPTO95 15th Annual International Cryptology Conference, 1995.
6. D. Brumley and D. Boneh, "Remote timing attacks are practical," Comput. Networks, vol. 48, no. 5, pp. 701–716, 2005.
7. C. W. Chuah and W. W. Koh, "Timing side channel attack on key derivation functions," in Lecture Notes in Electrical Engineering, 2017, vol. 424, pp. 266–273.
8. J. Alawatugoda, D. Jayasinghe, and R. Ragel, "Countermeasures against Bernstein's remote cache timing attack," in 2011 6th International Conference on Industrial and Information Systems, ICIIS 2011 - Conference Proceedings, 2011.
9. D. Bernstein, "Cache-timing attacks on AES," Compute, 2005.
10. R. Chen, Y. Mu, G. Yang, W. Susilo, and F. Guo, "Strongly leakage-resilient authenticated key exchange," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016, vol. 9610, pp. 19–36.
11. J. Alawatugoda, "On the leakage-resilient key exchange," Journal of Mathematical Cryptology, vol. 11, no. 4. pp. 215–269, 2017.
12. J. Di Wu, Y. M. Tseng, and S. S. Huang, "Efficient Leakage-Resilient Authenticated Key Agreement Protocol in the Continual Leakage eCK Model," IEEE Access, vol. 6, pp. 17130–17142, 2018.
13. D. Moriyama and T. Okamoto, "An eCK-Secure Authenticated Key Exchange Protocol without Random Oracles," in ProvSec 2009: Provable Security, 2009, pp. 154–167.
14. C. C. Wen, E. Dawson, J. M. González Nieto, and L. Simpson, "A framework for security analysis of key derivation functions," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2012, vol. 7232 LNCS, pp. 199–216.
15. H. Krawczyk, "Cryptographic extraction and key derivation: The HKDF scheme," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2010, vol. 6223 LNCS, pp. 631–648.
16. H. Krawczyk, "On Extract-then-Expand Key Derivation Functions and an HMAC-based KDF," Draft available http//www. ee. Tech. ac. il/hugo/, pp. 1–37, 2008.
17. F. F. Yao and Y. L. Yin, "Design and analysis of password-based key derivation functions," IEEE Trans. Inf. Theory, 2005.
18. W. Diffie, W. Diffie, and M. E. Hellman, "New Directions in Cryptography," IEEE Trans. Inf. Theory, vol. 22, no. 6, pp. 644–654, 1976.