Journal of Computing and Security

July 2018, Volume 5, Number 2 (pp. 59–65)

http://www.jcomsec.org

Security Analysis of Two Lightweight Certificateless Signature Schemes

Nasrollah Pakniat [a,*]

[a] *Information Science Research Center, Iranian Research Institute for Information Science and Technology (IRANDOC), Tehran, Iran.*

ARTICLE INFO.

Article history:
Received: 20 May 2018
Revised: 25 February 2019
Accepted: 13 March 2019
Published Online: 27 November 2018

Keywords:
Certificateless Cryptography, Cryptanalysis, Signature, Industrial Internet of Things, Healthcare Wireless Sensor Networks.

ABSTRACT

Certificateless cryptography can be considered as an intermediate solution to overcome the issues in traditional public key infrastructure (PKI) and identity-based public key cryptography (ID-PKC). There are a vast number of certificateless signature (CLS) schemes in the literature; however, most of them are not efficient enough to be utilized in limited resources environments such as Internet of Things (IoT) or Healthcare Wireless Sensor Networks (HWSN). Recently, two lightweight CLS schemes have been proposed by Karati et al. and Kumar et al. to be employed in IoT and HWSNs, respectively. While both schemes are claimed to be existentially unforgeable, in this paper, we show that both these signatures can easily be forged. More specifically, it is shown that 1) in Karati et al.'s scheme, a type 1 adversary, considered in certificateless cryptography, can generate a valid partial private key corresponding to any user of its choice and as a consequence, it can forge any users' signature on any message of its choice, and 2) in Kumar et al.'s scheme, both types of adversaries which are considered in certificateless cryptography are able to forge any signer's signature on an arbitrary message.

1 Introduction

Certificateless cryptography, introduced in 2003 by Al-riyami and Paterson [1], can be considered as an intermediate solution to overcome the issues in traditional public key infrastructure (PKI) and identity-based public key cryptography (ID-PKC) [2]. Whereas a trusted authority is needed in traditional PKI to bind the identity of an entity to his public key, ID-PKC requires a trusted private key generator to generate the private keys of users based on their identities. Therefore, the certificate management problem in the public-key setting is actually replaced by the key escrow problem. In certificateless cryptography, the users' private keys are still generated with the help of a third party, called the key generation center (KGC). However, the KGC doesn't have access to the final private keys generated by the users themselves (based on some private information obtained from the KGC and some secret values chosen by the users). The public key of a user is computed from the KGC's public parameters and some information, private to the user, and is published by the user himself.

Regarding the security of a certificateless cryptographic scheme, two types of adversaries are considered in the literature: a Type 1 adversary A_1 who simulates malicious ordinary users and a Type 2 Adversary A_2 who simulates a malicious KGC in a cer-

[*] Corresponding author.

Email address: pakniat@irandoc.ac.ir

https://dx.doi.org/10.22108/jcs.2019.110889
ISSN: 2322-4460 © 2018 JComSec. All rights reserved.

tificateless cryptographic scheme. To perform these simulations, A_1 is allowed to replace the public key of entities with other values of its choice and A_2 is allowed to get access to the master secret key.

The first certificateless signature (CLS) scheme was proposed in [1] by Al-Riyami and Paterson. After this seminal work, a vast number of certificateless signature schemes were proposed such as ordinary CLS schemes [1, 3–10], certificateless proxy signature schemes [11–14], certificateless aggregate signature schemes [4, 15–20], certificateless signature schemes with designated tester [21, 22], certificateless threshold signature schemes [23–25], certificateless ring signature schemes [26, 27], and etc. However, due to their heavy computational costs, most of these schemes can not be applied in limited resources environments such as Internet of Things (IoT) and Healthcare Wireless Sensor Networks (HWSN). As a consequence, new efforts have been put forth to construct lightweight cryptographic schemes in certificateless setting in order to be applicable in limited resources environments. In this regard, recently, two lightweight certificateless signature schemes have been proposed by Karati et al. [3] and Kumar et al. [4]. The authors of both papers claimed that their proposed CLS schemes are existentially unforgeable. However, in this paper, we disprove their claims and show that the CLS schemes of [3] and [4] are both insecure. This is done by showing that:

- In **Karati et al.'s CLS scheme**, a type 1 adversary of certificateless cryptography is able to generate a valid partial private key corresponding to any identity of its choice and then uses this generated partial private key to forge the signature of the corresponding user on any message of its choice.
- In **Kumar et al.'s CLS scheme**, both types of adversaries, considered in certificateless cryptography, are able to violate the unforgeability of the scheme. More precisely, 1) a type 1 adversary is able to forge any signer's signature on any message in this scheme as soon as it gets access to a pair of message and its corresponding signature of that signer, and 2) a type 2 adversary is able to forge each user's signature on any message in this scheme (without even requiring to see a signature of that signer).

The rest of this paper is organized as follows. In Section 2, we provide the framework and the security definition of CLS schemes. In Section 3, after reviewing the CLS scheme of [3], we provide the proof of its insecurity. Then, the CLS scheme of Kumar et al. [4] and analysis of its security are reviewed in Section 4. Finally, the conclusions are provided in Section 5.

2 Certificateless Signature Schemes

In this section, we provide the framework and the security definition of Certificateless signature schemes.

2.1 The Framework

There exist three entities in a CLS scheme: a key generation center (KGC) which helps users to generate their private keys, a signer, and a verifier. A CLS scheme consists of six algorithms: Setup, Set-Partial-Private-Key, Set-Secret-Value, Set-Public-Key, CLS-Sign and CLS-Verify. The details of these algorithms are described in the following:

Setup: Performed by KGC.

- Input: The security parameter k.
- Process:
  - Generates the master secret key MSK, and the public parameters $params$.
- Output: The master secret key MSK which will be secured by KGC and the public parameters $params$ which are published.

Set-Partial-Private-Key: Performed by KGC.

- Input: $params$, MSK and a user's identity ID_S.
- Process:
  - Computes a partial private key D_S corresponding to this user.
- Output: Partial private key D_S which will be sent securely to the user with identity ID_S.

Set-Private-Key: Performed by a user S.

- Input: $params$ and S's partial private key D_S.
- Process:
  - Generates a secret value x_S and computes the private key SK_S by using it and D_S.
- Output: SK_S which will be secured by the user S.

Set-Public-Key: Performed by a user S.

- Input: $params$ and S's private key SK_S.
- Process:
  - Computes the public key PK_S.
- Output: PK_S which will be published.

CLS-Sign: Performed by the user S.

- Input: $params$, the user's identity ID_S and his private key SK_S, and a message m.
- Process:
  - Generates a signature σ on the message m.
- Output: σ as the signature on m.

CLS-Verify: Performed by the verifier.

- Input: $params$, signer's identity ID_S and his public key PK_S, message m and a signature σ.
- Process:
  - Checks the validity of σ.

- Output: VALID if σ is a valid signature on m and INVALID otherwise.

2.2 Security Model

To call a CLS scheme secure, it should provide existentially unforgeability against adaptive chosen-message and -identity attacks in the adversarial model of certificateless cryptography which consists of the following two types of adversaries:

- A type-1 adversary (A_1), that has not access to the master secret key but can replace any signer's public key with any value of its choice.
- A type-2 Adversary (A_2), that has access to the master secret key but cannot replace public keys.

The security of a CLS scheme is modeled through the following two games played between a challenger C and adversaries A_1 or A_2.

Game 1: This game, played between C and A_1, consists of the following phases:

- Setup: In this phase, C generates the master secret key MSK and the public parameters $params$. It keeps MSK secure and sends $params$ to A_1.
- Queries: In this phase, A_1 can perform a polynomially bounded number of the following queries and C's answers to these queries are as follows:
  - ○ Request-Partial-Private-Key (ID_S): inputting ID_S to this query, A_1 will get S's partial private key D_S as the output.
  - ○ Request-Secret-Value (ID_S): inputting ID_S to this query, A_1 will get S's secret value x_S as the output.
  - ○ Request-Public-Key (ID_S): inputting ID_S to this query, A_1 will get S's public key PK_S as the output.
  - ○ Replace-Public-Key (ID_S, PK'_S): inputting ID_S and PK'_S to this query, PK'_S will be set as the public key corresponding to the user S.
  - ○ CL-Sign (ID_S, m): inputting ID_S and m to this query, A_1 will get σ as the output which is a valid signature of S on m.
- Output: Finally, when A_1 decides to end the queries phase, it outputs a signature σ on a message m on behalf of a targeted user with identity ID. It wins the game if the following conditions are fulfilled:
  - ○ The algorithm CLS-Verify outputs VALID on inputs $params$, m, σ, ID, and PK where, PK is the public key corresponding to the user with identity ID.
  - ○ The queries Request-Partial-Private-Key(ID) and CL-Sign(ID, m) weren't queried in the queries phase.

Definition 1. A CLS scheme is Type-1 secure against the adaptively chosen-message and -identity attack if the advantage of any polynomially bounded adversary A_1 in winning Game 1 be negligible.

Game 2: This game, played between C and A_2, consists of the following phases:

- Setup: In this phase, C generates the master secret key MSK and the public parameters $params$ and sends them to A_2.
- Queries: In this phase, A_2 can perform a polynomially bounded number of queries as in Game 1 and C answers them in the same way. The only constraint here is that A_2 is not allowed to replace any public keys. Note that A_2 knows MSK and can compute the partial private key of any identity by itself.
- Output: Finally, when A_2 decides to end the queries phase, it outputs a signature σ on a message m on behalf of a targeted user with identity ID. It wins the game if the following conditions are fulfilled:
  - ○ The algorithm CLS-Verify outputs VALID on inputs $params$, m, σ, ID, and PK where, PK is the public key corresponding to the user with identity ID.
  - ○ The queries Request-Secret-Value(ID) and CL-Sign(ID, m) weren't queried in the queries phase.

Definition 2. A CLS scheme is Type-2 secure against the adaptively chosen-message and -identity attack if the advantage of any polynomially bounded adversary A_2 in winning Game 2 be negligible.

3 Karati et al.'s CLS Scheme

In this section, we first review Karati et al.'s CLS scheme and then prove that it is completely insecure.

3.1 Review of the Scheme

The CLS scheme of Karati et al. [3] consists of the following algorithms:

Setup: Performed by KGC.

- Input: The security parameter k.
- Process:
  - ○ Generates two groups G_1 and G_2 with the same prime order p and an efficient bilinear pairing $e : G_1 \times G_1 \rightarrow G_2$.
  - ○ Chooses a generator $g_1 \in G_1$.
  - ○ Chooses a cryptographic hash function $H : \{0,\ 1\}^* \rightarrow Z_p^*$.
  - ○ Chooses a random $y \in Z_p^*$ as his master secret

key.

  - ○ Computes $g_2 = e(g_1, g_1)^y$ and $Y_{KGC} = g_1^y$.
- Output: The master secret key y which will be secured by KGC and the public parameters $params = (G_1,\ G_2,\ p,\ e,\ g_1,\ g_2,\ Y_{KGC},\ H)$ which will be published.

Set-Partial-Private-Key: Performed by KGC.

- Input: $params$, master secret key y and a user's identity $ID_i \in \{0,\ 1\}^*$.
- Process:
  - ○ Computes $h_i = H(ID_i)$.
  - ○ Chooses $r_i \in Z_p^*$ randomly and computes $R_i = g_1^{r_i}$ and $y_i = (g_1)^{\frac{y \cdot h_i}{h_i + r_i + y}}$.
- Output: Partial private key $D_i = (y_i, R_i)$ which will be sent securely to the user with identity ID_i. After receiving D_i from KGC, the user considers D_i genuine if:
$$e(g_1, Y_{KGC})^{h_i} = e(y_i, (g_1^{h_i} \cdot R_i \cdot Y_{KGC}))\quad(1)$$

Set-Private-Key: Performed by a user i.

- Input: $params$ and i's partial private key $D_i = (y_i, R_i)$.
- Process:
  - ○ Chooses $x_i, c_i \in Z_p^*$ randomly and sets $SK_i = (c_i, x_i, R_i)$.
- Output: SK_i which will be secured by the user i.

Set-Public-Key: Performed by a user i.

- Input: $params$, i's partial private key $D_i = (y_i, R_i)$ and his private key $SK_i = (c_i, x_i, R_i)$.
- Process:
  - ○ Computes $Y_i = \left(Y_{i1} = (y_i)^{\frac{1}{x_i}}, Y_{i2} = g_2^{c_i}\right)$ as the user's public key.
- Output: Y_i which will be published.

CLS-Sign: Performed by a user S.

- Input: $params$, the user's identity ID_S and his private key $SK_S = (c_S, x_S, R_S)$ and a message m.
- Process:
  - ○ Computes $h_S = H(ID_S)$.
  - ○ Chooses a random value $t \in Z_p^*$ and computes
$$\sigma_1 = g_2^t, \quad (2)$$
$$\sigma_2 = \left(g_1^{h_S} \cdot R_S \cdot Y_{KGC}\right)^{\left(\frac{c_S}{m} - t\right)x_S}. \quad (3)$$
- Output: $\sigma = (\sigma_1, \sigma_2)$ as the signature on m.

CLS-Verify: Performed by the verifier.

- Input: $params$, S's identity ID_S and his public key $Y_S = (Y_{S1}, Y_{S2})$, message m and a signature $\sigma = (\sigma_1, \sigma_2)$.
- Process:
  - ○ Computes $h_S = H(ID_S)$.
  - ○ Checks whether $\left(\frac{Y_{S2}^{\frac{1}{m}}}{\sigma_1}\right)^{h_S} \overset{?}{=} e\left(Y_{S1}, \sigma_2\right)$.
- Output: VALID if the above equation holds and INVALID otherwise.

3.2 Cryptanalysis of the Scheme

The authors of [3] claimed that their proposed scheme is a secure certificateless signature scheme. However, in this section, we disprove their claim. More specifically, we show that by accessing to a valid partial private key corresponding to any user, a valid partial private key corresponding to any other user can be generated. Thereupon, each user of this scheme can forge the signature of other users on any arbitrary message of his choice. This is formally stated and proved in the following theorem.

Theorem 1. *Let S with identity ID_S be an arbitrary user of Karati et al.'s scheme. Suppose that A_1 has access to a valid partial private key corresponding to S. Then, A_1 is able to generate a valid partial private key corresponding to any other user S' with arbitrary identity $ID_{S'}$ and as a consequence, he is able to forge S''s signature on any message of his choice.*

Proof. According to Set-Partial-Private-Key algorithm of Karati et al.'s CLS scheme, the partial private key corresponding to S with identity ID_S is a pair (y_S, R_S) where, $R_S = g_1^{r_S}$ and $y_i = (g_1)^{\frac{y \cdot h_S}{h_S + r_S + y}}$ in which $r_S \in Z_p^*$ is an unknown randomly chosen value, y is the master secret key and $h_S = H(ID_S)$. In the following, we show how A_1 is able to use S's partial private key to generate a valid partial private key corresponding to any other user S' with arbitrary identity $ID_{S'}$. To this end, A_1:

(1) Computes $h_{S'} = H(ID_{S'}) \in Z_p^*$.

(2) Computes $\alpha = \frac{h_{S'}}{h_S} \in Z_p^*$. Note that the output of $H(\cdot)$ is a member of Z_p^* and therefore, h_S^{-1} exists in Z_p^*.

(3) Computes $y_{S'} = y_S^{\alpha}$ and $R_{S'} = \frac{R_S}{g_1^{(\alpha - 1)\cdot h_S}}$.

(4) Sets $(y_{S'}, R_{S'})$ as the partial private key corresponding to the user S' with identity $ID_{S'}$.

Using the following relation, it can easily be verified that $(y_{S'}, R_{S'})$ is a valid partial private key corresponding to the user S' with identity $ID_{S'}$:

$$
\begin{aligned}
&e(y_{S'}, (g_1^{h_{S'}} \cdot R_{S'} \cdot Y_{KGC})) && (4)\\
&= e(y_S^{\alpha}, (g_1^{\alpha\cdot h_S} \cdot \frac{R_S}{g_1^{(\alpha-1)\cdot h_S}} \cdot Y_{KGC})) && (5)\\
&= e(y_S^{\alpha}, (g_1^{\alpha\cdot h_S-(\alpha-1)h_S} \cdot R_S \cdot Y_{KGC})) && (6)\\
&= e(y_S^{\alpha}, (g_1^{h_S} \cdot R_S \cdot Y_{KGC})) && (7)\\
&= e(y_S, (g_1^{h_S} \cdot R_S \cdot Y_{KGC}))^{\alpha} && (8)\\
&= e(g_1, Y_{KGC})^{h_S\cdot\alpha} && (9)\\
&= e(g_1, Y_{KGC})^{h_{S'}}, && (10)
\end{aligned}
$$

where, equality (9) is obtained from the fact that (y_S, R_S) is a valid partial private key generated by the KGC and therefore,

$$e(g_1, Y_{KGC}) = e(y_S, (g_1^{h_S} \cdot R_S \cdot Y_{KGC})). \tag{11}$$

After computing S''s partial private key, A_1 can perform Set-Private-Key and Set-Public-key (as explained in Katari et al.'s CLS scheme) instead of S' to compute a valid pair of private and public keys corresponding to S'. Now, using the private key of S', A_1 can forge S''s signature through CLS-Sign algorithm on any message of its choice. □

4 Kumar et al.'s CLS Scheme

In this section, we first review Kumar et al.'s CLS scheme and then prove that their scheme is forgeable.

4.1 Review of the Scheme

The CLS scheme of Kumar et al. [4] consists of the following algorithms:

Setup: Performed by KGC.

- Input: The security parameter k.
- Process:
  - ○ Chooses two groups G_1 and G_2 with the same prime order q and a generator P in G_1.
  - ○ Chooses a bilinear map $e : G_1 \times G_1 \rightarrow G_2$.
  - ○ Chooses a random $\alpha \in Z_q^*$ as the master secret key and sets $P_{Pub} = \alpha \cdot P$.
  - ○ Chooses cryptographic hash functions $H_1, H_2 : \{0,\ 1\}^* \rightarrow G_1$ and $H_3 : \{0,\ 1\}^* \rightarrow Z_q^*$.
- Output: The master secret key α which will be secured by KGC and the system parameters $params = (q,\ G_1,\ G_2,\ e,\ P,\ P_{Pub},\ H_1,\ H_2,\ H_3)$ which will be published.

Set-Partial-Private-Key: Performed by KGC.

- Input: $params$, master secret key α and a user's identity $ID_i \in \{0,\ 1\}^*$.
- Process:
  - ○ Computes $Q_{ID_i} = H_1(ID_i)$.
  - ○ Computes $D_i = \alpha \cdot Q_{ID_i}$.
- Output: Partial private key D_i which will be sent securely to the user with identity ID_i.

Set-Private-Key: Performed by a user i.

- Input: $params$ and i's identity ID_i.
- Process:
  - ○ Selects a random value $x_i \in Z_q^*$ as the i's secret key.
  - ○ Sets $SK_i = (x_i, D_i)$.
- Output: SK_i which will be secured by the user i.

Set-Public-Key: Performed by a user i.

- Input: $params$ and i's private key $SK_i = (x_i, D_i)$.
- Process:
  - ○ Computes $Y_i = x_i \cdot P$ as i's public key.
- Output: Y_i which will be published.

CLS-Sign: Performed by a user S.

- Input: $params$, the signer's identity ID_S, his public key Y_S, his private key $SK_S = (x_S, D_S)$, some state information Δ and a message m.
- Process:
  - ○ Chooses a random value $r \in Z_q^*$ and computes $R = r \cdot P \in G_1$.
  - ○ Computes $W = H_2(\Delta)$ and $h = H_3(m,\ ID_S,\ Y_S,\ R)$.
  - ○ Computes $V = D_S + r \cdot W + h \cdot x_S \cdot P_{Pub}$.
- Output: $\sigma = (R,\ V)$ as the signature on m under the state information Δ.

CLS-Verify: Performed by the verifier.

- Input: $params$, signer's identity ID_S and his public key Y_S, message m, some state information Δ and a signature $\sigma = (R,\ V)$.
- Process:
  - ○ Computes $Q_{ID_S} = H_1(ID_S)$, $W = H_2(\Delta)$ and $h = H_3(m,\ ID_S,\ Y_S,\ R)$.
  - ○ Verifies $e(V,\ P) \stackrel{?}{=} e(Q_{ID_S} + h \cdot Y_S, P_{Pub})e(R, W)$.
- Output: VALID if the above equation holds and INVALID otherwise.

4.2 Cryptanalysis of the Scheme

Kumar et al. claimed that their scheme is existentially unforgeable against adaptive chosen message attacks. However, in this section, we disprove their claim. We prove the insecurity of Kumar et al.'s CLS scheme by the following theorems:

Theorem 2. *Let S be a signer with identity ID_S who uses Kumar et al.'s CLS scheme. Suppose that a type 1 adversary A_1 has access to a tuple $(m,\ \sigma = (R, V)\ , \Delta)$, where σ is S's signature on message m under the state information Δ. Then, A_1 is able to forge S's signature on any new message m' under the same state information Δ.*

Proof. According to Kumar et al.'s CLS-Sign algorithm, the signature σ is as follows:

$$R = r \cdot P, \qquad V = D_S + r \cdot H_2(\Delta) + x_S \cdot h \cdot P_{Pub}, \tag{12}$$

where $h = H_3(m,\ ID_S,\ Y_S,\ R)$ and $r \in Z_q^*$ is a random value that is unknown to A_1. Now, in order to forge S's signature on a new massage m', A_1:

(1) Issues a Request-Secret-Value query on the input of ID_S and obtains x_S as the result.
(2) Computes $D_{S,\Delta} = V - x_S \cdot h \cdot P_{Pub} = D_S + r \cdot H_2(\Delta)$.
(3) Uses $D_{S,\Delta}$, x_S and R to forge S's signature on m' as follows:
    1. Computes $h' = H_2(m',\ ID_S,\ Y_S,\ R)$ and $V' = D_{S,\Delta} + h' \cdot x_S \cdot P_{Pub}$.
    2. Outputs $\sigma' = (R,\ V')$ as S's signature on message m'.

    It can be easily verified that the forged signature σ' is valid.

□

Theorem 3. *Let S be a signer with identity ID_S who uses Kumar et al.'s CLS scheme. Then, a type 2 adversary A_2 is able to forge S's signature on any message m of its choice under any arbitrary state information Δ.*

Proof. To forge S's signature on any arbitrary message m, A_2:

1. Chooses a random value $r \in Z_q^*$ and computes $R = r \cdot P$.
2. Computes $h = H_3(m,\ ID_S,\ Y_S,\ R)$ and $V = D_S + rH_2(\Delta) + h \cdot \alpha \cdot Y_S$.
3. Outputs $\sigma = (R,\ V)$ as S's signature on message m'.

Note that A_2 acts as the malicious key generation center and has access to partial private keys. It can be easily verified that the forged signature σ is valid. □

5 Conclusion

In this paper, the security of two recently proposed lightweight certificateless signature schemes is considered. We prove that in one of them, a type 1 adversary of certificateless cryptography can forge the signature of any user on any arbitrary message of his choice and in the other one, both considered types of adversaries in certificateless cryptography can forge valid signatures on behalf of any user on any message of their choices.

References

[1] Sattam S. Al-Riyami and Kenneth G. Paterson. Certificateless Public Key Cryptography. In Chi-Sung Laih, editor, *Advances in Cryptology - ASIACRYPT 2003*, pages 452–473, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg. doi:10.1007/978-3-540-40061-5_29.

[2] Adi Shamir. Identity-Based Cryptosystems and Signature Schemes. In George Robert Blakley and David Chaum, editors, *Advances in Cryptology*, pages 47–53, Berlin, Heidelberg, 1985. Springer Berlin Heidelberg. doi:10.1007/3-540-39568-7_5.

[3] A. Karati, S. H. Islam, and M. Karuppiah. Provably Secure and Lightweight Certificateless Signature Scheme for IIoT Environments. *IEEE Transactions on Industrial Informatics*, PP(99): in press, 2018. doi:10.1109/TII.2018.2794991.

[4] Pankaj Kumar, Saru Kumari, Vishnu Sharma, Arun Kumar Sangaiah, Jianghong Wei, and Xiong Li. A certificateless aggregate signature scheme for healthcare wireless sensor network. *Sustainable Computing: Informatics and Systems,*, page in press, 2017. ISSN 2210-5379. doi:10.1016/j.suscom.2017.09.002.

[5] Liaojun Pang, Yufei Hu, Yi Liu, Kedong Xu, and Huixian Li. Efficient and secure certificateless signature scheme in the standard model. *International Journal of Communication Systems*, 30 (5):e3041–n/a, 2017. doi:10.1002/dac.3041.

[6] Liangliang Wang, Kefei Chen, Yu Long, and Huige Wang. An efficient pairing-free certificateless signature scheme for resource-limited systems. *Science China Information Sciences*, 60 (11):119102, Dec 2016. doi:10.1007/s11432-015-0367-6.

[7] Yumin Yuan and Chenhui Wang. Certificateless signature scheme with security enhanced in the standard model. *Information Processing Letters*, 114(9):492 – 499, 2014. doi:10.1016/j.ipl.2014.04.004.

[8] Jianhong Zhang and Jane Mao. An efficient RSA-based certificateless signature scheme. *Journal of Systems and Software*, 85(3):638 – 642, 2012. doi:10.1016/j.jss.2011.09.036.

[9] Xinyi Huang, Yi Mu, Willy Susilo, Duncan S. Wong, and Wei Wu. Certificateless Signature Revisited. In *Information Security and Privacy*, pages 308–322, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg. doi:10.1007/978-3-540-73458-1_23.

[10] N. Pakniat and B. A. Vanda. Cryptanalysis and Improvement of a Pairing-Free Certificateless Signature Scheme. In *2018 15th International ISC (Iranian Society of Cryptology) Conference on Information Security and Cryptology (ISCISC)*, pages 1–5, Aug 2018. doi:10.1109/ISCISC.2018.8546984.

[11] Yang Lu and Jiguo Li. Provably secure certificateless proxy signature scheme in the standard model. *Theoretical Computer Science*, 639:42 – 59, 2016. doi:10.1016/j.tcs.2016.05.019.

[12] Ziba Eslami and Nasrollah Pakniat. A certificateless proxy signature scheme secure in standard model. In *International Conference on Latest Computational Technologies-ICLCT 2012*, pages 81–84, Planetary Scientific Research Center: Bangkok, 2012.

[13] Seung-Hyun Seo, Kyu Young Choi, Jung Yeon Hwang, and Seungjoo Kim. Efficient certificateless proxy signature scheme with provable security. *Information Sciences*, 188:322 – 337, 2012. doi:10.1016/j.ins.2011.11.005.

[14] C. Hu and D. Li. A New Type of Proxy Ring Signature Scheme with Revocable Anonymity. In *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007)*, volume 1, pages 866–868, 2007.

[15] Lin Cheng, Qiaoyan Wen, Zhengping Jin, Hua Zhang, and Liming Zhou. Cryptanalysis and improvement of a certificateless aggregate signature scheme. *Information Sciences*, 295:337 – 346, 2015. doi:10.1016/j.ins.2014.09.065.

[16] Yu-Chi Chen, Raylin Tso, Masahiro Mambo, Kaibin Huang, and Gwoboa Horng. Certificateless aggregate signature with efficient verification. *Security and Communication Networks*, 8 (13):2232–2243, 2015. doi:10.1002/sec.1166.

[17] Shi-Jinn Horng, Shiang-Feng Tzeng, Po-Hsian Huang, Xian Wang, Tianrui Li, and Muhammad Khurram Khan. An efficient certificateless aggregate signature with conditional privacy-preserving for vehicular sensor networks. *Information Sciences*, 317:48 – 66, 2015. doi:10.1016/j.ins.2015.04.033.

[18] Hu Xiong, Zhi Guan, Zhong Chen, and Fagen Li. An efficient certificateless aggregate signature with constant pairing computations. *Information Sciences*, 219:225 – 235, 2013. doi:10.1016/j.ins.2012.07.004.

[19] Ziba Eslami and Nasrollah Pakniat. Certificateless aggregate signcryption: Security model and a concrete construction secure in the random oracle model. *Journal of King Saud University - Computer and Information Sciences*, 26(3):276 – 286, 2014. doi:10.1016/j.jksuci.2014.03.006.

[20] N. Pakniat and M. Noroozi. Cryptanalysis of a certificateless aggregate signature scheme. In *the 9th Conference of Command, Control, Communications and Computer Intelligence*, pages 1–5, 2016.

[21] Yang Chen, Yang Zhao, Hu Xiong, and Feng Yue. A Certificateless Strong Designated Verifier Signature Scheme with Non-delegatability. *International Journal of Network Security*, 19(4):573–582, 2017. doi:10.6633/IJNS.201707.19(4).10.

[22] Xinyi Huang, W. Susilo, Yi Mu, and Futai Zhang. Certificateless Designated Verifier Signature Schemes. In *20th International Conference on Advanced Information Networking and Applications - Volume 1 (AINA'06)*, volume 2, pages 15–19, 2006.

[23] Hong Yuan, Futai Zhang, Xinyi Huang, Yi Mu, Willy Susilo, and Lei Zhang. Certificateless threshold signature scheme from bilinear maps. *Information Sciences*, 180(23):4714 – 4728, 2010. doi:10.1016/j.ins.2010.07.021.

[24] Licheng Wang, Zhenfu Cao, Xiangxue Li, and Haifeng Qian. Simulatability and security of certificateless threshold signatures. *Information Sciences*, 177(6):1382 – 1394, 2007. doi:10.1016/j.ins.2006.08.008.

[25] Licheng Wang, Zhenfu Cao, Xiangxue Li, and Haifeng Qian. Certificateless Threshold Signature Schemes. In Yue Hao, Jiming Liu, Yu-Ping Wang, Yiu-ming Cheung, Hujun Yin, Licheng Jiao, Jianfeng Ma, and Yong-Chang Jiao, editors, *Computational Intelligence and Security*, pages 104–109, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. doi:10.1007/11596981_15.

[26] Deng, Lunzhi. Certificateless ring signature based on rsa problem and dl problem. *RAIRO-Theor. Inf. Appl.*, 49(4):307–318, 2015. doi:10.1051/ita/2016013.

[27] Lijun Zhu and Futai Zhang. An efficient certificateless ring signature scheme. *Wuhan University Journal of Natural Sciences*, 13(5):567, 2008. doi:10.1007/s11859-008-0511-8.

Nasrollah Pakniat has a PhD in Mathematics, graduated in 2015 from Shahid Beheshti University. He received his MSc degree in Computer Science from Shahid Beheshti University in 2011. He holds a BSc degree in Computer Science from Shahid Bahonar University of Kerman in 2008. He began his scientific experience in 2016 as a faculty member in Iranian Research Institute for Information Science and Technology (IranDoc). His research interests include cryptography, network security and text mining.