# Analysis and Construction of Efficient RFID Authentication Protocol with Backward Privacy

Shaohui Wang[1,2,3,*], Sujuan Liu[1,2], Danwei Chen[1,2],

[1] College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210046, China
[2] Jiangsu High Technology Research Key Laboratory for Wireless Sensor Networks, Nanjing, Jiangsu 210003, China
[3] Network and Data Security Key Laboratory of Sichuan Province
wangshaohui@njupt.edu.cn

**Abstract.** Privacy of RFID systems is receiving increasing attentions in the RFID community and an important issue required as to the security of RFID system. Backward privacy means the adversary can not trace the tag later even if he reveals the internal states of the tag sometimes before. In this paper, we analyze two recently proposed RFID authentication schemes: Randomized GPS and Randomized Hashed GPS scheme. We show both of them can not provide backward privacy in Juels and Weis privacy model, which allows the adversary to know whether the reader authenticates the tag successfully or not. In addition, we present a new protocol, called Challenge-Hiding GPS, based on the Schnorr identification scheme. The challenge is hidden from the eavesdropping through the technique of Diffie-Hellman key agreement protocol. The new protocol can satisfy backward privacy, and it has less communication overheads and almost the same computation, compared with the two schemes analyzed.

**Keywords:** RFID; Elliptic Curve Cryptography (ECC); Mutual Authentication; Backward Privacy

## 1 Introduction

As Radio Frequency Identification (RFID) systems are becoming more common (for example in access control, product tracking, e-ticketing, electronic passports), managing the associated privacy and security concerns becomes more important. Since RFID tags are primarily used for authentication purposes, 'security' in this context means that it should be infeasible to fake a legitimate tag, and 'Privacy', on

[*] Corresponding author

the other hand, means that adversaries should not be able to identify, trace, or link tag appearances.

To settle the security and privacy problems, several authentication protocols were presented. Feldhofer et. al. [1] proposed a challenge-response authentication protocol based on AES algorithm; HB+[2] protocol, a very efficient protocol presented by Juels and Weis, is based on the well known LPN problem, but it can not resist man-in-the-middle attack[3], and the subsequent modifications[4,5] all can not resist this attack[6,7]. To measure the privacy level of various RFID protocols, several models for privacy preserving RFID authentication systems have already been proposed, such as Juels and Weis [8](JW model), Burmenster, van Le and de Medeiros [9] and Vaudenay [10]. In the JW model, the adversary has the ability to corrupt the tag and retrieve the internal secrets, and he also knows the authentication result of the reader. Backward privacy, proposed in [11], means that if the adversary reveals the internal state of a tag at some time $t$, the adversary is not able to tell whether a transaction after time $t+\tau$ (for some $\tau > 0$) involves the tag, provided that the adversary does not eavesdrop on the tag continuously after time $t$.

Usually it is believed public-key cryptography is too slow, complex and power-hungry for RFID. However, recent publications on compact and efficient Elliptic Curve Cryptography (ECC) implementations challenge this assumption [12, 13]. One of the first ECC based authentication protocols is the EC-RAC protocol[14] that has been proposed to address tracking attacks. However, it is shown that EC-RAC is vulnerable to various man-in-the-middle and replay attacks[15-17]. As a result, the EC-RAC protocol has been gradually revised in [18] to tackle the known attacks. In [19], Bringer et.al. have a research on the identification scheme with privacy requirement. They propose a framework which enables to transform some generic ZK scheme into private scheme and they apply as a relevant example this framework to the GPS scheme[20] to propose two efficient schemes(Randomized GPS and Randomized Hashed GPS).

In this paper, we give an analysis of Randomized GPS and Randomized Hashed GPS, and show they can not provide backward privacy in the JW privacy model. Besides we propose an efficient ECC-based authentication protocol for RFID system named Challenge-Hiding GPS scheme, and the scheme can provide backward privacy. In addition, compared with the two schemes in [19], our scheme has less communication overhead and almost the same computation.

The paper is organized as follows: Section 2 gives some preliminaries and recalls the JW privacy model; we present the analysis of Randomized GPS and Randomized Hashed GPS schemes as to a JW adversary in section 3; The Challenge-Hiding GPS scheme which satisfy backward privacy is presented in section 4, and we conclude the paper in section 5.

## 2 Preliminaries

In this part, we briefly present the preliminaries used in this paper. The schemes we mention are all based upon elliptic curve cryptography(ECC). The security of the ECC lies on the difficulty of solving the elliptic curve discrete logarithm problem (ECDLP), and it can achieve same security as of RSA with the key of fewer bits. Let

$G$ denote group of points on an elliptic curve with prime order $q$, and $P$ is a generator. $+/-$ means elliptic curve point addition/subtraction. $H(\cdot)$ is a collision-resistant hash function. Some mainly used hard problems related to ECC are given below:

**Definition 1 (Elliptic Curve Discrete Logarithm Problem (ECDLP)).** Given $P, Q \in G$, it is hard to find the integer $k \in Z_q^*$ such that $Q = kP$.

**Definition 2 (Computational Diffie-Hellman Problem (CDHP)).** For any $a, b \in Z_q^*$, given $(P, aP, bP)$, the computation of $abP$ is hard.

**Definition 3 (Decisional Diffie-Hellman Problem (DDHP)).** For any $a, b, c \in Z_q^*$, given $(P, aP, bP, cP)$, it is hard to decide whether or not $cP = abP$, i.e. decide $c \equiv ab \bmod q$ or not.

## 2.1 RFID Systems

We assume that an RFID system is composed with one reader and many tags. The reader is not corruptible and all the data stored in reader side are secure. Only the wireless link established between the reader and the involving tag during a protocol instance is insecure. Tags are not tamper-proofed.

**Definition 2.1 RFID Authentication scheme.** An RFID authentication scheme is defined by two setup algorithms and the actual protocol.

- SetupReader($1^s$) is used to generate the required system public parameters $K_P$ and reader's private parameters $K_S$ by supplying a security parameter $s$.
- SetupTag ($ID$) is used to generate necessary tag secrets key $K_{ID}$ and memory states $S_{ID}$ by inputting $K_P$ and a custom unique $ID$. $S_{ID}$ can be updated during the protocol. Notice that $K_{ID}$ and $S_{ID}$ are not public and are not available to the adversary unless the tag is corrupted.
- the actual protocol used to identify/authenticate tags with the reader.

The main security objective of an RFID system is to ensure that only legitimate tags are accepted by honest readers (tag authentication). Many application cases additionally require reader to determine the authentic tag identity(tag identification). Moreover, there are several applications (e.g., electronic tickets) where reader authentication is a fundamental security property. Here we only consider tag authentication.

The most deterrent privacy risk concerns the tracking of tag users, which allows the creation and misuse of detailed user profiles in an RFID system and an RFID system should provide anonymity (confidentiality of the tag identity) as well as untraceability (unlinkability of the communication of a tag) even if the state of a tag has been disclosed.

### 2.2 JW Privacy Model for RFID Systems

Here we briefly summarize JW privacy model[8], which based on indistinguishability of tags. The oracles the adversary can access include: $CreatTag(ID)$ allows the creation of a free tag; $Launch()$ starts a protocol instance at reader's side and a unique handle $\pi$ of this instance is returned; $Sendreader(m,\pi)$ sends a message $m$ to the reader for the handle $\pi$, and $SendTag(m,\pi)$ sends a message $m$ to the tag determined by handle $\pi$; $result(\pi)$ returns either 1 if the instance $\pi$ completed with success or 0 otherwise; $Corrupt(tag)$ returns all the internal secrets of $tag$.

Here we give the backward privacy definition based on the notion of indistinguishability game.

**Definition 2.2 Backward Privacy.** Backward privacy is defined using the game played between the adversary $A$ and a collection of reader and tag instances. $A$ runs the game whose setting is as follows:

First the system is set up, and the adversary $A$ obtains the corresponding public parameters. Then via the learning phase, $A$ can access to all the oracles above. After that, the challenger chooses two tags $\{Tag_0, Tag_1\}$, and both tags can be corrupted by the adversary already. After a randomly bit $b \in \{0,1\}$ is chosen, the adversary can make a polynomial number of oracle calls to the system, but cannot corrupt the challenged tag $Tag_b$ any more. At last, the adversary outputs a guess bit $b' \in \{0,1\}$ indicating his guess of the value of $b$. The success of $A$ in winning the game and thus breaking the notion of backward privacy is quantified in terms of $A$'s advantage in distinguishing $\{Tag_0, Tag_1\}$, i.e. it correctly guesses b.

We say the protocol is considered backward privacy if ($\varepsilon$ is negligible):

$$pr(A\ guesses\ b\ correctly) \le 0.5 + \varepsilon$$

## 3 Remarks on Randomized GPS and Randomized Hashed GPS

In this part, we first review the Randomized GPS and Randomized Hashed GPS schemes[19], then we give an impersonate attack on both two schemes, and man-in-the-middle attack on Randomized GPS scheme to show they can not provide backward privacy as to adversary in JW model.

**Randomized GPS.** The secret/public key pairs of the tag and the reader are $(s, I = sP)$ and $(v, U = vP)$, and the scheme is executed as follows:

1. the tag randomly selects $r_1, r_2 \in Z_q^*$, computes and sends the reader $A_1 = r_1P$ and $A_2 = r_2U$;
2. After receiving the messages, the reader randomly picks $c \in Z_q^*$, and sends it to the tag;
3. the tag computes and sends the reader $y = r_1 + r_2 + sc$ **;**

4. reader checks whether the equation $yU = vA_1 + A_2 + cvI$ holds. If it holds, the reader accepts the tag; Otherwise the reader rejects the tag as illegitimate.

**Randomized Hashed GPS.** The secret/public key pairs of the tag and the reader are the same as the above, and the scheme is executed as follows:

1. the tag randomly selects $r_1, r_2 \in Z_q^*$ , computes and sends the reader $z = H(r_1P, r_2U)$;
2. the reader randomly picks $c \in Z_q^*$, and sends it to the tag;
3. After receiving the challenge, the tag computes $y = r_1 + r_2 + sc$**,** and sends the reader $A_1 = r_1P, A_2 = r_2U$ and $y$**;**
4. reader checks whether the equations $yU = vA_1 + A_2 + cvI$ and $z = H(A_1, A_2)$ hold. If they hold, the reader accepts the tag; Otherwise the reader rejects the tag as illegitimate.

As to a JW adversary, he can access to oracle $result(\pi)$, and by definition of oracle $Corrupt(tag)$, the adversary knows the corrupted tag's secret key $s$. We present an impersonation attack on Randomized (Hashed) GPS scheme and man-in-the-middle attack on Randomized GPS scheme to track the identity of the tag.

**Impersonation attack on Randomized (Hash) GPS scheme.** Assume the adversary has obtained some tag's authentication messages, i.e. $\{A_1, A_2, c, y\}$ in Randomized GPS scheme**,** in order to test whether the random tag is the corrupted one, the attacker impersonates the tag to have an authentication operation with the reader. The attack is illustrated as follows:

1. the adversary replays the authentication messages $A_1$ and $A_2$;
2. the reader randomly picks the challenge $c^* \in Z_q^*$**,** and sends it to the tag;
3. the tag computes and sends the response $y^* = y - sc + c^*s$**.**

If the reader accepts the tag as legitimate, the adversary can decide the tag is the corrupted one. Because if both tags are the same, the response $y^* = y - sc + c^*s = r_1 + r_2 + c^*s$ is a right response**;** Otherwise the correct response should be $r_1 + r_2 + c^*s^*$, where $s^*$ is the secret of the uncorrupted tag, which is not the same as $y^*$.

Although in Randomized Hashed GPS scheme, the hash function is applied to the first message, and the author claimed Randomized Hashed GPS scheme can enhance privacy, it is easy to see that this scheme can not resist the impersonation attack we present.

**Man-in-the-middle attack on Randomized GPS scheme.** After the random tag sends the reader $A_1 = r_1P, A_2 = r_2U$ with two randomly selected $r_1, r_2 \in Z_q^*$, the adversary executes the man-in-the-middle attack as follows:

1. After obtaining the challenge $c \in Z_q^*$ sent by the reader, the adversary selects a random $c^*$, computes and sends the tag $c + c^*$;
2. After the adversary gets the tag's response $y$, he changes it as $y^* = y - c^*s$ with the corrupted tag's secret $s$. The adversary sends the reader $y^*$ at last.

If the reader accepts the tag as legitimate, the adversary can determine the tag is the same as the corrupted one. Because now the response $y^* = y - c^* s = r_1 + r_2 + cs$ is the right response; Otherwise the changed response is $r_1 + r_2 + (c + c^*)s^* - c^* s$, and the right response should be $r_1 + r_2 + cs^*$, where $s^*$ is the real secret of the tag. These two values are not the same, so the reader will reject the authentication.

## 4 Our Construction with Backward Privacy

From the analysis in the section 3, we can see if the adversary can not access to $result(\pi)$ oracle, it is difficult to execute many forms of security attacks, because the adversary can not determine the effect of their changes on the communication messages. In this part, we give our construction of the ECC-based authentication scheme with backward privacy in the JW model. We name it as Challenge-Hiding GPS scheme because as to a passive adversary, he can not deduce the real challenge used in the protocol.

### 4.1 Our Constructions

In our scheme, the secret/public key pairs of the tag and the reader are $(s, I = sP)$ and $(v, U = vP)$, and the scheme is executed as follows:

1. the tag randomly selects $r \in Z_q^*$, computes and sends the reader $A_1 = rP$;
2. the reader randomly picks $c \in Z_q^*$, and sends it to the tag;
3. After receiving the message $c$, the tag first computes $A_2 = rU$, and the actual challenge $c^* = H(A_2, c)$. At last the tag computes and sends the reader $y = r + sc^*$;
4. When receiving the response $y$, the reader computes $A_2' = vA_1$, $c' = H(A_2', c)$, and checks whether there exists tag's public key $I$ satisfying equation $yP = A_1 + c'I$. if the equation holds, the reader accepts the tag; Otherwise the reader rejects the tag.

We can see in our scheme, the real challenge $c^*$ is computed using the message $c$ from the reader and a Diffie-Hellman key agreement value $A_2$ between the tag and the reader. While in the schemes in section 3 and Schnorr scheme, the challenge is sent by the reader. As to a passive adversary, given $A_1$ and $U$, he can not obtain the value of $A_2$ because of the difficulty of Computational Diffie-Hellman Problem. So, the actual challenge is hiding from the passive adversary.

## 4.2 Performance and Security Analysis of Our Scheme

Before giving the security analysis of our scheme, we first compare our Challenge-Hiding scheme(CH-GPS) with the Randomized GPS(R-GPS) scheme and Randomized Hashed GPS(RH-GPS) scheme according to computation and communication overhead in the following table 1, where ECPM/ECA means Elliptic Curve point multiplication/addition operation; AM/AA means ordinary arithmetic multiplication/addition operation, Hash means hash function and CO is communication overhead:

**Table 4.1** The comparison of our scheme with Randomized (Hashed) GPS scheme

| Schemes | ECPM | ECA | AM | AA | Hash | CO |
|---|---|---|---|---|---|---|
| CH-GPS(Tag) | 2 | 0 | 1 | 1 | 1 | 2 |
| R-GPS (Tag) | 2 | 0 | 1 | 2 | 0 | 3 |
| RH-GPS(Tag) | 2 | 0 | 1 | 2 | 1 | 4 |
| CH-GPS(Reader) | 3 | 1 | 0 | 0 | 1 | 1 |
| R-GPS (Reader) | 3 | 2 | 1 | 0 | 0 | 1 |
| RH-GPS(Reader) | 3 | 2 | 1 | 0 | 1 | 1 |

From the comparison, we can conclude that our scheme has the best communication overhead just the same as the basic Schnorr identification scheme**.** As to computation, our scheme is better than the Randomized Hashed GPS scheme and has more a hash function operation than that of the Randomized GPS scheme.

In the following, we give the security analysis of the authenticity and privacy of Challenge-Hiding GPS scheme proposed.

**Theorem 4.1 (**Authenticity**).** Assume $H(\cdot)$ is preimage and collision resistant hash function, and assume the hardness of the DH problem, Challenge-Hiding GPS scheme satisfies Honest-Verifier Zero-knowledge in the random oracle model.

Proof: Honest-Verifier Zero-Knowledge means there exists a simulator $Sim$ able to simulate a protocol instance given the prover's identity $I$ and a challenge $c$, i.e. $Sim(c;I)$ outputs a pair $A$ and $y$, such that $[A:c:y]$ is a valid identifying transcript.

The eavesdropping adversary learns the tuple $(A_1:c:y)$ just as the Schnorr identification scheme. It is easy to see that the random variables $A_1$**,** $c$, $y$ are individually uniformly distributed on their domains. However, the real challenge is not the value $c$ but $c^*$ generated by the scheme. If the verifier publishes his secret key to the simulator, the adversary can deduce the real challenge, and the views of the adversary in our scheme is just the same as in Schnorr scheme.

In the random oracle model, as to a challenge $\tilde{c}$ , the simulator $Sim$ can first choose randomly $\tilde{y}\in Z_q$ and $\tilde{c}'\in Z_q$**,** then computes $\tilde{A}=\tilde{y}P-\tilde{c}'I$**,** and sets the hash value $H(v\tilde{A},\tilde{c})$ as $\tilde{c}'$ . The tuples $(A_1:c:y)$ and $(\tilde{A}:\tilde{c}:\tilde{y})$ are then identically distributed.

**Theorem 4.2(**Backward Privacy**).** Assume $H(\cdot)$ is preimage and collision resistant hash function, and assume the hardness of the DH and CDH problem, Challenge-Hiding GPS scheme can provide with backward privacy in JW model.

Proof: In the JW security model, the adversary can corrupt the tag and retrieve the secret of the tag, i.e. the value $s$. In the learning phrase, the adversary can not get the information of the reader's secret key because of the zero-knowledge property.

After selecting the challenged tag $Tag_b$, the adversary can actively involve in the authentication. He can impersonate the legitimate tag or the reader, but from the CDH problem and the random distribution of the hash function, he can not deduce the real challenge used in each authentication. Here to track the identity of the tag, it is meaningless for the adversary to generate a new commitment $A_1$ to send when impersonating the tag (In this way, the adversary will know the hiding challenge).

In the equation of $y = r + sc^*$, there must exist two unknown variables $r$ and $c^*$. And from the verification equation $yP = A_1 + c^*I$, the adversary can not link the identity of the challenge tag with some public key $I$ because of the hardness of DH problem. That is to say, the view of the adversary is uniformly distributed, so the adversary can not have non-negligible advantage to guess the bit $b$.

Here we show that if there exists an algorithm $ALG_1$ to break the backward privacy with advantage $\varepsilon$, we can construct an algorithm $ALG_2$ to break the DDH problem. The input to the $ALG_2$ is $(P, P_1 = aP, bP, hP)$, $ALG_2$ randomly selects $s \in Z_q$ as the secret of the tag, and $bP$ is the public key of the reader, which all send to algorithm $ALG_1$. To execute the authentication, the $ALG_2$ can randomly select $r \in Z_q^*$, and sets $A_1 = rP_1$; and as to the challenge $c$, he can compute the real challenge $c^* = H(rhP, c)$. We can see if $hP = abP$, then $c^*$ is the correct challenge; otherwise it is not computed correctly. The response of the tag is $y = r + c^*s$, so the verification equation can be modified as $yP_1 = A_1 + c^*sP_1$.

If $hP = abP$ does not hold, the views of the $ALG_1$ are randomly distributed; while if $hP = abP$ holds, the views of the $ALG_1$ are real authentication distribution. So we can get:

$$pr(ALG_2 \;\; win) = pr(ALG_1 \;\; win / hP = abP) +$$
$$pr(ALG_1 \;\; win / hP \neq abP) = \frac{1}{2} * (\frac{1}{2} + \varepsilon) + \frac{1}{2} * \frac{1}{2} = \frac{1}{2} + \frac{1}{4}\varepsilon$$

## 5 Conclusions

Privacy is an important issue required as to the security of RFID system, and backward privacy is a very strong privacy definition. In this paper, we remark on the security of two efficient public key authentication schemes, and show they can not provide backward privacy as to the adversary in JW privacy model. Via hiding the challenge using the technique of Diffie-Hellman key agreement scheme, we present a new scheme satisfying backward privacy, and our scheme has the best communication overheads and the same computation efficiency, compared to these two schemes.

**Acknowledgments.** This work is supported by the Priority Academic Program Development of Jiangsu Higher Education Institutions(PAPD), National Natural Science Funds (Grant No.60903181) and Nanjing University of Posts and Telecommunications Funds (Grant No.NY208072).

## References

1. Feldhofer, M., Dominikus, S., and Wolkerstorfer, J.: Strong Authentication for RFID Systems using the AES Algorithm[C]. In Cryptographic Hardware and Embedded Systems (CHES'04), LNCS, volume 3156, pp. 357–370. Springer-Verlag, 2004.
2. Juels, A., Weis, S.: Authenticating Pervasive Devices with Human Protocols. In Advances in Cryptology - CRYPTO'05, LNCS, volume 3126, pp. 293–308. Springer-Verlag, 2005.
3. Gilbert, H., Robshaw, M., Sibert, H.: An Active Attack Against HB+ - a Provably Secure Lightweight Authentication Protocol. IET Electronic Letters, volume 41(21), pp. 1169–1170, 2005.
4. Bringer, J., Chabanne, H., Dottax, E.: HB++: a Lightweight Authentication Protocol Secure against Some Attacks. In Security, Privacy and Trust in Pervasive and Ubiquitous Computing (SecPerU '06), pp. 28–33. IEEE Computer Society, 2006.
5. Bringer, J., Chabanne, H.: Trusted-HB: A Low-Cost Version of HB+ Secure Against Man-in-the-Middle Attacks. IEEE Transactions on Information Theory, volume 54(9), pages 4339–4342, 2008.
6. Gilbert, H., Robshaw, M., Seurin, Y.: Good variants of HB+ are hard to find. In Proc. In Financial Cryptography and Data Security, pp. 156-170, 2008.
7. Frumkin, D., Shamir, A.: Un-Trusted-HB: Security Vulnerabilities of Trusted-HB. In International Workshop on RFID Security (RFIDsec'09), pp. 62–71, 2009.
8. Juels, A., Weis, S.A.: Defining strong privacy for RFID. In PERCOMW, pages 342-347. IEEE Computer Society, 2007.
9. Le, T.V., Burmester, M., de Medeiros, B.: Universally composable and forward-secure RFID authentication and authenticated key exchange. In ASIACCS 2007, pages 242-252. ACM, 2007.
10. Vaudenay, S.: On privacy models for RFID. In ASIACRYPT, pages 68-87, 2007.
11. Ohkubo, M., Suzuki, K., Kinoshita, S.: Cryptographic Approach to "Privacy-friendly" Tags. Proceedings of RFID Privacy Workshop, MIT, 2003.
12. Hein, D., Wolkerstorfer, J., Felber, N.: ECC is Ready for RFID - A Proof in Silicon. In Selected Areas in Cryptography, LNCS, volume 5381, pp. 401–413. Springer-Verlag, 2009.
13. Lee, Y.K., Batina, L., Singelee, D., Verbauwhede, I.: Low-Cost Untraceable Authentication Protocols for RFID. In Proceedings of the 3rd ACM Conference on Wireless Network Security (WiSec'10), pp. 55–64. ACM, 2010.
14. Lee, Y.K., Batina, L., Verbauwhede, I.: EC-RAC(ECDLP based Randomized Access Control): Provably Secure RFID Authentication Protocol. In IEEE International Conference on RFID 2008, pp. 97-104, IEEE, 2008.
15. Bringer, J., Chabanne, H., Icart, T.: Cryptanalysis of EC-RAC, an RFID Identification Protocol. In International Conference on Cryptology and Network Security - CANS'08, LNCS, volume 5339, pp. 149–161.Springer-Verlag, 2008.

16. Deursen, T., Radomirovic, S.: EC-RAC: Enriching a Capacious RFID Attack Collection. In International Workshop on RFID Security (RFIDSEC'10), LNCS, volume 6370, pp. 75–90. Springer-Verlag, 2010.
17. Fan, J., Hermans, J., Vercauteren, F.: On the Claimed Privacy of EC-RAC III. In International Workshop on RFID Security (RFIDSEC'10), LNCS, volume 6370, pages 66–74. Springer-Verlag, 2010.
18. Lee, Y.K., Batina, L., Verbauwhede, I.: Untraceable RFID Authentication Protocols: Revision of EC-RAC. In IEEE International Conference on RFID, pp. 178–185. IEEE, 2009.
19. Bringer, J., Chabanne, H., Icart, T.: Efficient zero-knowledge identification schemes which respect privacy. In Proceedings of ASIACCS. 2009, pp. 195-205.
20. Girault, M., Poupard, G., Stern, J.: On the fly authentication and signature schemes based on groups of unknown order. J. Cryptology, 19(4):463-487, 2006.