

A Generic Construction for Intrusion-Resilient Signatures from Linear Feedback Shift Register*

ZHENG GONG, XIANGXUE LI[†], DONG ZHENG AND KEFEI CHEN

Department of Computer Science and Engineering

[†]School of Information Security Engineering

Shanghai Jiaotong University

Shanghai 200030, P.R. China

E-mail: {neoyan; kfchen}@sjtu.edu.cn

[†]State Key Laboratory of Information Security

Graduate School of Chinese Academy of Sciences

Beijing 100039, P.R. China

With the development of various cryptographic primitives deployed on insecure devices, key exposure seems inevitable. Generalized from forward-secure signatures and key-insulated signatures, intrusion-resilient signatures (IRs) was first introduced by Itkis and Reyzin, which help us to minimize the damage of key exposure. IRs provide the unforgeability for the past and future time periods unless both the signer and the home base modules were compromised simultaneously (even in this worst case, IRs preserve the unforgeability for the past periods). In this paper, we propose a practical generic construction for IRs. By applying our proposal, one can simply transform any signature scheme to a practical intrusion-resilient signature scheme. In particular, we present a concrete paradigm from linear feedback shift register (LFSR). Our LFSR-based paradigm's base and signer secret keys can be reduced to just a half length of the regular ones, which is useful to save the communication and storage costs. Moreover, if the underlying signature is unforgeable in the standard model, then our construction is also intrusion-resilient in the standard model without any extra assumptions.

Keywords: digital signature, key exposure, forward security, intrusion-resilient, linear feedback shift register

1. INTRODUCTION

For all cryptographic applications, thanks to jumbled management or trojan program, key exposure seems unavoidable in practice. An adversary can easily obtain a secret key without breaking the underlying cryptographic assumption on which the security of the system is based. Forward-secure signature, which was introduced by Anderson [1] and formalized by Bellare and Miner [2], has an attractive property that the adversary can not forge any signature of all past time periods even if he compromised the signer's key at a certain time period. Due to this appealing property, forward-secure cryptosystems were intensively studied. Instructive examples can be found in [3-8].

However, forward-secure systems can not protect the future periods if while the adversary compromises the current period's secret. To eliminate this limitation, Dodis *et al.* [9] proposed a key-insulated (KI for short) model where two independent modules are

Received November 7, 2006; revised April 10, 2007; accepted March 20, 2008.

Communicated by Wen-Guey Tzeng.

* This work was supported by NSFC (No. 60573030, 60703030, 60703031) and the National Laboratory for Modern Communications Science Foundation of China (No. 51436040405JW0304).

used to protect the secret: a signer who has the secret signing key to generate the signatures, and a home base that updates the signer's secret key. KI model enables not only forward security, but also the unforgeability for the future even if the signer module is compromised. (Notice that the base module still needs uncompromised.)

Still, the security of the past time periods is not preserved in the KI model if the base and the signer modules were compromised simultaneously. This limitation was fixed by Itkis and Reyzin who proposed the intrusion-resilient (IR for short) model [10]. Intrusion-resilient model does not only preserve the security of past and future time periods when both the signer and the base are not compromised simultaneously, but also the security of past time periods in the case of simultaneous compromise. Itkis and Reyzin [10] also figured out one tremendous advance, that belongs to the intrusion-resilient solution, is a great forward to *obsolesce the certificate revocation* in public key infrastructures, since certificates would have to be revoked only in the unlikely case that the signer and the base modules are compromised simultaneously.

Our Contributions: Motivated by [3], we first propose a generic construction that transforms any signature (even with special properties, *e.g.*, blind signature, undeniable signature, *etc.*) to practical intrusion-resilient signature (IRS for short). Based on linear feedback shift register (LFSR for short) and its sequence operations, we give a concrete paradigm from LFSR. Compares with the previous generic construction [12], our LFSR-based concrete paradigm's base and signer secret keys can be reduced to just a half length of the regular ones, which is useful to save the communication and storage costs in applications. Moreover, if the underlying signature is unforgeable in the standard model, then our scheme is also intrusion-resilient in the standard model without any extra assumptions.

Organization: The remainder of the paper is organized as follows. After reviewed some preliminaries in section 2. Section 3 describes our generic construction for the intrusion-resilient signature scheme, and then we analyze the security and performance of the construction in section 4. Section 5 gives a concrete paradigm from LFSR. Section 6 concludes the paper.

2. PRELIMINARIES

In this section, some definitions for LFSR will be briefly explained. LFSR is a widely-used primitive for the construction of pseudorandom number generators and public key cryptosystems [14-16]. Let q be a prime power, we denote an irreducible polynomial $f(x)$ over $GF(q)$, such that

$$f(x) = x^n - a_1x^{n-1} + a_2x^{n-2} - \dots + (-1)^n a_n, \quad (1)$$

where all $a_1, \dots, a_n \in GF(q)$. Let γ be a root of $f(x)$ in the extension $GF(q^n)$. The sequence s denotes a LFSR sequence of order n over $GF(q)$ that generated by the following recurrence.

$$s_{k+n} = a_1s_{k+n-1} + a_2s_{k+n-2} + \dots + (-1)^{n+1}a_ns_k. \quad (2)$$

Let $\bar{s}_i = (s_i, s_{i+1}, \dots, s_{i+n-1})$, for $i = 0, 1, \dots, n - 1$. Our initial state of \bar{s}_i is given by $s_i = Tr(\gamma^i)$, where $Tr(\cdot)$ is the trace map from $GF(q^n)$ to $GF(q)$. Thus the period P of the sequence s_i is equal to the order of the root γ .

In LFSR sequences, there is a sequence operation that will be used in our scheme, which can be fast performed from the theory of LFSR [17].

Definition 1 *Sequence Operation (SO)*: Given A_k and an random integer l , where $0 \leq k, l < P$, compute A_{kl} .

In LFSR sequences, there are two main computational hard assumptions.

Definition 2 *LFSR-Based Discrete Logarithm Problem (LFSR-DLP)*: Given state A_1 and A_l , where $1 < l < P$, to determine l .

Definition 3 *LFSR-Based Diffie-Hellman Problem (LFSR-DHP)*: Given state A_1 along with A_k and A_l , where $0 \leq k, l < P$, to determine A_{kl} .

It was proven that LFSR-DLP and LFSR-DHP are computational equivalent to the DLP and DHP, respectively, in the group of prime order [16].

Short Representation: LFSR sequences are closely related to the minimal polynomials of finite field elements due to the Newton Identity [16]. Let the set $A_k = (s_k, s_{2k}, \dots, s_{rk})$. Observe from the Newton Identity and the proof given by [16], one just needs a *half length representation* than the finite field case (for $n \cdot \log q$ bits to q^n), the length of A_k can be defined by

$$|A_k| = \begin{cases} \frac{n}{2} \cdot \log q, & \text{if } q = p^2 \text{ and } n \text{ is even,} \\ \frac{n-1}{2} \cdot \log q, & \text{if } q = p^2 \text{ and } n \text{ is odd.} \end{cases} \quad (3)$$

For more details about LFSR sequences and its short representation, see [16-19].

3. INTRUSION-RESILIENT SIGNATURE FROM ANY SIGNATURE SCHEME

In this section, a generic construction for the IRSs is described. First we review the functional definition and the security model for IRSs.

3.1 Functional Definition

Here the notion of the components of the intrusion-resilient model is recalled [10]. $SK_{t,r}$ denotes the secret key, where t is the time period and r is the refresh number.

Definition 4 An intrusion-resilient signature scheme *IR-SIG* is a tuple of polynomial bound algorithms (*IR-GEN*, *IR-SIGN*, *IR-VER*, *UB*, *US*, *RB*, *RS*):

1. *IR-GEN* (The key generation algorithm):
Input: secure parameters, the total number T of time periods.
Output: initial signer key $SKS_{0,0}$, initial base key $SKB_{0,0}$, and public key PK .
2. *IR-SIGN* (The signing algorithm):
Input: current signer key $SKS_{t,r}$ and message msg .
Output: signature (t, S) on msg for time period t .
3. *IR-VER* (The verifying algorithm):
Input: message msg , signature (t, S) and public key PK .
Output: “valid” or “invalid”.
4. *UB* (The base key update algorithm):
Input: current base key $SKB_{t,r}$.
Output: new base key $SKB_{(t+1),0}$ and key update message SKU_t .
5. *US* (The signer key update algorithm):
Input: current signer secret key $SKS_{t,r}$ and key update message SKU_t .
Output: new signer secret key $SKS_{(t+1),0}$.
6. *RB* (The base key refresh algorithm):
Input: current base key $SKB_{t,r}$.
Output: new base key $SKB_{t,(r+1)}$ and corresponding key refresh message $SKR_{t,r}$.
7. *RS* (The signer refresh algorithm):
Input: current signer key $SKS_{t,r}$ and key refresh message $SKR_{t,r}$.
Output: new signer key $SKS_{t,(r+1)}$ (corresponding to base key $SKB_{t,(r+1)}$).

We stress that one can still achieve the intrusion-resilient security by lifting some above restrictions. The functional definition of the KI signature scheme [9] allows the use of *UB*, but restricting SKB_t does not change for every period t .

3.2 Security Model

In order to formalize security, we recall the security model for the intrusion-resilient signature schemes [12]. Let T be the total period of the system. Let r denote the current number of times the keys are refreshed in the period $t \in \{1, T\}$. Maximally, there will be $RN(t) + 1$ instances of the signer and base keys, where $RN(\cdot)$ is a method only used for notational convenience. Recall that each update is immediately followed by a refresh. Thus, keys with refresh index 0 are never actually used. The keys will be generated by the following algorithm.

Experiment *Generate-Key*(k, T, RN)

$$\begin{aligned}
& t \leftarrow 0; r \leftarrow 0; \\
& (SKS_{t,r}, SKB_{t,r}, PK) \leftarrow IR-Gen(1^k, T); \\
& \text{for } t = 1 \text{ to } T \\
& \quad (SKB_{t,0}, SKU_{t-1}) \leftarrow UB(SK B_{t-1,r}); \\
& \quad SKS_{t,0} \leftarrow US(SK S_{t-1,r}, SKU_{t-1}); \\
& \quad \text{for } r = 1 \text{ to } RN(t) \\
& \quad \quad (SKB_{t,r}, SKR_{t,r-1}) \leftarrow RB(SK B_{t,r-1}); \\
& \quad \quad SKS_{t,r} \leftarrow RS(SK S_{t,r-1}, SKR_{t,r-1}).
\end{aligned} \tag{4}$$

Let SKS^* , SKB^* , SKU^* , SKR^* be the sets consisting of the signer and base keys and update and refresh messages, respectively. Generated during the above algorithm. The sets contain all the secrets that can be directly stolen by the adversary. To formally define the security, an adaptive adversary can access two oracles as follows.

- $OSig$, the signing oracle (constructed by using SKS^*), which on input (m, t, r) , output $IR-SIGN(SKS_{t,r}^*, m)$.
- $OSec$, the key exposure oracle, which
 1. on input (“s”, t, r) for $1 \leq t \leq T, 1 \leq r \leq RN(t)$ outputs $SKS_{t,r}^*$;
 2. on input (“b”, t, r) for $1 \leq t \leq T, 1 \leq r \leq RN(t)$ outputs $SKB_{t,r}^*$;
 3. on input (“u”, t) for $1 \leq t \leq T - 1$ outputs SKU_t^* and $SKR_{t+1,0}^*$;
 4. on input (“r”, t, r) for $1 \leq t \leq T, 1 \leq r \leq RN(t)$ outputs $SKR_{t,r}^*$.

For any set of valid key exposure queries Q , time period $t \geq 1$ and refresh number $r, 1 \leq r \leq RN(t)$, we say that the key $SKS_{t,r}^*$ is Q -exposed:

- if (“s”, t, r) $\in Q$; or
- if $r > 1, (“r”, t, r - 1) \in Q$, and $SKS_{t,r-1}^*$ is Q -exposed; or
- if $r = 1, (“u”, t - 1) \in Q$, and $SKS_{t-1,RN(t-1)}^*$ is Q -exposed.

Replacing SKS^* with SKB^* throughout the above definition yields the definition of base key exposure. Both definitions are recursive, with direct exposure as the base case. Thus, we say that the scheme is (t, Q) -compromised, if either

- $SKS_{t,r}^*$ is Q -exposed for some $r, 1 \leq r \leq RN(t)$; or
- $SKS_{t',r}^*$ and $SKB_{t',r}^*$ are both Q -exposed for some $t' < t$.

The following experiment captures adversary’s functionality. Intuitively, the adversary succeeds if he generates a valid signature without “cheating”: not obtaining this signature from $OSig$, asking only legal queries (e.g., no out of bounds queries), and not compromising the scheme for the given time period. We call this adversary (fully) adaptive because he is allowed to decide which keys and signatures to query based on all of the previous answers.

Experiment $IR-CMA(\mathcal{A}, k, T, RN)$

```

Generate-Keys( $k, T, RN$ );
( $m, j, sig$ )  $\leftarrow \mathcal{A}^{OSig, OSec}(1^k, T, PK, RN)$ ;
if ( $IR-VER(m, j, sig) = \text{“invalid”}$  or
    ( $m, j$ ) was queried by  $\mathcal{A}$  to  $OSig$  or
    there was an illegal query or
    the scheme is  $(j, Q)$ -compromised)
    then return 0;
else return 1.
    
```

(5)

Definition 5 Let $IR[k, T, RN]$ be a signature scheme with security parameter k , number of time period T , and table RN of T refresh numbers. For forger F , the advantage that adversary \mathcal{A} compromises the scheme is

$$Adv_{\mathcal{A}}^{IR}(F, IR[k, T, RN]) = Pr[IR-CMA(F, k, T, RN) = 1].$$

We say $IR[k, T, RN]$ is intrusion-resilient, if \mathcal{A} runs in polynomial time τ and asks at most q_s oracle queries, where ε is a negligible probability under security parameter k .

$$Adv_{\mathcal{A}}^{IR}(F, IR[k, T, RN]) < \varepsilon.$$

In an intrusion-resilient signature scheme, the secrets are divided into two modules: the base and the signer. The later is used to sign arbitrary messages. If an adversary only compromises the signer secret key, he can not forge signatures in the past time periods before the signer secret key was obtained. If an adversary only compromises the signer module, he can only forge signatures in the current time period while the signer secret were obtained. If the adversary can compromise the base module, he still can not forge any signature in any time period. In the worst situation that both the base and the signer modules are compromised simultaneously, the scheme still remains forward-secure.

3.3 Our Generic Construction

In this section, we give a generic construction that transforms any unforgeable signature scheme $SIG = (KG, SIGN, VER)$ into an intrusion-resilient signature scheme $IR-SIG = (IR-GEN, IR-SIGN, IR-VER, IR-UPD)$. The notion of unforgeability that we use for the underlying signature scheme is the strong notion of security for digital signature as formalized in [20]. Since the base secret key and the signer secret key will be changed at every beginning of the period, we combine the update and refresh algorithms by the algorithm $IR-UPD$, and ignore the refreshment times value r for the convenience. First we describe some functions that will be used in our construction. For each period $t = 1, 2, \dots, T$, the current signer key pair (SKS_t, PKS_t) denotes the secret-public keys for SIG , (SKB_t, PKB_t) denotes the current base key pair. Algorithm $IR-SKG(\cdot, \cdot)$ denotes an intrusion-resilient secret key generator, which updates current secret keys SKB_t, SKS_t to SKB_{t+1}, SKS_{t+1} . For simplicity, $IR-SKG$ implements both the update and refresh algorithms that defined in Definition 4, where the refresh times are omitted in our construction). The key generator can be easily built from any forward-secure pseudorandom number generator [3]. The algorithm $PKG(\cdot, \cdot)$ computes the public keys from the corresponding secret keys. A detailed description of our scheme is given below:

IR-GEN: During the initialization, the system parameters are created for the scheme. We assume the scheme totally contains T periods of time, and period 1 is the beginning period of *IR-SIG*.

1. Generate two secure random numbers SKB_0 and SKS_0 .
2. $(PKB_0, PKS_0) \leftarrow PKG(SKB_0, SKS_0)$.
3. For $t = 1, 2, \dots, T$, do
 - (a) $(SKB_t, SKS_t) \leftarrow IR-SKG(SKB_{t-1}, SKS_{t-1})$.
 - (b) $(PKB_t, PKS_t) \leftarrow PKG(SKB_t, SKS_t)$.
 - (c) $CERT_t \leftarrow (t, PKS_0, PKS_t, SIGN_{SKS_0}(t, PKS_0, PKS_t))$.
4. Erase all secret-public keys except SKB_1 and SKS_1 . Store SKB_1 and SKS_1 as the initial base and signer secret key, respectively. Store $CERT_t, t = 1, 2, \dots, T$ and publish PKS_0 .

IR-SIGN: Given the current period signer secret key SKS_t , assuming a user asks a signature on an arbitrary message msg , then:

1. Reload current period $CERT_t$.
2. Sign msg with SKS_t , output a signature $S = (CERT_t, SIGN_{SKS_t}(msg))$.

IR-VER: Given a signature pair (PKS_0, S) , the verification is proceeded as follows:

1. Get $CERT_t$ and $SIGN_{SKS_t}(msg)$ from S .
2. Check if the initial signer public key in $CERT_t$ equals PKS_0 , then verify $SIGN_{SKS_0}(t, PKS_0, PKS_t)$ with PKS_0 . If all succeed then parse $CERT_t$ to get the current period signer public key PKS_t .
3. Verify $SIGN_{SKS_t}(msg)$ with the current signer public key PKS_t . If success output valid, otherwise output invalid.

IR-UPD: At the beginning of each period, do the following steps to evolve the keys.

1. $(SKB_{t+1}, SKS_{t+1}) \leftarrow IR-SKG(SKB_t, SKS_t)$.
2. $(PKB_{t+1}, PKS_{t+1}) \leftarrow PKG(SKB_{t+1}, SKS_{t+1})$.
3. Reload $CERT_{t+1}$ and verify it with PKS_0 , and check if the signer public key PKS_{t+1} equals the key that generated in previous step. If the above procedure succeeds, then output valid, otherwise output invalid.

From the generic construction, we stress that the secret key generator function $IR-SKG(\cdot, \cdot)$ must obtain forward-security. In the following section, the security of the scheme will be formally analyzed.

4. ANALYSIS OF THE GENERIC CONSTRUCTION

4.1 Intrusion-Resilient Security

Let $SIG = (KG, SIGN, VER)$ be a regular signature scheme under the strong notion of security for digital signature scheme [20], and $IR-SKG$ is a pseudorandom number generator with forward security under the notion formalized in [21]. With the above primitives, the following theorem summarizes our result and straightforward to prove.

Theorem 1 $IR-SIG = (IR-GEN, IR-SIGN, IR-VER, IR-UPD)$ based on underlying regular signature scheme SIG and $IR-SKG$ is an unforgeable signature scheme with intrusion-resilient security.

Proof: Considering different kinds of intrusions, we analyze the fully security under three types of the compromise modes in intrusion-resilient model.

Type-1. First we assume a successful forger \mathcal{F}_1 against $IR-SIG$ with non-negligible probability ε_1 in polynomial time t_1 , but he only can compromise the signer module.

One can use \mathcal{F}_1 to build an adversary \mathcal{A}_1 against underlying signature scheme SIG as follows. Since \mathcal{F}_1 compromised the signer secret key SKS_t of time period t , so he can forge the signatures in current time period t , but without the current base secret key SKB_t , because $IR-SKG$ is forward secure, \mathcal{F}_1 can not get any information about $SKS_{t'}$, $SKB_{t'}$ that $t' \neq t$, if \mathcal{F}_1 can either forge a valid intrusion-resilient signature S' in the past or the future, then \mathcal{A}_1 runs \mathcal{F}_1 in polynomial time t_1 , he can get a valid signature pair (t', S') , under the probability ε_1 . Notice that S' is a valid signature of SIG . With the help of \mathcal{F}_1 , \mathcal{A}_1 can forge a valid signature of SIG at least with probability ε_1 in polynomial time t_1 . this conflicts with the underlying signature scheme SIG is unforgeable under the strong security notion for digital signatures.

From the above we can see a forger only compromised signer module can not forge any signature in the past and the future, which means $IR-SIG$ satisfies forward-secure model [2].

Type-2. Considering the full attack in the KI model [9], we assume a successful forger \mathcal{F}_2 against $IR-SIG$ with non-negligible probability ε_2 in polynomial time t_2 , and \mathcal{F}_2 has the signer and the base modules in different periods, denotes by SKS_t and $SKB_{t'}$, $t \neq t'$.

One can use \mathcal{F}_2 to build an adversary \mathcal{A}_2 against underlying signature scheme SIG as follows. Since \mathcal{F}_2 compromised the signer key SKS_t of time period t , so he can forge the signatures in time period t , but without the current base key SKB_t , as $(SKB_{t+1}, SKS_{t+1}) \leftarrow IR-SKG(SK B_t, SKS_t)$, \mathcal{F}_2 can not get any information about both the keys in the past and the future, if \mathcal{F}_2 can derive any secret of the key, it means we can use \mathcal{F}_2 reverse the one way function $IR-SKG$. If \mathcal{F}_2 can either forge a valid intrusion-resilient signature S' in the time period t' , $t' \neq t$, then \mathcal{A}_2 runs \mathcal{F}_2 in polynomial time t_2 , he can get a valid signature pair (t', S') , with the probability ε_2 . Notice that S' is a valid signature of SIG . With the help of \mathcal{F}_2 , \mathcal{A}_2 can forge a valid signature of SIG with ε_2 probability in polynomial time t_2 , this makes contradiction to the predefinition that the underlying signature scheme SIG is unforgeable under the strong security notion for digital signature.

From the above we can see a forger, who compromised the signer and the base modules but unsimultaneously, can not forge any signature in the past and the future, which means $IR-SIG$ satisfies the KI security model [9].

Type-3. With the worst consideration, we assume a powerful forger in intrusion-resilient model [10] that the forger compromised both the base and signer simultaneously, e.g., obtained the key pair SKS_t, SKB_t . we define \mathcal{F}_3 against $IR-SIG$ with non-negligible probability ε_3 in polynomial time t_3 .

One can use \mathcal{F}_3 to build an adversary \mathcal{A}_3 against underlying signature scheme SIG as follows. Because \mathcal{F}_3 compromised SKS_t, SKB_t , so \mathcal{F}_3 can easily forge the signatures in the current time period t , and follow the algorithm $(SKB_{t+1}, SKS_{t+1}) \leftarrow IR-SKG(SK B_t, SKS_t)$, he can derive the next period keys then the future signatures are not safe now. but $IR-SKG$ is forward secure, \mathcal{F}_3 can not get any information about $SKS_{t'}$, $SKB_{t'}$ that $t' < t$ since it faced one way assumption of $IR-SKG$, if \mathcal{F}_3 can forge a valid intrusion-resilient signature S' in the past period t' , $t' < t$, then \mathcal{A}_3 runs \mathcal{F}_3 in polynomial time t_3 , he can get a valid signature pair (t', S') , under the probability ε_3 . Notice that S' is a valid signature of

SIG. With the help of \mathcal{F}_3 , \mathcal{A}_3 can forge a valid signature of *SIG* with ε_3 probability in polynomial time t_3 , this derives to conflict with the premise that the underlying signature scheme *SIG* is unforgeable under the strong security notion for digital signature.

From the above we can see a forger compromised both the base and signer modules simultaneously still can not forge the signature in the past, which means *IR-SIG* is forward secure under this powerful intrusion and satisfies the IR security model given in previous section 3.2.

Summarize the above proofs, we stress that the generic construction satisfies all the conditions to obtain the intrusion-resilient security, and the security is tightly reduced to the existential unforgeability of the underlying signature, without any extra assumptions. \square

4.2 Advantages

Compares with previous intrusion-resilient signature schemes, such as [10, 12], *etc.*, our generic construction has a number of advantages.

Simplicity. Based on some cryptography primitives, our generic construction is very simple and practical. The construction has no complex computation or algorithm, but just an practical way to realize the intrusion-resilient property, which consists of theoretical and practical advantages for a wide environments.

Adaptability. In the generic construction, the transformation can be adapted to variants of signature schemes, *e.g.*, RSA, ElGamal, *etc.* Furthermore, the transformation can provide this useful property not only for the regular signature schemes, but also the schemes with special properties (*e.g.* blind signature, undeniable signature, *etc.*).

Efficiency. The largest computational costs of our scheme are generating the period certificates, but this work can be done outside of the construction beforehand. The computation of signing algorithm in the construction equals to the underlying signature scheme. Verification cost equals to twice operations of the underlying signature verification (one for the verification of the certificate, one for the signature).

5. A CONCRETE INTRUSION-RESILIENT SIGNATURE SCHEME

In this section, we will propose a concrete intrusion-resilient signature scheme based on LFSR in detail, and show the short representation advantage of the concrete paradigm that why we choose LFSR.

5.1 Intrusion-Resilient Secret Key Generator from LFSR

First we construct an intrusion-resilient secret key generator from LFSR. In our construction, LFSR shows great advantages to satisfy the security requirements. Let *LFSR-SKG*(\cdot, \cdot) be a secret key generator function, which follows the definitions mentioned in section 2.2. Let γ be a root of $f(x)$ in the extension $GF(q^n)$. The sequence is generated by a LFSR of order n over $GF(q)$, which is defined by the Eq. (3) in the section 2.2. We denote the set $A_k = (s_k, s_{2k}, \dots, s_{rk})$. The algorithm of the function is described below.

LFSR-SKG Algorithm

1. First, generate a LFSR sequence s , randomly select two different sets A_{k_0}, A_{l_0} from the sequence. Let the base secret key $SKB_0 = A_{k_0}$, the signer secret key $SKS_0 = A_{l_0}$.
2. Take SKS_0 and SKB_0 as two initial values, which will be used to compute the secret keys for the next period. For $t = 1, \dots, T$ do
 - (a) Extract $s_{k_{t-1}}, s_{l_{t-1}}$ from $A_{k_{t-1}}(SKB_{t-1}), A_{l_{t-1}}(SKS_{t-1})$,
 - (b) Compute $A_{l_t}(SKS_t)$ from $s_{k_{t-1}}$ and $A_{l_{t-1}}(SKS_{t-1})$ using **SO**, then erase $A_{l_{t-1}}$,
 - (c) Compute $A_{k_t}(SKB_t)$ from $s_{l_{t-1}}$ and $A_{k_{t-1}}(SKB_{t-1})$ using **SO**, then erase $A_{k_{t-1}}$.

The above steps can be formally described as

$$(SKB_t, SKS_t) \leftarrow LFSR-SKG(SKB_{t-1}, SKS_{t-1}). \quad (6)$$

Since we can set the length of the LFSR element s_k , it is convenient that use *LFSR-SKG* to generate any signature's secret parameters. For example, choose 160 bits length for ElGamal signature, 1,024 bits for RSA signature and so on. Because of the less computational costs and the representation of finite field elements by LFSR [16], the algorithm's computation and bandwidth can all directly reduced. Intuitively, the construction is not limited to LFSR but also can be alternated by any other forward-secure secret key generator function.

5.2 Concrete Paradigm

Based on *LFSR-SKG*, we give an intrusion-resilient signature paradigm from DSS [11]. Since we choose the previous *LFSR-SKG* as the *IR-SKG* function, it can be simply extended to any other signature still keeps the intrusion-resilient property. For brevity, we omit the details of DSS [11] here. Let the signature scheme $DSS-SIG = (DSS-KG, DSS-SIGN, DSS-VER)$, $DSS-PKG(\cdot)$ is the public key generator, which inputs the signer secret key x and outputs $y = g^x$ as the public key. With the similar definitions, the concrete paradigm is described below.

Initialization: Generate a LFSR sequence s , $|s_i| = 160$ bit, randomly select two different sets A_{k_0}, A_{l_0} from the sequence. Let the base secret key $SKB_0 = A_{k_0}$, the signer secret key $SKS_0 = A_{l_0}$. Take SKB_0 and SKS_0 as initial values, we will use them to compute the next period secret keys. For $t = 1, \dots, T$, do

1. $(SKB_t, SKS_t) \leftarrow LFSR-SKG(SKB_{t-1}, SKS_{t-1})$. Notice that SKS_t is a valid secret key for DSS.
2. $PKS_{t-1} \leftarrow DSS-PKG(SKS_{t-1})$.
3. $CERT_t \leftarrow (t, PKS_0, PKS_t, DSS-SIGN_{SKS_0}(t, PKS_0, PKS_t))$.

Erase all secret-public keys except SKB_1 and SKS_1 . Store SKB_1 and SKS_1 as the initial base key and signer secret key, respectively. Store $CERT_t$, $t = 1, \dots, T$ and publish PKS_0 .

Signature Generation: With the current signer key SKS_t , assume a user asks a signature on an arbitrary message msg , then:

1. Reload current period $CERT_t$.
2. Sign the message msg with SKS_t under DSS, then output the signature $S = (CERT_t, DSS-SIGN_{SKS_t}(msg))$.

Verification: Given a signature pair (PKS_0, S) , the verification proceeds:

1. Get $CERT_t$ and $DSS-SIGN_{SKS_t}(msg)$ from S .
2. Check if the initial signer public key in $CERT_t$ equals PKS_0 , then verify $DSS-SIGN_{SKS_t}(t, PKS_0, PKS_t)$ with PKS_0 under algorithm $DSS-VER$. If all succeed then Parse $CERT_t$ to get the current period signer public key PKS_t , otherwise output invalid and abort.
3. Verify $DSS-SIGN_{SKS_t}(msg)$ with PKS_t under $DSS-VER$. If success output valid, otherwise output invalid.

Key-Update: At the beginning of each period, do the following steps to evolve the keys.

1. $(SKB_{t+1}, SKS_{t+1}) \leftarrow LFSR-SKG(SKB_t, SKS_t)$.
2. $PKS_{t+1} \leftarrow DSS-PKG(SKS_{t+1})$.
3. Reload $CERT_{t+1}$ and verify it use $DSS-VER$ under PKS_0 ; and check if the signer public key PKS_{t+1} in it equals the key that generated in previous step. If all of these checks succeed output valid, otherwise output invalid.

Implementation Issue: Follow our generic construction, we simply transform DSS signature as the underlying signature to the intrusion-resilient signature. Since we can simply set the length of the random number k and LFSR element s_i , the transformation can be adapted to variants of signature schemes, such as RSA, ElGamal, *etc.* Furthermore, the transformation can provide this useful property not only to regular signature schemes, but also the schemes with special capabilities (*e.g.* blind signature, undeniable signature). Due to the short representation property on LFSR set elementary which is described in section 2, the scheme's base secret key A_k and the signer secret key A_i can be reduced to just a half length of the regular ones without losing any security concerns. The security and computation comparisons between short representation LFSR-based system and regular finite-field system can be found in a technical report [22]. This short representation is useful to save the communication and storage costs in applications.

6. CONCLUSION

In this paper, we proposed a generic construction that can simply transform any signature (even with special properties, *e.g.*, blind signature, undeniable signature, *etc.*) into practical intrusion-resilient signature. In particular, we presented a transformation paradigm from LFSR. The intrusion-resilient property of the scheme is easily achieved and directly supported by the unforgeability of the underlying signature scheme without any extra assumptions. The advantages show that the generic construction is practical and convenient. Future work is to find some other cryptosystems which will also enjoy the advantages of LFSR.

REFERENCES

1. R. Anderson, "Two remarks on public-key cryptology," Invited Lecture, in *Proceedings of the 4th Annual Conference on Computer and Communications Security*, 1997.
2. M. Bellare and S. Miner, "A forward-secure digital signature scheme," in *Proceedings of Advances in Cryptology – CRYPTO*, LNCS 1666, 1999, pp. 431-438.
3. H. Krawczyk, "Simple forward-secure signatures from any signature scheme," in *Proceedings of the 7th ACM Conference on Computer and Communications Security*, 2000, pp. 108-115.
4. G. Itkis and L. Reyzin, "Forward-secure signatures with optimal signing and verifying," in *Proceedings of Advances in Cryptology – CRYPTO*, LNCS 2139, 2001, pp. 332-354.
5. A. Kozlov and L. Reyzin, "Forward-secure signatures with fast key update," in *Proceedings of the 3rd International Conference on Security in Communication Networks*, LNCS 2576, 2002, pp. 241-256.
6. T. Malkin, D. Micciancio, and S. Miner, "Efficient generic forward-secure signatures with an unbounded number of time periods," in *Proceedings of Advances in Cryptology – EUROCRYPT*, LNCS 2332, 2002, pp. 400-417.
7. E. Cronin, "On the performance, feasibility, and use of forward-secure signatures," in *Proceedings of the 10th ACM Conference on Computer and Communications Security*, 2003, pp. 131-144.
8. J. Camenisch and M. Kopolowski, "Fine-grained forward-secure signature schemes without random oracle," *Discrete Applied Mathematics*, Vol. 154, 2006, pp. 175-188.
9. Y. Dodis, J. Katz, S. Xu, and M. Yung, "Key-insulated public key cryptosystems," in *Proceedings of Advances in Cryptology – EUROCRYPT*, LNCS 2332, 2002, pp. 130-135.
10. G. Itkis and L. Reyzin, "SiBIR: signer-base intrusion-resilient signatures," in *Proceedings of Advances in Cryptology – CRYPTO*, LNCS 2442, 2002, pp. 449-514.
11. NIST, *Digital Signature Standard*, Federal Information Processing Standards Publication 186, U.S. Department of Commerce/N.I.S.T., 1994.
12. G. Itkis, "Intrusion-resilient signatures: generic constructions, or defeating strong adversary with minimal assumptions," in *Proceedings of the 3rd Conference on Security in Communication Networks*, LNCS 2576, 2003, pp. 102-118.
13. R. Cramer and V. Shoup, "Signature schemes based on the strong RSA assumption," *ACM Transactions on Information and Systems Security*, Vol. 3, 2000, pp. 161-185.
14. G. Gong and L. Harn, "Public-key cryptosystems based on cubic finite field extensions," *IEEE Transactions on Information Theory*, Vol. 45, 1999, pp. 2601-2605.
15. A. Lenstra and E. Verheul, "The XTR public key system," in *Proceedings of Advances in Cryptology – CRYPTO*, LNCS 1880, 2000, pp. 1-19.
16. K. J. Giulian and G. Gong, "New LFSR-based cryptosystems and the trace discrete log problem (trace-DLP)," in *Proceedings of International Conference on Sequences and Their Applications*, LNCS 3486, 2004, pp. 298-312.
17. S. Golomb, *Shift Register Sequences*, Laguna Hills, CA, Aegean Park, 1982.
18. H. Niederreiter, "Finite fields and cryptology," *Finite Fields, Coding Theory, and Advances in Communications and Computing*, 1993, pp. 359-373.
19. C. Tan, X. Yi, and C. Siew. "On the n th order shift register based discrete logarithm,"

- IEICE Transactions on Fundamentals*, Vol. E86-A, 2003, pp. 1213-1216.
20. S. Goldwasser, S. Micali, and R. L. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks," *SIAM Journal on Computing*, Vol. 17, 1988, pp. 281-308.
 21. M. Bellare and B. Yee, "Forward-security in private-key cryptography," in *Proceedings of the Cryptographers' Track at the RSA Conference*, LNCS 2612, 2003, pp. 1-18.
 22. S. Sin, *GH-PKC Software Implementation*, Technical Report, Waterloo University, <http://comsec.uwaterloo.ca/projects.html#gh>.



Zheng Gong (龚征) received the B.S. degree in Computer Science and Technology in Nan Chang University, China, and received the M.S. degree in Computer Science and Technology in South China University of Technology, China. Now he is a Doctor Candidate in Shanghai Jiaotong University, China. His recent research directions are cryptography and provable security.



Xiangxue Li (李祥学) received the Ph.D. degree in Cryptography from Shanghai JiaoTong University in 2006. He is now with the School of Information Security Engineering, Shanghai Jiaotong University. His areas of research include provable security, pseudorandom sequence, coding theory, secure distributed storage and new cryptographic technology. His research is funded by NSFC, 863 and other open funds.



Dong Zheng (郑东) received the Ph.D. degree in Cryptography from Xidian University in 1999. He is a Professor of the School of Information Security Engineering, Shanghai Jiaotong University. From 2002 to 2007, he was with the Department of Computer Science and Engineering, Shanghai Jiaotong University. His areas of research include provable security and new cryptographic technology, especially, coding theory and its application in secure distributed storage. His research is funded by NSFC, 863 and private companies.



Kefei Chen (陈克非) was born in 1959. He received his Ph.D. degree in Justus Liebig University Giessen, Germany, 1994. His main research areas are classical and modern cryptography, theory and technology of network security, *etc.* Since 1996, he came to Shanghai Jiaotong University and became the Professor at the Department of Computer Science and Engineering. Up to now (1996-2007), he has published more than 80 academic papers on cryptology.