Pairing Free Certificate Based Signcryption Schemes Using ECQV Implicit Certificates

A. Braeken
Industrial Engineering (INDI)
Vrije Universiteit Brussel (VUB)
Pleinlaan 2, 1050 Brussl, Belgium
[e-mail: abraeken@vub.be]

Received December 4, 2017; revised June 7, 2018; accepted July 28, 2018; published March 31 2019

Abstract

Signcryption schemes offer the possibility to simultaneously sign and encrypt a message. In order to guarantee the authentication of both signer and receiver in the most efficient way during the signcryption, certificate based solutions have been proposed in literature.
We first compare into detail three recently proposed certificate based signcryption systems relying on the elliptic curve discrete logarithm problem and without the usage of compute intensive pairing operations. Next, we demonstrate how the performance of these certificate based systems can be improved by using the Elliptic Curve Qu Vanstone (ECQV) implicit certificates. What is more, generalized signcryption schemes are easily derived from these schemes and the anonymity feature of sender and receiver is already inherently included or can be very efficiently obtained without a significant additional cost.

Keywords: ID Based, Signcryption, Implicit certificates, Elliptic Curve Cryptography, Authentication, Anonymity

1. Introduction

Many applications like electronic payments, supply chain management and data collection/monitoring systems require a strong level of user authentication. Authentication is typically obtained through public key infrastructure (PKI) mechanisms, managed by a certificate authority (CA). However, for small devices in an Internet of Things (IoT) context, a PKI requires too high computation, maintenance, and storage. Consequently, more efficient approaches are needed.

We identify three different alternatives in literature to establish user authentication, being the identity (ID) based schemes [1], the certificateless [2], and certificate based [3] approaches. In the ID based schemes, a private key generator (PKG) constructs the private and public key of the user with the property that the public key is equal to a known identity of the user. Although this leads to simple key management, the ID based mechanisms are composed of computationally demanding cryptographic pairing operations, have inherent key escrow, and require a secure channel between the PKG and the user to share the private key. In the certificateless schemes, the private key of the user is generated by means of secret information coming both from the PKG and the user itself. Therefore, certificateless schemes do not have inherent key escrow, but still require a secure channel between the PKG and the user.

Only the certificate based systems are able to address all of the above mentioned problems and in particular have no need for secure channels in the derivation of the key material. These schemes make use of a certificate authority (CA) for the generation of the certificates. In the certificate based approach, the user first generates its own key pair and requests a certificate of the CA on it. As a result of this process, the public key of the user needs to be extended with an additional parameter, derived from the CA's certificate and responsible for the relation between identity and the first part of the public key. Note that the link between user and public key is not validated in the beginning, but the actual validation is obtained only by including this additional public key parameter in the rest of the security protocol.

In this paper, we will focus on certificate based signcryption schemes [4]. Signcryption schemes are very interesting as they allow to perform both encryption and signature generation in one single phase. They are much more efficient than the traditional approach in which the message is first encrypted and then signed. Consequently, signcryption schemes are able to offer simultaneously confidentiality, authentication, non-repudiation, and integrity.

In literature, there have been recently two different certificate based pairing free systems described, which are proven to be secure in the random oracle model against chosen-ciphertext attacks and existentially unforgeable against chosen-message attacks. The system in [5] is based on the discrete logarithm problem (DL) and the other system in [6] on the elliptic curve discrete logarithm problem (ECDLP). In this paper, the scheme of [5] will be explicitly translated in elliptic curve (EC) terminology, and thus compared with [6]. In addition, another scheme is proposed similar as the one of [6], but slightly more efficient since it uses additions instead of inverse operations in the field. It is based on ideas coming from the signcryption scheme described in [7]. The similarities and differences in these three schemes are discussed and the efficiency analysis, provided in [5], is questioned.

In order to further improve the efficiency of certificate based signcryption schemes, we propose to use an alternative process for the generation and usage of the certificates, by applying the Elliptic Curve Qu Vanstone (ECQV) implicit certificates. Here, the CA generates

a certificate based on the identity of the user and some random values. From this certificate, the user can derive its key pair and any other user is able to find the same public key, given the identity and the certificate of the user, relying on the authenticity of the public key of the CA. Note that the computation of the public key can be performed offline. We show how to translate the traditional certificate based signcryption schemes to signcryption schemes using the implicit ECQV certificates, which results in less cryptographic operations. In order to make the difference between the traditional certificate based approach and this proposed approach, we will use the terms of explicit and implicit certificate based schemes.

Finally, we explain how the previous schemes can be easily adapted to a generalized signcryption (GSC) scheme, providing one single framework to either establish confidentiality, authentication, or a combination of both. We also show how anonymity of sender and receiver is already included in the schemes or can be easily added to the other schemes without a significant additional computation or communication cost.

To summarize, the main contributions of the paper are the following:

- Classification of the different proposed explicit certificate based signcryption schemes
  and proposal of a new one, which is slightly better than the other existing proposals in literature.
- Proposal of implicit certificate based signcryption schemes, relying on ECQV implicit certificates and linked with the explicit based signcryption schemes, which are more performant than these explicit certificate based signcryption schemes from literature.
- Classification and comparison with respect to the number of compute intensive operations and time performance of the implicit and explicit based signcryption schemes.
- Proposal of a certificate based GSC.
- Proposal of a certificate based signcryption scheme, providing anonymity of sender and receiver.

The outline of the paper is as follows. In Section 2, related work is described. Section 3 discusses the preliminaries. In Section 4, the different explicit certificate based signcryption systems are presented and compared. Section 5 describes the certificate based approach with ECQV implicit certificates, the corresponding signcryption schemes based on it, and an associated performance analysis. In Section 6, the relation with a GSC scheme and the inclusion of anonymity is described. Section 7 describes the security analysis. Finally, Section 8 presents the conclusions of the the paper.

2. Related Work

In 2002, Malone [8] introduced the first ID based signcryption scheme, together with a comprehensive security model. The classical ID based signcryption schemes make use of computationally intensive pairing operations. As shown in [9], for binary fields, pairing operations behave almost 5 times worse than EC point multiplication operations in timing and energy performance.

In 2008, the introduction of the certificateless approach in signcryption schemes has been proposed in [10,11]. The same year, also certificate based signcryption schemes [12] have been introduced. Most of the certificate based and certificateless signcryption schemes are also based on pairing operation. However, very recently two paring free certificate based systems

have been made proposed [5,6]. A performance comparison in [5] was given to compare the schemes between [5,6,13,14], showing that [5] was outperforming the others. The schemes [13,14] are making use of pairing operations. Unfortunately, we will show that wrong conclusions are made for the performance comparison between [5] and [6], probably due to a wrong translation, as [5] was expressed as a discrete logarithm problem (DLP) and [6] as an ECDLP. In addition, we add another certificate based and pairing free signature scheme, similar to the scheme of [6], following ideas of [7], where the signature scheme is based on the proposal of Schnorr [15].

On the other hand, many pairing free signcryption schemes based on elliptic curve cryptography (ECC) without the specific condition of ID based authentication can also be found in literature, see survey [16]. In these schemes, the guarantee that a given public key belongs to a certain user is explicitly assumed, for instance by a third party who is checking the integrity of the stored public key and identity data. This is a quite strong requirement. In particular, among the most efficient proposals in literature, we distinguish [7] where an efficient EC based GSC scheme is discussed. Also an anonymous EC based signcryption scheme, called ASEC, has been described in [17].

The newly proposed type of certificate based signcryption scheme will rely on the ECQV Implicit Certificate Scheme [18] as key management protocol, which uses elliptic curve operations and results in much more lightweight public key cryptographic (PKC) solution, compared to RSA based PKC systems [19].

To conclude, this paper will firstly analyze three variants for pairing free signcryption schemes with traditional certificates, based on inputs of [5-7]. Next, a certificate based approach with ECQV implicit certificates will be proposed, leading to more efficient certificate based signcryption schemes. Finally, it will be shown how GSC schemes and anonymity can be obtained in the proposed schemes. In particular, the performance of ASEC [17] will be drastically improved.

3. Preliminaries

Elliptic Curve Cryptography (ECC) is based on the algebraic structure of elliptic curves over finite fields. We denote the curve in the finite field $GF(2^p)$ by $E_p(a,b)$, defined by the equation $y^2 + xy = x^3 + ax + b$ with a and b two constants in $GF(2^p)$. We denote by P the base point generator of $E_p(a,b)$ of order 2^p. The EC based public key cryptography (PKC) system is based on the following two problems.

- Elliptic Curve Discrete Logarithm Problem (ECDLP) states that given two EC points P and Q of $E_p(a,b)$, it is computationally hard for any polynomial-time bounded algorithm to determine a parameter x in $GF(2^p)^*$, such that $Q=xP$.

- The Computational Discrete Logarithm Problem (CDLP) states that given 3 points, P, xP, yP (x,y in $GF(2^p)^*$) of $E_p(a,b)$, it is computationally infeasible to derive the EC point $xyP=yxP$.

In addition, a one-way cryptographic hash function (e.g. SHA2, SHA3) that results in a number of $GF(2^p)$ is denoted by $H(.)$. Given the messages M_1 and M_2, the concatenation of them is denoted by $M_1 | M_2$ and the bitwise XOR operation by $M_1 \oplus M_2$. We assume that the length of the message is less or equal than the size of the hash function output. If not, an encryption algorithm, like e.g. AES, should be used, instead of the xor operation to encrypt the

message.

4. Certificate based signcryption

A traditional certificate based signcryption scheme, as proposed in [5,6], consists of the following 5 phases. The sender and receiver are denoted by *S* and *R* respectively.

- Setup: In this phase, the CA generates the master secret key *msk* and system parameters *params*, based on a given security parameter. These system parameters *params* are published.
- *SetKeyPair*: This algorithm is working at the user's side. Given *params*, the private key sk_U and public key pk_U of the user with identity ID_U are generated. The public key together with the user's identity is sent to the CA.
- *Certification*: The CA generates based on the user's identity ID_U and public key pk_U together with the system parameters *params*, a certificate $cert_U$ for each user. The CA sends the certificate to the user over an open channel.
- *Signcryption* $S_{SR}(.)$: This function is executed by the sender *S* and has the goal to encrypt and sign the message *m*. The input of the function contains the message *m*, the identity ID_S, certificate $cert_S$ and private key sk_S of the sender, the identity ID_R and public key pk_R of the receiver, together with the system parameters *params*. The result is called the signcrypted message, denoted by:

$$C_{SR}=S_{SR}(m,ID_S,ID_R,cert_S,sk_S,pk_S,pk_R,,params)$$

- *Unsigncryption* $U_{RS}(.)$: This function is executed by the receiver *R*, after reception of the message C_{SR} and has the goal to derive the original message *m* and to verify the corresponding signature on it. The input of the function contains the identity ID_S and public key pk_S of the sender, the identity ID_R, the certificate $cert_R$, private key sk_R and the public key pk_R of the receiver, together with the system parameters *params*. The output of the function

$$U_{RS}(C_{SR},ID_S,ID_R,cert_R,sk_R,pk_S,pk_R,,params)$$

  is equal to *m'* if the verification of the signature is correct. If the signature is not valid, the output equals to $\perp$. The signcryption algorithm is correct if *m* equals to *m'*.

Table 1. summarizes the notations frequently used in this paper.

Table 1. Notations

$msk = \alpha$	Master secret key
$G_{CA} = \alpha P$	Master public key, *P* is generator point
$params = params=\{P,EC,G_{CA}\}$	Public parameters
ID_S,ID_R	Identity sender and receiver
(sk_S,pk_S), (sk_R,pk_R)	Private, public key of sender and receiver
$cert_S,cert_R$	Certificate of sender and receiver
m	Message

We now discuss the three types of signcryption algorithms. For the first three phases, they all satisfy the same steps, leading to the same notations (see [6]).

- Setup: The CA defines the public system parameters, consisting of an EC in $GF(2^p)^*$, a generator P on that curve and an EC point $G_{CA}=\alpha P$. The random value $\alpha \in GF(2^p)^*$ used in the computation of G_{CA} corresponds with the master key. To conclude $params=\{P,EC,G_{CA}\}$ and $msk=\alpha$.
- *SetKeyPair*: Given the identity ID_U of the user and *params*, the user selects a random value $d_U \in GF(2^p)^*$ as the private key, thus $sk_U=d_U$. The corresponding public key equals to $pk_U= P_U = d_UP$. The tuple (ID_U,P_U) is sent to the CA.
- *Certification*: Based on the user's input (ID_U,P_U), the CA selects a random value $r_U \in GF(2^p)^*$ and computes $R_U = r_UP$. Next, the certificate for the user is defined as

$$cert_U = r_U + \alpha H(ID_U|P_U|R_U). \quad (1)$$

Both $cert_U$ and R_U are sent to the user. The public key of the user is the tuple (P_U, R_U).

We now discuss the signcryption and unsigncryption algorithms between S and R for the three different schemes, based on [5,6,7] respectively.

4.1 Scheme 1

The signcryption $S_{SR}(m,ID_S,ID_R,cert_S,sk_S,pk_S,pk_R,params)$ with $pk_s=(P_S, R_S)$ and $pk_r=(P_R, R_R)$ consists of the following steps.

- Choose a random value $r \in GF(2^p)^*$, compute $R=rP$.
- $k = r(P_R+R_R+H(ID_R|P_R|R_R)G_{CA})$
- $C_1 = m \oplus H(k)$
- $C_2=cert_S+d_SH(P_S|R_S|C_1|R)+rH(ID_S|R_S|C_1|R)$

The signcryption algorithm has output $C_{SR} =(R,C_1,C_2)$.

The unsigncryption process, denoted by U_{RS} $(C_{SR},ID_S,ID_R,cert_R,sk_R,pk_S,pk_R,params)$, with $pk_s=(P_S, R_S)$ and $pk_r=(P_R, R_R)$ consists of the following steps.

- The receiver first checks if
  $C_2P = R_S + H(ID_S|P_S|R_S)G_{CCA} + H(P_S|R_S|C_1|R)P_S + H(ID_S|R_S|C_1|R)R$
- If this check is positive, then the key is defined by the equality:
  $k=(d_R + cert_R)R.$
  and thus the final message m is derived by
  $m = C_1 \oplus H(k)$

Otherwise, the output equals to $\perp$. Note that the scheme is correct since

$$\begin{aligned} C_2P &= (cert_S+d_SH(P_S|R_S|C_1|R)+rH(ID_S|R_S|C_1|R))P \\ &= cert_SP + d_SH(P_S|R_S|C_1|R)P+rH(ID_S|R_S|C_1|R)P \\ &= (r_S+\alpha H(ID_S|P_S|R_S))P + H(P_S|R_S|C_1|R)d_SP + H(ID_S|R_S|C_1|R)rP \end{aligned}$$

$$= R_S + H(ID_S|P_S|R_S)G_{CA} + H(P_S|R_S|C_1|R)P_S + H(ID_S|R_S|C_1|R)R$$

Also, for the key derivation, we see that

$$\begin{aligned} k &= (d_R + cert_R)R \\ &= (d_R + r_R + \alpha H(ID_R|P_R|R_R))R \\ &= d_R rP + r_R rP + \alpha H(ID_R|P_R|R_R)rP \\ &= rP_R + rR_R + rH(ID_R|P_R|R_R)G_{CA} \end{aligned}$$

4.2 Scheme 2

The signcryption phase with inputs $m, ID_S, ID_R, cert_S, d_S, (P_S, R_S), (P_R, R_R), params$, consists of the following steps.

- Choose a random value $r \in GF(2^p)^*$, compute $R=rP$.
- $k = r(P_R+R_R+H(ID_R|P_R|R_R)G_{CA})$
- $C_1 = m \oplus H(k)$
- $h = H(m|R|ID_S|P_S|R_S)$
- $C_2=r(d_S+cert_S+h)^{-1}$

The signcryption algorithm has output $C_{SR} = (h,C_1,C_2)$.

For the unsigncryption process with input $C_{SR}, ID_S, ID_R, cert_R, sk_R, (P_S, R_S), (P_R, R_R)_,, params$, the receiver first computes

$$R' = C_2(P_S + R_S + H(ID_S|P_S|R_S)G_{CA} + hP)$$

Next, also the key $k'=(d_R + cert_R)R'$ is derived in order to find $m' = C_1 \oplus H(k')$.
Finally, the signature is verified by checking the following equality

$$h = H(m'|R'|ID_S|P_S|R_S).$$

If so, $m=m'$, or otherwise the output equals to $\perp$.

The correctness of the scheme follows from the following reasoning:

$$\begin{aligned} R' &= C_2(P_S + R_S + H(ID_S|P_S|R_S)G_{CA} + hP) \\ &= r(d_S+cert_S+h)^{-1} (d_S P + r_S P + H(ID_S|P_S|R_S)\ \alpha P + hP) \\ &= r(d_S+cert_S+h)^{-1} (d_S P + cert_S P + hP) = rP = R \end{aligned}$$

4.3 Scheme 3

The signcryption phase with inputs $m, ID_S, ID_R, cert_S, d_S, (P_S,R_S), (P_R, R_R), params$, consists of the following steps.

- Choose a random value $r \in GF(2^p)^*$, compute $R=rP$.
- $k = r(P_R+R_R+H(ID_R|P_R|R_R)G_{CA})$
- $C_1 = m \oplus H(k)$
- $h = H(m|R|ID_S|P_S|R_S)$
- $C_2=r - h(d_S+cert_S)$

The signcryption algorithm has output $C_{SR} = (h,C_1,C_2)$.

For the unsigncryption process with inputs C_{SR}, ID_S, ID_R,$cert_R$, d_R, (P_S, R_S), (P_R, R_R), *params*, the receiver first computes

$$R' = C_2P + h(P_S + R_S + H(ID_S|P_S|R_S)G_{CA})$$

Next, also the key $k'=(d_R + cert_R)R'$ is derived in order to find $m' = C_1 \oplus H(k')$. Finally, the signature is verified by checking the following equality

$$h = H(m'|R'|ID_S|P_S|R_S).$$

If so, $m=m'$, ortherwise the output equals to $\perp$.

The correctness of the scheme for the key derivation is similar as the two other schemes. For the authentication, it follows from the fact that

$$\begin{aligned} R' &= C_2P + h(P_S + R_S + H(ID_S|P_S|R_S)G_{CA}) \\ &= (r - h(d_S+cert_S))P + h(P_S + R_S + H(ID_S|P_S|R_S)G_{CA}) \\ &= rP - hd_S P - cert_S P + hP_S + hR_S + hH(ID_S|P_S|R_S)G_{CA} \\ &= rP - hd_S P - hcert_S P + hP_S + hcert_S P = rP \end{aligned}$$

Sender: $S_{SR}(m,ID_S,ID_R,cert_S,sk_S,pk_S,pk_R,,params)$	**Receiver:** $U_{RS}(C_{SR},ID_S,ID_R,cert_R,sk_R,pk_S,pk_R,,params)$
Scheme 1 $r \in GF(2^p)^* \rightarrow R=rP$ $k = r(P_R+R_R+H(ID_R\|P_R\|R_R)G_{CA})$ $C_1 = m \oplus H(k)$ $C_2=cert_S+d_SH(P_S\|R_S\|C_1\|R)+rH(ID_S\|R_S\|C_1\|R)$ *Output:* $C_{SR} =(R,C_1,C_2)$	*Check:* $C_2P=R_S + H(ID_S\|P_S\|R_S)G_{CCA} + H(P_S\|R_S\|C_1\|R)P_S + H(ID_S\|R_S\|C_1\|R)R$ *If pos:* $k= (d_R + cert_R)R.$ $m = C_1 \oplus H(k)$
Scheme 2: $r \in GF(2^p)^* \rightarrow R=rP$ $k = r(P_R+R_R+H(ID_R\|P_R\|R_R)G_{CA})$ $C_1 = m \oplus H(k)$ $h = H(m\|R\|ID_S\|P_S\|R_S)$ $C_2=r(d_S+cert_S+h)^{-1}$ *Output:* $C_{SR} =(h,C_1,C_2)$	$R' = C_2(P_S + R_S + H(ID_S\|P_S\|R_S)G_{CA} + hP)$ $k= (d_R + cert_R)R$ $m' = C_1 \oplus H(k)$ *Check:* $h = H(m'\|R'\|ID_S\|P_S\|R_S)$ *If pos:* $m'=m$
Scheme 3: $r \in GF(2^p)^* \rightarrow R=rP$ $k = r(P_R+R_R+H(ID_R\|P_R\|R_R)G_{CA})$ $C_1 = m \oplus H(k)$ $h = H(m\|R\|ID_S\|P_S\|R_S)$ $C_2=r - h(d_S+cert_S)$ *Output:* $C_{SR} =(h,C_1,C_2)$	$R' = C_2P + h(P_S + R_S + H(ID_S\|P_S\|R_S)G_{CA})$ $k' = (d_R + cert_R)R$ $m' = C_1 \oplus H(k')$ *Check:* $h = H(m'\|R'\|ID_S\|P_S\|R_S)$ *If pos:* $m'=m$

Fig. 1. Comparison of the explicit certificate based signcryption schemes 1, 2 and 3

4.4 Comparison of the Schemes

Fig. 1 summarizes the different steps in the three different schemes. Based on that, **Table 1**. compares the differences in computational efforts between the three different schemes for the different types of involved operations.

Table 1. Comparison of the number of cryptographic operations in the explicit certificate based signcryption schemes

Operation	**Scheme 1**		**Scheme 2**		**Scheme 3**	
	S	U	S	U	S	U
EC Multiplication	3	5	3	4	3	4
EC Addition	2	3	2	3	2	3
Hash	4	4	3	3	3	3
Field addition	2	1	2	1	2	1
Field multiplication	2	0	1	0	1	0
Field inverse	0	0	1	0	0	0

From the definition of the 3 schemes, we can conclude that they all follow a similar structure. The key derivation k and the encryption of the message, corresponding to the parameter C_1, is exactly the same in the 3 schemes. The main difference is between scheme 1 versus schemes 2 and 3. For scheme 1 the EC point R is part of the signcryption message, while this is in schemes 2 and 3 only a hash value h. Consequently, with respect to communication efficiency, schemes 2 and 3 outperform scheme 1. As a result of this construction, the unsigncryption process in scheme 1 can be split into two separate processes for the decryption and the signature verification, while in schemes 2 and 3 the decryption is required before the signature verification can be finalized. This could allow to dedicate the verification in the unsigncryption to a powerful server. However, this feature comes also with a main global computational cost of one additional EC multiplication during the unsigncryption process for scheme 1.

Finally, the difference in efficiency between schemes 2 and 3 is mainly in the computation for the signature verification, where scheme 3 slightly outperforms scheme 2 as it is only using additions in the field, instead of field multiplications. Consequently, it can be seen that the conclusion on the comparison given in [5], between the schemes [5] and [6] is not correct, as [5] was assumed to outperform [6].

5. Signcryption with ECQV Certificates

5.1 Key management

For the certificate based signcryption schemes relying on ECQV certificates, we need to slightly adapt phases 2 (*SetKeyPair*) and 3(*Certification*). Phase 1 (*Setup*) is still valid. We rename the second phase to the *InitializeKeyPair* phase.

- *InitializeKeyPair*: The input of this function consists of the identity ID_U of the user and *params*. The output, sent to the CA, corresponds with the tuple (ID_U,R_{ID}), where $R_U = r_U P$ containing the random value $r_U \in GF(2^p)^*$ chosen by the user.
- *Certification*: In this function, the CA also chooses a random value $r_{CA} \in GF(2^p)^*$ in order to compute $R_{CA} = r_{CA}P$. Then the certificate $cert_U$ is defined by

$$cert_U = R_{CA} + R_U$$

The value $r = H(cert_U|ID_U)\ r_{CA} + \alpha$ is computed. Both $cert_U$ and r are sent to the user. Based on the received tuple $(r, cert_U)$, the user is able to compute its private key by

$$d_U = H(cert_U|ID_U)\ r_U + r$$

and the corresponding public key equals to $P_U = d_U P$. The key pair (d_U,P_U) is accepted by the user only if the public key P_U also satisfies the following equality

$$P_U = H(cert_U|ID_U)cert_U + G_{CA} \tag{2}$$

This follows from the fact that

$$\begin{aligned} P_U &= d_U P = H(cert_U|ID_U)\ r_U P + rP \\ &= H(cert_U|ID_U)\ r_U P + (\ H(cert_U|ID_U)\ r_{CA} + \alpha)P \\ &= H(cert_U|ID_U)\ (r_U + r_{CA})P + \alpha P = P_U \end{aligned}$$

Consequently, based on the information $(ID_U,cert_U)$, any other user is able to find the public key uniquely bounded to the user with identity ID_U.

$$P_U = H(cert_U|ID_U)cert_U + G_{CA},$$

This computation of the public key only requires one EC addition and one EC multiplication. In addition, no separate value for the public key needs to be sent. A formal proof on the security of this ECQV scheme can be found in [20]. We now show how the three above described certificate based signcryption schemes can be considerably simplified by working with these new implicit certificate based credentials of the user.

5.2 Signcryption and unsigncryption processes

The framework for the three implicit certificate based signcryption schemes is very similar. This framework is first discussed and then the different steps in the three schemes are further detailed in the paragraphs below.

The signcryption algorithm $S_{SR}(.)$ is defined by $C_{SR} = S_{SR}\ (m,ID_S,ID_R,sk_S,pk_S,cert_R,params)$. First, the sender computes the public key of the receiver using ID_R and $cert_R$:

$$P_R = H(cert_R|ID_R)cert_R + G_{CA}$$

Next, each of the three schemes perform some specific steps, explained below.

The unsigncryption scheme $U_{SR}(.)$ is defined by $U_{SR}(C_{SR},ID_S,ID_R,sk_R,cert_S,pk_{R,},params)$. Upon arrival, the receiver first computes the public key of the sender using ID_S and $cert_s$ by

$$P_S = H(cert_S|ID_S)cert_S + G_{CA}$$

Next, each of the three schemes perform again some specific steps, explained below.

5.2.1 Scheme 1

The signcryption $S_{SR}(m,ID_S,ID_R,cert_S,sk_S,pk_S,pk_R,,params)$ with $pk_s=P_S$ and $pk_r=P_R$ consists of the following steps.

- Choose a random value $r \in GF(2^p)^*$, compute $R=rP$.
- $k = rP_R$
- $C_1 = m \oplus H(k)$
- $C_2= d_SH(P_S|R_S|C_1|R)+rH(ID_S|R_S|C_1|R)$

The signcryption algorithm has output $C_{SR} =(R,C_1,C_2)$.

The unsigncryption process, denoted by U_{RS} $(C_{SR},ID_S,ID_R,cert_R,sk_R,pk_S,pk_R,,params)$, with $pk_s=P_S$ and $pk_r=P_R$ consists of the following steps.

- The receiver first checks if
  $C_2P = H(P_S|R_S|C_1|R)P_S + H(ID_S|R_S|C_1|R)R$
- If this check is positive, then the key is defined by the equality:
  $k=d_RR$.

and thus the final message m is derived by

$$m = C_1 \oplus H(k)$$

Otherwise, the output equals to $\perp$.

5.2.2 Scheme 2

The sender performs the following steps in the signcryption process.

- Choose a random value $r \in GF(2^p)^*$, compute $R=rP$.
- $k = rP_R$
- $C_1 = m \oplus H(k)$
- $h = H(m|R|ID_S|P_S|cert_S)$
- $C_2=r(d_S+h)^{-1}$

The signcryption algorithm has output $C_{SR}= (h,C_1,C_2)$.

For the unsigncryption process, the receiver first computes

$$R' = C_2(P_S + hP)$$

Next, also the key $k'=d_RR'$ is derived, resulting in the message $m' = C_1 \oplus H(k')$.
Finally, the signature is verified by checking the equality $h = H(m'|R'|ID_S|P_S|cert_S)$. If so, $m=m'$, otherwise the output equals to $\perp$.

The correctness of the scheme follows from the following derivations

$R' = C_2(P_S + hP)$
$= r(d_S+h)^{-1}(P_S + hP) = rP = R$

5.2.2 Scheme 3

The following steps are performed by the sender.

- Choose a random value $r \in GF(2^p)^*$, compute $R=rP$.
- $k = rP_R$
- $C_1 = m \oplus H(k)$
- $h = H(m|R|ID_S|P_S|cert_S)$
- $C_2=r - hd_S$

The signcryption algorithm has output $C_{SR}= (h,C_1,C_2)$.

For the unsigncryption process, the receiver first computes $R' = C_2P + hP_S$
Next, the key $k'=d_RR'$ is derived and thus $m' = C_1 \oplus H(k')$.
Finally, the signature is verified by checking the equality $h = H(m'|R'|ID_S|P_S|cert_S)$. If so, $m=m'$, otherwise the output equals to $\perp$.

The correctness of the scheme follows from the fact that

$R' =(r - hd_S)P + hP_S = rP = R$

Sender: $S_{SR}(m,ID_S,ID_R,cert_S,sk_S,pk_S,pk_R,params)$	**Receiver:** $U_{RS}(C_{SR},ID_S,ID_R,cert_R,sk_R,pk_S,pk_R,params)$
Scheme 1 $r \in GF(2^p)^* \rightarrow R=rP$ $k = rP_R$ $C_1 = m \oplus H(k)$ $C_2= d_SH(P_S\|R_S\|C_1\|R)+rH(ID_S\|R_S\|C_1\|R)$ *Output:* $C_{SR} =(R,C_1,C_2)$	*Check:* $C_2P=H(P_S\|R_S\|C_1\|R)P_S +$ $H(ID_S\|R_S\|C_1\|R)R$ *If pos:* $k= d_RR.$ $m = C_1 \oplus H(k)$
Scheme 2: $r \in GF(2^p)^* \rightarrow R=rP$ $k = rP_R$ $C_1 = m \oplus H(k)$ $h = H(m\|R\|ID_S\|P_S\|cert_S)$ $C_2=r(d_S+h)^{-1}$ *Output:* $C_{SR} =(h,C_1,C_2)$	$R' = C_2(P_S + hP)$ $k= d_RR$ $m' = C_1 \oplus H(k)$ *Check:* $h = H(m'\|R'\|ID_S\|P_S\|cert_S)$ *If pos:* $m'=m$

Scheme 3:	
$r \in GF(2^p)^* \rightarrow R=rP$	$R' = C_2P + hP_S$
$k = rP_R$	$k' = d_R R$
$C_1 = m \oplus H(k)$	$m' = C_1 \oplus H(k')$
$h = H(m\|R\|ID_S\|P_S\|cert_S)$	$Check: h = H(m'\|R'\|ID_S\|P_S\|cert_S)$
$C_2 = r - hd_S$	$If\ pos: m'=m$
$Output: C_{SR} = (h, C_1, C_2)$	

Fig. 2. Comparison of the implicit certificate based signcryption schemes 1, 2 and 3

5.3 Comparison of both types of certificate based signcryption schemes

Fig 2. summarizes the different steps in the implicit certificate based signcryption schemes, where the public keys of the other entity are considered to be computed offline. A complete analysis of the operations in the three different signcryption schemes using ECQV implicit certificates is summarized in **Table 2**. It is reasonable to assume that the public keys are already computed in advance, as it can happen for instance through the computation by a separated and dedicated server, having strong computational capacity.

Table 2. Comparison of the number of cryptographic operations in the implicit certificate based signcryption schemes (public keys are computed offline)

Operation	**Scheme 1**		**Scheme 2**		**Scheme 3**	
	S	U	S	U	S	U
EC Multiplication	2	4	2	3	2	3
EC Addition	0	1	0	1	0	1
Hash	3	3	2	2	2	2
Field addition	1	0	1	0	1	0
Field multiplication	2	0	1	0	1	0
Field inverse	0	0	1	0	0	0

The main difference between the two approaches, using explicit and implicit based certificates, is that the link between the public key and the identity of the user is checked beforehand in the ECQV implicit certificates based approach, while in the traditional certificate based approach this is incorporated in the actual signcryption and unsigncryption processes. The incorporation is both at the encryption/decryption phase in the derivation of its key k and the signature definition/verification step. The length of the user's credentials is shorter in the ECQV implicit certificates schemes as the implicit certificate is used to derive the public key, while in the traditional certificate based schemes the public key consists of one additional certificate related parameter.

To conclude, when comparing **Table 1** and **2** for the most compute intensive operations, the EC multiplication and EC addition, the difference between both approaches in the signcryption and unsigncryption phase for the three signcryption schemes is in all cases equal to two EC additions and one EC multiplication.

When we also include the complexity of computing the public keys in the case of the ECQV implicit certificates (see Equation 2), the difference between both approaches is only one EC addition.

Fig. 3 visualizes the difference in performance expressed in timing (μs) between the different schemes, explicit and implicit, for signcryption and unsigncryption respectively. The performance numbers are based on the results obtained from [21], where the cryptographic operations are implemented on a personal computer with a 2.50 GHz CPU and 8 GByte RAM and the Windows 7 OS. Overall, there is a 30% improvement when considering the implicit versus the explicit approach. The difference between performance of the signcryption scheme among the 3 schemes is negligible, however the unsigncryption schemes 2 and 3 are almost 25% more efficient than Scheme 1.

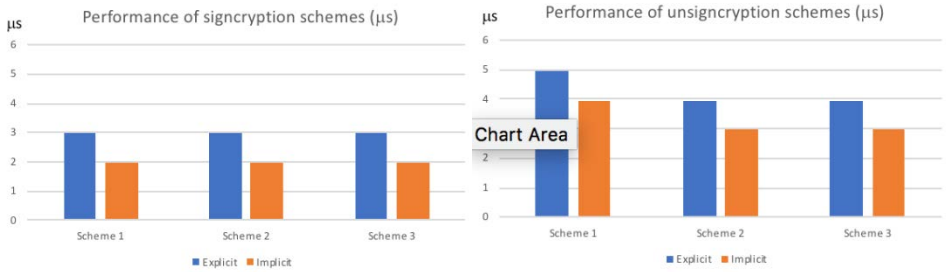


Fig. 3. Comparison of the performance (time in μs) for schemes 1, 2 and 3. The left hand side represents the signcryption (sender side) and the right side represents the unsigncryption (receiver side)

6. Extensions

We now show how the above described schemes can be easily transformed in a GSC scheme and how the anonymity of sender and receiver is obtained in the three schemes.

6.1 Generalized signcryption

There are 3 main scenarios in a GSC scheme. Note that due to the difference in role of the certificate for the traditional certificate based schemes and the certificate based schemes with ECQV implicit certificates, the input parameters of the signcryption and unsigncryption algorithms differ for both. **Table 3** summarizes the differences for the input parameters in both types of certificate based schemes.

Table 3. Differences in input parameters for the traditional certificate based and ECQV implicit certificate based signcryption schemes

	Traditionnal	**ECQV based**
Signcryption	$cert_S$ $pk_R = (P_R, P_R)$	0 $cert_R$
Unsigncryption	$cert_R$ $pk_S = (P_S, P_S)$	0 $cert_S$

The notations below are for the traditional certificate based schemes. Using **Table 3**, the conversion for the ECQV implicit certificates based schemes can be made.

- Signcryption scenario: sender and receiver are determined and the message is encrypted and provided with a signature.

$$C_{SR} = S_{SR}(m, ID_S, ID_R, cert_S, sk_S, pk_S, pk_{R,}, params)$$
$$m' \text{ or } \perp = U_{RS}(C_{SR}, ID_S, ID_R, cert_R, sk_R, pk_S, pk_{R,}, params)$$

- Signature scenario: Only the sender is determined. We denote an unknown receiver by $R=\emptyset$. The message is only provided with a signature and no encryption is performed.

$$C_{S\emptyset} = S_{S\emptyset}(m, ID_S, 0, cert_S, sk_S, pk_S, 0, params)$$
$$m' \text{ or } \perp = U_{\emptyset S}(C_{S\emptyset}, ID_S, 0, c, 0, pk_S, 0, params)$$

- Encryption scenario: Only the receiver is determined. The message is encrypted and signed by an anonymous sender and thus we denote $S=\emptyset$.

$$C_{\emptyset R} = S_{\emptyset R}(m, 0, ID_R, 0, 0, pk_{R,}, params)$$
$$m' \text{ or } \perp = U_{R\emptyset}(C_{\emptyset R}, 0, ID_R, sk_R, 0, pk_{R,}, params)$$

Inserting 0 as indicated by the input parameters of the algorithms in the signature and encryption scenarios and assuming $H(0)=0$, transforms each of the 6 previous described signcryption schemes to a GSC scheme.

6.2 Anonymous signcryption

In all of the above described schemes, the anonymity of the receiver is obtained due to the ECDLP. In Scheme 2 and 3 for both types of certificate based mechanisms, the identity of the sender is also hidden for any outsider, since the hash value to be verified in the signature includes the original message that can only be derived by the intended receiver. However, in Scheme 1, the verification is done solely based on the received message R, C_1, C_2 and the public key and identity of the sender. Consequently, in the assumption that the public keys of all users in the system are known to everybody, the verification of the signature leads to the corresponding sender. However, by multiplying C_2 with $H(k)$, a hiding factor is included, which can only be verified by the intended receiver. Note that this operation only requires one additional field multiplication during both the signcryption and unsigncryption phase.

As a consequence, the proposed schemes represent the most efficient anonymous signcryption schemes. In order to do a fair comparison with the most efficient one in the state of the art, ASEC [16], we consider the certificate based schemes using the ECQV implicit certificates with the assumption that the public keys are generated offline. Note that in ASEC also the assumption has been made that the validity of the public key is obtained offline, by the storage in a third party protected environment. In the ASEC scheme, 3 EC point multiplications during signcryption and 5 EC point multiplications and 2 EC additions during unsigncryption are required. Consequently, as can be seen from Table 3, the three proposed certificate based signcryption schemes drastically outperform ASEC.

7. Security analysis

The security analysis is based on the proof by contradiction, similar like in [22]. In order to formally define the ECDLP as expressed in [23], we need to consider the following two distributions

$D_{real}=\{r \in GF(2^p)^*, R=rP : (P,R,r)\}$
$D_{rand}=\{r,k \in GF(2^p)^*, R=rP: (P,R,k)\}$

The advantage of any probabilistic, polynomial-time, 0/1-valued distinguisher D in solving ECDLP on $E_p(a,b)$ is defined as

$$\mathrm{Adv}_{D,Ep(a,b)}^{ECDLP}=|Pr((P,R,r) \in D_{real}:D(P,R,r)=1) - Pr(P,R,k) \in D_{rand} : D(P,R,r)=1) |$$

where the probability $Pr(.)$ is taken over random choices of r and k. The distinguisher D is said to be a (t,ε)- ECDLP distinguisher for $E_p(a,b)$ if D runs at most in time t such that $\mathrm{Adv}_{D,Ep(a,b)}^{ECDLP} \geq \varepsilon$. The following assumption holds.

ECDLP assumption: For every probabilistic, polynomial-time, 0/1-valued distinguisher D, we assume that $\mathrm{Adv}_{D,Ep(a,b)}^{ECDLP} < \varepsilon$, for any sufficiently small $\varepsilon > 0$.

Consequently, no (t,ε)- ECDLP distinguisher for $E_p(a,b)$ exists. There are two types of adversaries to be considered. An adversary of type I is an outsider or certified user and an adversary of type II is assumed to possess the master key α of the CA.

Theorem 1: *The proposed explicit and implicit certificate based signcryption schemes are provably secure against both types of adversaries under the ECDLP assumption.*

Proof: We will describe the proof for the third type of signcryption scheme relying on the ECQV implicit certificates. The proof is similar for the other variants. Let us assume that an adversary is able to solve the ECDLP and thus can find the value r from the points P and $R=rP$ of $E_p(a,b)$. The following oracle is now defined.

Reveal: The output of the query corresponds with the value r through the solution of ECDLP by using the points P and $R=rP$ of $E_p(a,b)$.

The adversary A executes then two algorithms, Algorithm 1 (Alg1) and Algorithm 2 (Alg2), for the proposed signcryption scheme SC. We now define $Succ1_{SC,A}^{ECDLP} = Pr(Alg1 = 1)-1$, similar as in [23]. Then, the advantage function for Algorithm 1 is defined as

$$Adv1_{SC,A}^{ECDLP}(t,q_R)=max_A\{ Succ1_{SC,A}^{ECDLP} \},$$

where the maximum is taken over all A with execution time t and q_R is the number of queries to the Reveal oracle. The proposed SC is said to provide confidentiality if $Adv1_{SC,A}^{ECDLP}(t,q_R) < \varepsilon$, for any sufficiently small $\varepsilon>0$.

We also define $Succ2_{SC,A}^{ECDLP} = Pr(Alg2 = 1)-1$, similar as in [24]. Then, the advantage function for Algorithm 2 is defined as

$$Adv2_{SC,A}^{ECDLP}(t,q_R)=max_A\{ Succ2_{SC,A}^{ECDLP} \},$$

where the maximum is taken over all A with execution time t and q_R is the number of queries to the Reveal oracle. The proposed SC is said to provide the security features authentication, integrity, unforgeability, and forward secrecy if $Adv2_{SC,A}^{ECDLP}(t,q_R)< \varepsilon$, for any sufficiently small $\varepsilon >0$. Both algorithms are defined as follows.

Algorithm 1
Capture output of SC: (h,C_1,C_2)
Compute $R = C_2P + hP_S$
Call *Reveal* oracle. Output $r=Reveal(E_p(a,b),P,R)$
Use value r, compute $k=rP_R$
Retrieve message $m = C_1 \oplus H(k)$

Algorithm 2
Capture output of SC: (h,C_1,C_2)
Compute $R = C_2P + hP_S$
Call *Reveal* oracle. Output $r=Reveal(E_p(a,b),P,R)$
Use value r, compute $k=rP_R$
Retrieve message $m = C_1 \oplus H(k)$
Change m to m'
Compute $C_1' = m \oplus H(k)$
Compute $h' = H(m'\|R\|ID_S\|P_S\|cert_S)$
If *Type I* adversary: Call *Reveal* oracle. Output $d_S=Reveal(E_p(a,b),P,P_S)$
If *Type II* adversary: Call *Reveal* oracle. Output $d_S=Reveal(E_p(a,b),P,R_S)$
Compute $C_2'= r - h'd_S$
Send (h',C_1',C_2') to verifier
Verifier computes $R'= C_2'P + h'P_S$
Verifier checks if $H(m'\|R'\|ID_S\|P_S\|cert_S)=h'$
If the verification is successful then return 1
else return 0

Based on the definitions and notations described above, we now show that the proposed signcryption scheme satisfies confidentiality, authentication, unforgeability and forward secrecy.

Confidentiality: When following the steps as defined in Algorithm 1, the value r can be computed by the adversary using the point R. As a consequence, the adversary is able to derive the secret key k and to decrypt the ciphertext.
However, due to the computational difficulty of the ECDLP, it is impossible for the attacker to derive r and thus the $Adv1_{SC,A}^{ECDLP}(t,q_R)< \varepsilon$, for any sufficiently small $\varepsilon >0$. Consequently, the attacker will also not be able to find the key and to decrypt the ciphertext. The confidentiality of the protocol is thus guaranteed.

Authentication: When following the steps as defined in Algorithm 2, the values r and d_S can be derived by the adversary. In this way, the values of m, C_1 and C_2 can be modified by the

adversary. Again, due to the computational difficulty of the ECDLP, it is impossible for the attacker to derive r and thus the $Adv2_{SC,A}^{ECDLP}(t,q_R)< \varepsilon$, for any sufficiently small $\varepsilon >0$. Consequently, without being able to modify m, C_1 and C_2, authentication is guaranteed and attacks like man-in-the-middle and replay are avoided.

Unforgeability: In order to forge the message (h,C_1,C_2) from the SC algorithm, the adversary need to be in possession of both the private key of the sender d_S and the random value r. Due to the computational difficulty of the ECDLP, $Adv2_{SC,A}^{ECDLP}(t,q_R)< \varepsilon$, for any sufficiently small $\varepsilon >0$, this is not possible and consequently unforgeability is guaranteed.

Forward secrecy: In order to offer forward secrecy, the adversary should not be able to recover the messages sent in previous SC rounds, even if the adversary obtains afterwards the knowledge of the private key of the sender d_S. This feature is valid in the proposed SC scheme, as the secret key is based on the usage of a random value r, which cannot be derived without being able to solve the ECDKP.

7. Conclusion

In case there is no secure channel between the user and the CA, only the certificate based approach is able to offer identity authentication. We focus in this paper on certificate based signcryption schemes solely using EC operations and no pairings. We show how the recently proposed traditional certificate based signcryption schemes (also called explicit certificate based) can be improved by using ECQV implicit certificates. In particular, the usage of the ECQV implicit certificates allows an improvement of the complexity of the signature schemes with one EC addition. Moreover, when the validities of the public keys of the receiver and sender are checked offline or through a separate and dedicated server, both the signcryption and unsigncryption processes even further outperform with one EC addition and one EC multiplication. Finally, we show that these schemes can also be applied as GSC schemes and that anonymity is already inherently involved in the proposed schemes or can be easily added without significant cost.

Consequently, in many application areas where pairing based protocols are used (eg. cloud computing, voting, payment, etc.), the proposed algorithms together with their underlying identity based approaches will often lead to a significantly more efficient system. To conclude, this paper describes certificate based signcryption schemes, which can be used as building blocks in many protocols establishing privacy and authentication for constrained environments.

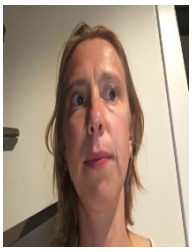
References

[1] A. Shamir, “Identity-Based Cryptosystems and Signature Schemes,” *Adv. cryptology*, vol. 196, pp. 47–53, 1984. Article (CrossRef Link)

[2] S.S. Al-Riyami and K.G. Paterson, “Certificateless Public Key Cryptography,” in *Proc. of Int. Conf. Theory Appl. Cryptology Inform., Security*, Taipei, Taiwan, pp. 452–473, 2003. Article (CrossRef Link)

[3] C. Gentry, “Certificate-Based Encryption and the Certificate Revocation Problem,” *Int. Conf. Theory Appl. Cryptographic Techn.*, pp. 272–293, 2003. Article (CrossRef Link)

[4] Y. Zheng, "Digital Signcryption or How to Achieve Cost (Signature & Encryption) ≪ Cost (Signature) + Cost (Encryption)," *Annu. Int. Cryptolofy Conf.*, pp. 165–179, 1997. Article (CrossRef Link)
[5] Minh-Ha Le and Seong Oun Hwang, "Certificate-Based Signcryption Scheme without Pairing: Directly Verifying Signcrypted Messages Using a Public Key," *ETRI Journal*, vol. 38, no. 4, pp. 724-734, 2016. Article (CrossRef Link)
[6] Y. Lu and J. Li, "Provably Secure Certificate Based Signcryption Scheme without Pairings," *KSII Transactions on Internet and Information Systems*, vol.8, no. 7, pp. 2554-2571, 2014. Article (CrossRef Link)
[7] A. Braeken and P. Porambage, "Efficient Generalized Signcryption Scheme based on ECC," *Int. Journal on Cryptography and Information Security (IJCIS)*, vol. 5, no. 2, 2015.
[8] J. Malone-Lee, "Identity based signcryption," *Cryptology ePrintArchive*, 2002. http://eprint. iacr.org/2002/098.pdf.
[9] Piotr Szczechowiak, Leonardo B. Oliveira, Michael Scott, Martin Collier, and Ricardo Dahab, "NanoECC: Testing the Limits of Elliptic Curve Cryptography in Sensor Networks," in *Proc. of European conference on Wireless Sensor Networks (EWSN'08)*, 2008. Article (CrossRef Link)
[10] M. Barbosa and P. Farshim, "Certificateless signcryption," in *Proc. of ACM Symposium on Information, Computer and Communications Security*, pp. 369-372, 2008. Article (CrossRef Link)
[11] S.S.D. Selvi, S.S. Vivek, D. Shukla, P.R. Chandrasekaran "Efficient and Provably Secure Certificateless Multi-receiver Signcryption," *Int. Conf. ProvSec*, pp. 52–67, 2008. Article (CrossRef Link)
[12] F. Li, X. Xin, and Y. Hu, "Efficient Certificate-Based Signcryption Scheme from Bilinear Pairings," *Int. J. Comput. Appl.*, vol. 30, no. 2, pp. 129–133, 2008. Article (CrossRef Link)
[13] M. Luo, Y. Wen, and H. Zhao, "A Certificate-Based Signcryption Scheme," *Int. Conf. Comput. Sci. Inform. Technol.*, pp. 17–23, 2008. Article (CrossRef Link)
[14] J. Li, X. Huang, M. Honga, Y. Zhang, "Certificate-Based Signcryption with Enhanced Security Features," *Comput. Math. Appl.*, vol. 64, no. 6, pp. 1587–1601, 2012. Article (CrossRef Link)
[15] C.P. Schnorr, "Efficient identification and signatures for smart cards," *in Proceedings of the Cryptology*, LNCS, vol. 435, pp. 239–251, 1990. Article (CrossRef Link)
[16] A.K. Singh, "A Review of Elliptic Curve based Signcryption Schemes," *Int. Journal of Computer Applications*, vol. 102, no. 6, 2014. Article (CrossRef Link)
[17] A. Braeken and P. Porambage, "ASEC: Anonym Signcryption Scheme Based on EC operations", International Journal of Computer Applications, vol. 5, no. 7, pp. 90-96, 2015.
[18] Certicom Research 2013, SEC4: Elliptic Curve Qu-Vanstone Implicit Certificate Scheme, Standards for Efficient Cryptography Group, Version 1.0 (Jan 2013).
[19] D. Hankerson, A. J. Menezes, and S. Vanstone, "Guide to Elliptic Curve Cryptography", ISBN: 038795273X, Springer-Verlag New York, Inc., 2003.
[20] D.R. Brown, R. Gallant, and S.A. Vanstone, Provably Secure Implicit Certificate Schemes, In Financial Cryptography, pp. 156-165, Springer, 2001. Article (CrossRef Link)
[21] D. He, S. Zeadally, H. Wang, Q. Liu, "Lightweight data Aggregation Scheme against Internal Attackers in Smart Grid Using Elliptic Curve Cryptography," *Wireless Communications and Mobile Computing*, vol. 2017, 11 pages, 2017. Article (CrossRef Link)
[22] Y.H. Chuang, Y.M. Tseng, "An efficient dynamic group key agreement protocol for imbalanced wireless networks," *International Journal of Network Management*, vol. 20, no. 4, pp. 167–180, 2010. Article (CrossRef Link)
[23] R. Dutta, R. Barua, "Provably Secure Constant Round Contributory Group Key Agreement," *IEEE Transactions on Information Theory*, vol. 54, no. 5, pp. 2007–2025, 2008. Article (CrossRef Link)
[24] J. Baek, R. Steinfeld, Y. Zheng, "Formal Proofs for the Security of Signcryption," *Journal of Cryptology*, vol. 20, no. 2, pp. 203–235, 2007. Article (CrossRef Link)



An Braeken obtained her MSc Degree in Mathematics from the University of Gent in 2002. In 2006, she received her PhD in engineering sciences from the KULeuven at the research group COSIC (Computer Security and Industrial Cryptography). She became professor in 2007 at the Erasmushogeschool Brussel (currently since 2013, Vrije Universiteit Brussel) in the Industrial Sciences Department. Prior to joining the Erasmushogeschool Brussel, she worked for almost 2 years at the management consulting company Boston Consulting Group (BCG). Her current interests include cryptography, security protocols for sensor networks, secure and private localization techniques, and FPGA implementations. She is (co-)author of over 110 publications. She has been member of the program committee for numerous conferences and workshops (ACCSE 2017**,** Sensorcomm 2017, HPCS2017, SecureEdge 2017, CloudTech 2017, CSICT 2017, ICAT2E 2017) and been member of the editorial board for Security and Communications journal. She is regularly reviewing papers for multiple journals, as the Journal of Applied Sciences, Sensors, Concurrency and Computation,… In addition, she is since 2015 reviewer for several EU proposals and ongoing projects, submitted under the programs of H2020, Marie Curie and ITN.
She has cooperated and coordinated more than 12 national and international projects. She has been STSM manager in the COST AAPELE project (2014-2017) and is currently in the management committee of the COST RECODIS project (2016-2019).