cryptography

MDPI

Article

I2PA: An Efficient ABC for IoT

Ibou Sene [1,2,*], Abdoul Aziz Ciss [1] and Oumar Niang [1]

[1] Laboratoire de Traitement de l'Information et des Systèmes Intelligents, Ecole Polytechnique de Thiès, P.O. Box A10 Thiès, Senegal; aaciss@ept.sn (A.A.C.); oniang@ept.sn (O.N.)

[2] Ecole Doctorale Développement Durable et Société, Université de Thiès, P.O. Box 967 Thiès, Senegal

* Correspondence: senei@ept.sn

Received: 9 May 2019; Accepted: 18 June 2019; Published: 21 June 2019

Abstract: The Internet of Things (IoT) is very attractive because of its promises. However, it brings many challenges, mainly issues about privacy preservation and lightweight cryptography. Many schemes have been designed so far but none of them simultaneously takes into account these aspects. In this paper, we propose an efficient attribute-based credential scheme for IoT devices. We use elliptic curve cryptography without pairing, blind signing, and zero-knowledge proof. Our scheme supports block signing, selective disclosure, and randomization. It provides data minimization and transaction unlinkability. Our construction is efficient since smaller key size can be used, and computing time can be reduced. As a result, it is a suitable solution for IoT devices characterized by three major constraints, namely low-energy power, small storage capacity, and low computing power.

Keywords: IoT; credential; privacy; anonymity; ECC; ZKP; blind signing; selective disclosure; randomization

1. Introduction

The Internet has changed our way of living. In fact, it has become an integral part of our life. As a ubiquitous communication platform, it has undergone remarkable development in recent years. The concept of the Internet of Things (IoT) has emerged, and envisages the integration of all real-world objects into the Internet. The evolution of the IoT concept has given rise to other concepts such as IoE (Internet of Everything) or IoV(Internet of Vulnerabilities). Some people even talk about IoP [1] (Internet of People). According to Cisco forecasts [2], there will be 50 billion connected devices by 2020.

In most cases, communications between devices require authentication itself based on authentication factors [3]. Authentication is also based on identification, which makes activities of an entity traceable since each device is directly or indirectly associated with its owner. The current infrastructure is based on centralized architecture, which allows no control of data by their owners; this is a threat to privacy preservation. Security in current infrastructures is guaranteed in most cases by Public Key Infrastructures(PKIs) that can guarantee that messages are not compromised, and only recipients are able to open and read them (integrity, confidentiality, and authenticity are guaranteed). PKIs' main objective is to guarantee key encryption authenticity. However, they cannot protect users' privacy and users cannot get a credential on a subset of their attributes without letting the certification authority (CA) see the resulting credential. Moreover, when authenticating to a service provider, users must show their whole credentials instead of just proving their eligibility for that access. We must therefore think to migrate to decentralized and user-centric architecture.

The ultimate challenge can be summarized in this fundamental question: How is it possible to minimize data disclosed about oneself and provide only the bare minimum necessary? Presently, to the best of our knowledge, the most convenient way to protect users' privacy remains using Anonymous Credentials systems also known as Attribute-Based Credentials (ABCs). ABCs are building blocks for user-centric identity management [4]. With ABCs, it is possible to get a signature on a set of attributes and then use this later to access other services. Seeing the abstractness of some attributes (nationality, age class, occupation, affiliation, profession, etc.), the entire mechanism can remain anonymous and therefore guarantees the privacy of concerned entities'. Authentication is carried out by revealing the bare minimum necessary. Better, it is possible, for a set of attributes, to prove their possession instead of revealing their values, from where derives the concept I2PA, meaning "I Prove Possession of Attributes" .

In terms of anonymous credentials, there are two flagship schemes—Idemix of IBM [5] and U-prove of Microsoft [6]. There are many other contributions [7–12]; however, as discussed in Section 2, when tackling privacy concerns, none of those schemes is fully adapted in an IoT environment characterized by three major constraints, namely low computing power, very limited storage capacity, and low energy autonomy.

The relevance of our contribution lies in the fact that it offers a good level of security and drastically reduces key size. Credential randomization, selective disclosure, and unlinkability features are very interesting results that contribute in guaranteeing non-traceability. Edwards curves, known as the curves in which cryptographic calculations are faster [13], are privileged. We then win in memory usage, performance in computing time, and bandwidth usage. Analysis presented in Section 7 and performance evaluation conducted in Section 8 show that our scheme is very efficient. Its level of abstractness makes it applicable in any IoT environment.

The rest of this paper is organized as follows. We start by discussing related works in Section 2, review mathematic backgrounds in Section 3, while definitions of some leading concepts are presented in Section 4. Section 5 presents an architecture of ABC systems and in Section 6, our contribution is presented. Section 7 is related to complexity analysis and performance evaluation is presented in Section 8. This paper is ended by a conclusion and perspectives in Section 9.

2. Related Works

Credentials are essential in identity-management systems. They attest that an entity has a certain knowledge, skill, characteristic, etc. [14]. ABCs involve attributes of an entity without including identity information, which allows linking the credential to its owner [14–16]. They allow a user to prove properties about herself anonymously [11,15,17]. Even when attributes are hidden, the verifier can still assess the validity of the credential [15]. There are two major families of ABC systems, namely those based on blind signatures (BS) and those based on zero-knowledge proofs (ZKP). In terms of anonymous credential, there are two leading schemes—Idemix of IBM [5] and U-prove of Microsoft [6]. While Idemix's building block [5] is based on Camenish—Lysyanskaya signature scheme [18] (CL-Signature), U-prove is based on Stefan Brand's digital signature [19] instead. There are many other contributions including IRMA of Radboud University of Nijmegen [12] (based on Idemix). However, when tackling privacy concerns, none of those schemes is fully adapted in an IoT environment characterized by three major constraints, namely low computing power, very limited storage capacity, and low energy autonomy. In fact, these models are based either on RSA cryptosystems [7] or on pairing-based cryptography [20]. Some of them are based on elliptic curve cryptography (ECC) without pairing (U-prove, for instance) but do not fully take into account some fundamental features relative to privacy preserving. Indeed, the unlinkability feature is not fully taken into account by U-prove, and can only be achieved by using different credentials [16,21,22]. With reference to RSA-based cryptosystems, the major problem remains key size, which will necessarily give rise to problems of storage, performance in computing time, and bandwidth usage. Pairing-based cryptosystems, although considered to be very robust, are not applicable in an IoT context given that they are too greedy in terms of computing time. The relative computation cost of a pairing is approximately 20 times higher than that of the scalar multiplication. The model presented by Gergely et al. [8] is very efficient in a sensor network but does not guarantee anonymity since the verifier has a database of identifiers. Persiano et al. [11] proposed a scheme for multi-show non-transferable credentials. However, authors in [16] shown that this scheme does not support advanced issuance or carry-over attributes, and de-anonymization is partially supported. De Fuentes et al. [14] conducted an assessment of ABCs in smart cities context considering Idemix, U-Prove, and VANET-updated Persiano systems. Their experimental results show that Idemix is the most promising approach both in terms of performance and the set of smart city road traffic services that could adopt it. Dzurenda et al. [23] proposed ecHM12, a new ABC scheme based on EC and HM12 [24] scheme. They assert that their scheme meets all standard requirements on ABC schemes. However, they do not deal with credentials with more than three attributes and the time of their verification protocol depends on the number of revoked users in the system. They also claim that their solution has good impact on bandwidth whereas 12 variables are exchanged between the user and the verifier in the proving protocol. Furthermore, their proving protocol is very greedy in terms of computing time; 19 scalar multiplications and eight additions of points are carried out over the curve. Their revocation process also requires linear time in the number of users. They assume that current servers have high computing power to guarantee usability of their scheme. As a result, this scheme may not be applicable in a system with lots of users and where devices have low computing power. Even if Idemix is the most advanced in terms of implementation, it is subject to improvement given the key size required. Sinha et al. [9] show that at equal security level, an RSA key is at least six times longer than an ECC key. Furthermore, in most cases, operations are faster on ECC than in RSA cryptosystem. Another important point is that in an RSA cryptosystem, when key size is no longer enough to guarantee a secure level, it is recommended to double it, which is not necessarily the case for ECC cryptosystem. Another important contribution [10] is built on top of Idemix. It has been successfully implemented, deployed, and tested. However, criticisms of Idemix remain valid toward it. U-prove, by its generality, can be implemented either based on subgroup or using ECC. However, as mentioned earlier, the unlinkability feature requires additional storage space.

3. Mathematic Background

ECC was presented independently by Koblitz [25] and Miller [26] in the 1980s. Their structure of group and performance in computing time make them a new direction in cryptography [27,28]. ECC has been proven to be one of the best public key cryptosystem (PKC) alternatives for constrained applications [29–31]. The following section is a brief description of ECC.

3.1. Definition

An elliptic curve **E** over a field $\mathbb{K}$ can be described as a set of $\mathbb{K} \times \mathbb{K}$ satisfying the equation:

$$\mathbf{E} : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \tag{1}$$

where $a_i \in \mathbb{K}$ to which we add a point at infinity $\mathcal{O}$, defined as the intersection of all vertical lines. An additional requirement is that the curve must be "smooth" [3,32]. Depending on the characteristic of $\mathbb{K}$, the equation below can be simplified [3,32]:

1. When $char(\mathbb{K}) \neq 2, 3$ the equation can be simplified to $y^2 = x^3 + ax + b,$ where a, b $\in \mathbb{K}$.
2. When $char(\mathbb{K}) = 2$ and $a_1 \neq 0$, the equation can be simplified to $y^2 + xy = x^3 + ax^2 + b$, where a, b $\in \mathbb{K}$. This curve is said to be non-supersingular. If $a_1 = 0$, the equation can be simplified to $y^2 + cy = x^3 + ax + b$, where a, b, c $\in \mathbb{K}$. This curve is said to be supersingular.
3. When $char(\mathbb{K}) = 3$ and $a_1^2 \neq -a_2$ the equation can be simplified to $y^2 = x^3 + ax^2 + b$, where a, b $\in \mathbb{K}$. This curve is said to be non-supersingular. If $a_1^2 = -a_2$, the equation can be simplified to $y^2 = x^3 + ax + b,\ a, b \in \mathbb{K}$. This curve is said to be supersingular.

The "Figure 1" is an illustration of the curve $y^2 = x^3 - x$ over $\mathbb{R}$.

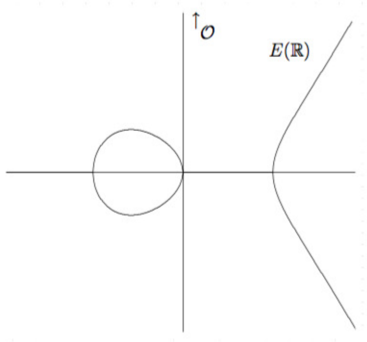


Figure 1. An Elliptic Curve of equation $y^2 = x^3 - x$ over $\mathbb{R}$.

Theorem 1 (Hasse's theorem). *Let E be an elliptic curve defined over a finite field $\mathbb{K}$ with q elements, then the result of Hasse states that: $|\#E(\mathbb{K}) - q - 1| \leq 2\sqrt{q}$.*

3.2. Edwards' Curves

Edwards, generalizing an example from Euler and Gauss, introduced an addition law for the curves $x^2 + y^2 = c^2(1 + x^2y^2)$ over a non-binary field $\mathbb{K}$. He showed that every elliptic curve over a non-binary field $\mathbb{K}$ can be expressed in the form $x^2 + y^2 = c^2(1 + x^2y^2)$ if $\mathbb{K}$ is algebraically closed. Bernstein and Lange [33] generalized the addition law of the curves $x^2 + y^2 = c^2(1 + dx^2y^2)$. Let $char(\mathbb{K}) \neq 2, 3$ and let $E(\mathbb{K})$ has a unique point of order 2. Then, E can be written in Edwards form [33–35]:

$$E_d : x^2 + y^2 = 1 + dx^2y^2, where\ d \notin \{0, 1\} \tag{2}$$

Let $P_1(x_1, y_1)$, $P_2(x_2, y_2)$, and $P_3(x_3, y_3)$ be points of the curve such that $P_3 = P_1 \bigoplus P_2$. The Edwards addition law over E_d is described as follow:

$$(x_3, y_3) = \left(\frac{x_1y_2 + x_2y_1}{1 + dx_1x_2y_1y_2}, \frac{y_1y_2 - x_1x_2}{1 - dx_1x_2y_1y_2} \right) \tag{3}$$

This group law is complete, strongly uniform and presents interesting results:

- If d is a non-square in $\mathbb{K}$, the addition law is complete. This ensures that denominators are never zero.
- The addition law is strongly unified, i.e., it can also be used for doubling.
- The point (0,1) is the neutral element.
- The point (0,−1) has order 2.
- The points (±1,0) have order 4.
- The inverse of (x, y) is $(-x, y)$.

In addition, Edwards curve is known to be the fastest among all families of elliptic curves used to implement cryptography.

3.3. Elliptic Curve Discrete Logarithm Problem (ECDLP)

The basic operations of ECC are point scalar computations (also known as scalar multiplication) of the form:

$$Q = k.P = \underbrace{P + P + \ldots + P}_{\text{k times}}$$

The ECDLP is the problem of retrieving k given P and Q, where P and Q are points of the curve and k is an integer uniformly chosen at random from [1, r−1] (r denotes the order of P). The assumed difficulty of this problem is the basis of security in elliptic curve public key cryptosystems. Point scalar multiplication can be performed efficiently using algorithms such as double-and-add.

4. Definition

The aim of this section is to define some leading terms in this paper, namely attribute, credential, blind signing, and zero-knowledge proof.

4.1. Attribute

An attribute is a characteristic or a qualification of a person. It can either be an identifying or non-identifying property. For example, "full name", "address", and "social security number" are identifying attributes. Attributes such as "is a student" and "is a teenager" are non-identifying as they do not uniquely identify a person; such properties can belong to other people as well.

4.2. Credential

A credential is a set of attributes digitally signed by a trusted third party (issuer), i.e., a set of attributes together with the corresponding cryptographic information. A credential is similar to a certificate in terms of content, but they are different in the way they are used. While certificate usage requires showing all its content, a credential can be used by showing some parts and hiding or proving knowledge of others. Credentials are important in identity-management systems. They certify that an entity has certain characteristics, knowledges, skills, etc. One main point is that credentials involve attributes of an entity without including identity information, which allows linking of the credential to its owner. As illustrated in the "Figure 2", a credential has three main parts: a secret key of its owner, a set of attributes, and a signature of an issuer on that set. A credential includes more information such as expiry date.

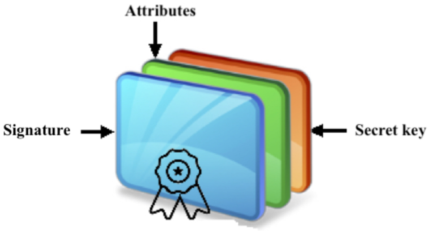


Figure 2. Credential's structure.

4.3. Zero-Knowledge Proofs

Recent decades have witnessed the emergence of several new cryptographic notions. In 1985, Goldwasser, Micali and Rackoff [36] introduced the concept of zero-knowledge interactive proofs that enables an entity, a prover, to convince another entity, a verifier, of the validity of a statement without revealing anything else beyond the assertion of this statement. Zero-knowledge proofs are elegant techniques to limit the amount of information transferred from a prover to a verifier in a cryptographic protocol. The "Table 1" describes Schnorr's proof of knowledge also known as Schnorr's Identification protocol. Given a group G of prime order q, in which the discrete logarithm problem (DLP) is hard, and a generator g ($\langle g \rangle = G$), a prover proves to a verifier that he knows a secret value x, uniformly chosen at random from $\mathbb{Z}_q^*$, corresponding to a public value $h = g^x \in G$.

Table 1. Schnorr's proof of knowledge.

Prover	**Public**	**Verifier**
Secret x	$q, g, h = g^x$	
$w \in_R \mathbb{Z}_q^*$ $a = g^w$ in G	$\xrightarrow{a}$	
	$\xleftarrow{c}$	$c \in_R \mathbb{Z}_q$
$r = cx + w(mod\ q)$	$\xrightarrow{r}$	$a \overset{?}{=} g^r.h^{-c}$ in G

There is a formal symbolic notation of ZKP as described below:

$$PK\{(\alpha) : h = g^\alpha\}$$

4.4. Blind Signing

In many applications involving anonymity, it is often desirable to allow a participant to sign a document without knowing its content; this is known as blind signature. It is typically used in privacy-related protocols where the signer and credential owner are different parties. It can be used in e-cash or a privacy vote.

4.5. Blindness

Let U_0 and U_1 be two honest users and A be a Probabilistic Polynomial-Time (PPT) adversary which plays the role of the signer engaged in the issuing scheme with U_0 and U_1 on messages m_b and m_{1-b} where b is chosen uniformly at random from $\{0, 1\}$. U_0 and U_1 output the signatures σ_b and σ_{1-b}. After that, $(m_b, m_{1-b}, \sigma_b, \sigma_{1-b})$ is sent to A which outputs $b' \in \{0, 1\}$. For all A, U_0, U_1, constant c, large n, we have $\Pr[b = b'] < n^{-c}$.

5. Architecture

A general architecture of privacy-ABC consists of three main entities—a user (credential owner also known as prover), an issuer (trusted third party or credential signer), and a verifier (services provider or secure resource owner). The issuer issues credentials for users, which can later be used for authentication purposes. Such an architecture may optionally include an entity that takes care of revocation of credentials (revocation authority) and another entity (inspector) that can revoke the anonymity of users. Inspection and revocation are beyond the scope of this paper. Different entities and their relations are illustrated in Figure 3.

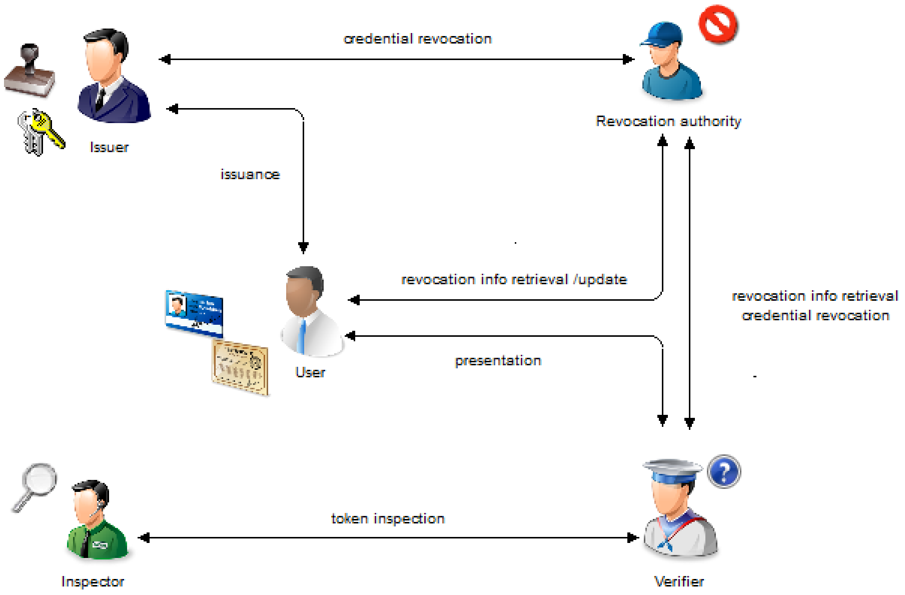


Figure 3. Entities of ABC.

The main phases of a privacy-ABC system are described below:

- Set-up: Performed to output system's parameters.
- Issuance: An interactive protocol between users and issuer. By issuing a credential to a user, the issuer guarantees the correctness of attributes contained in the credential.
- Presentation: An interactive protocol in which a user reveals or proves to a verifier possession of some attributes or claims about attributes. This phase is also known as verification.
- Inspection: Provides conditional anonymity. It enables a trusted party, the so-called inspector, to revoke, in some conditions, anonymity of cheating provers.
- Revocation: Ends the validity of credentials whenever necessary.

At the core of privacy-ABC system, untraceability and unlinkability are the most important privacy-related features. Additional features are also supported [11,16,21,37]; without exhaustivity, we cite:

- Authenticity: feature that guarantees that an ABC cannot be modified.
- Non-transferability: feature that prevents the user from transferring her ABC to another user.
- Minimal information: feature that guarantees that during the verification protocols no other information is revealed to the verifier beyond the disclosed attributes, and the corresponding issuer, etc.
- Multi-show unlinkability: feature that guarantees that different presentations of a given ABC cannot be linked.
- Issuance unlinkability: feature which guarantees that the presentation of an ABC cannot be linked to its issuance.
- Selective disclosure: Allows a user to prove only a subset of attributes to a verifier.
- Carry-over attributes: Enables users to carry over some attributes from an existing ABC into a new one without disclosing them to the issuer.
- Predicate proof: Allows logical operators to be applied on attributes without disclosing them.
- Proof of holdership: A cryptographic evidence for proving ownership of ABC.
- Unforgeability: feature that guarantees that no malicious third party can forge a valid ABC.
- Etc.

6. Contribution

In this section, our scheme is described. We focus on three main phases, namely set-up, issuance, and verification (showing, presentation). We also present a selective disclosure protocol and randomization of credential.

6.1. Set-Up

The set-up is the first algorithm to be run to extract system's parameters from a security parameter k. This algorithm is described as follows:

- $\{p, q, E(\mathbb{F}_p), P, P_{pub}, \mathcal{H}\ \} \longleftarrow 1^k$

where:

- k: a security parameter.
- p: a prime number that defines the field $\mathbb{F}_p$.
- E: an elliptic curve defined over $\mathbb{F}_p$.
- $P \in E(\mathbb{F}_p)$: a base point of prime order q.
- $x \in_R \mathbb{F}_q^*$: issuer's secret key.
- $P_{pub} = x.P$: issuer's public key.
- $\mathcal{H}$: a hash function defined as follow:

$$\begin{aligned} \mathcal{H} :& E(\mathbb{F}_p)^2 \longrightarrow \mathbb{F}_p^* \\ &(P, Q) \longmapsto \mathcal{H}(P, Q) \end{aligned}$$

6.2. Issuance

Once the set-up algorithm is successfully executed, the issuer is ready to issue credentials. When a user wants to be issued a credential, he runs an interactive algorithm with the issuer at the end of which a credential should be issued for him. Issuance unlinkability property should be ensured. The way it is done is that not only the issuer can not store the issued credential, but also, thanks to the blinding mechanism, he may not see the credential's content while issuing.

6.3. Signature on a Single Message

The issuing protocol on a single message is an interactive protocol of three steps as described below:

- Blinding: The issuer generates a random integer $\bar{k} \in \mathbb{F}_q^*$, computes the resulting point $\bar{R} = \bar{k}.P$ and sends it to the user. The latter generates random factors α and β, blinds her document, and sends it to the issuer.
- Singing: The issuer signs the blinded document with his secret key x by computing $\bar{s} \equiv \bar{h}x + \bar{k}(mod\ q)$ and sends the blinded and signed document to the user.
- Unblinding: The user unblinds the blinded and signed document with her blind factors α and β without invalidating it.

For convenient notation, proofs of knowledge would be noted PK. The corresponding proofs are those mentioned in the so-called protocol. Details of the issuing protocol on a single message are described in "Table 2".

Table 2. Issuance protocol on a single message.

User **Secret m_0**	**Public** $k, p, q, P, P_{pub}, E, \mathcal{H}$	**Issuer** **Secret x**
Blinding		
$\alpha, \beta \in_R \mathbb{F}_q^*$	$\xleftarrow{\bar{R}}$	$\bar{k} \in_R \mathbb{F}_q^*, \bar{R} = \bar{k}.P$
$R = \alpha.\bar{R} + \beta.P$		
$P_0 = m_0.P$		
$h = \mathcal{H}(P_0, R)$		
$\bar{h} \equiv h\alpha^{-1}(mod\ q)$		
$PK\{(\mu) : P_0 = \mu.P\}$	$\xrightarrow{\bar{h}, P_0, PK}$	Signing
Verification	$\xleftarrow{\bar{s}}$	$\bar{s} \equiv \bar{h}x + \bar{k}(mod\ q)$
$\bar{s}.P \overset{?}{=} \bar{h}.P_{pub} + \bar{R}$		
Unblinding		
$s \equiv \alpha\bar{s} + \beta(mod\ q)$		
Outputs (R,s) the blind version of $(\bar{R}, \bar{s})$		

Theorem 2. *The proposed scheme is fully blind.*

Proof. From $\bar{s} \equiv \bar{h}x + \bar{k}(\text{mod } q)$, we have $\alpha \equiv hx(\bar{s} - \bar{k})^{-1}(\text{mod } q)$. Thus, $\alpha \in \mathbb{F}_q^*$ is unique. It follows that $\beta \equiv s - \alpha\bar{s}(\text{mod } q)$ is also unique. Thus, there always exist α and β, regardless of $(\bar{R}, \bar{h}, \bar{s})$ and (m, R, s), such that $(\bar{R}, \bar{h}, \bar{s})$ and (m, R, s) have the same relation. Therefore, an adversary A outputs a correct value b' with probability exactly $\frac{1}{2}$. As a result, the issuing protocol is fully blind. □

Theorem 3. *The proposed scheme is $(\epsilon', t', q_i, q_h)$-secure in the sense of unforgeability under chosen message attack (UE-CMA) in the random oracle model, assuming that the (ϵ, t)-ECDL assumption holds in $\langle P \rangle$, where $t' = t + \mathcal{O}(q_i)T$, $\epsilon' = (1 - \frac{q_h q_i}{q})(1 - \frac{1}{q})(\frac{1}{q_h})\epsilon$ and q_i, q_h are the number of issue and hashing queries, respectively, the adversary is allowed to perform, while T denotes the time for a scalar multiplication operation.*

Proof. Assuming that there exists a forger $\mathcal{A}$ that can forge a credential while playing the game of chosen message attack (CMA) [38], we construct an algorithm $\mathcal{M}^{\mathcal{A}}$ that uses $\mathcal{A}$ to solve the discrete logarithm problem. Without losing generality, we assume that $q_i q_h < q$. The following section is adapted from the proof presented by Joseph K Lui et al. [39].

- Set-up: $\mathcal{M}^{\mathcal{A}}$ receives the problem $P_1 : (k, p, q, E(\mathbb{F}_p), P, P_{sec})$ and should find $z \in \mathbb{F}_p$ such that $P_{sec} = z.P$. It chooses a hash function $\mathcal{H}$ which behaves like a random oracle, sets $P_{pub} = P_{sec}$ and sends the public parameters $(k, p, q, E(\mathbb{F}_p), P, P_{pub}, \mathcal{H})$ to $\mathcal{A}$ expecting it forges a credential. $\mathcal{M}^{\mathcal{A}}$ and $\mathcal{A}$ start playing the game of chosen message attack [38].
- Hashing oracle: $\mathcal{M}^{\mathcal{A}}$ starts by initializing an empty database. When $\mathcal{A}$ sends m_i for hashing, $\mathcal{M}^{\mathcal{A}}$ checks whether or not that message has already been sent. If so, it picks h_i from the database and returns it as a response to that query, otherwise it picks $h_i \in_R \mathbb{F}_p^*$, stores the couple (m_i, h_i) in the database and returns h_i as a response to that query.
- Issuing oracle: When $\mathcal{A}$ queries the issuing oracle for the message m_i, $\mathcal{M}^{\mathcal{A}}$ first checks whether m_i has already been queried for issuance. If so, it aborts and the game is stopped (Event 1) [38], otherwise, it computes $h_i = RO(m_i)$ (where RO denotes the hashing oracle), picks $s_i \in_R \mathbb{F}_p^*$, computes $R_i = s_i.P - h_i.P_{pub}$ and sends (R_i, s_i) to $\mathcal{A}$ as response to its issuing query. As we can see, the algorithm is valid since $s_i.P = h_i.P_{pub} + R_i$.
- Forging step: Finally, $\mathcal{A}$ outputs a forged signature $\sigma^* = (s^*, R^*)$ on message m^* with h^*. It computes $\frac{s^* - r^*}{h^*}$ and extracts z as solution of the DLP.
- Probability analysis: The simulation fails if $\mathcal{A}$ queries the same message for issuance (Event 1). This happens with probability at most $\frac{q_h}{q}$. Hence, the simulation is successful with probability at least $(1 - \frac{q_h}{q})^{q_i} \geq (1 - \frac{q_h q_i}{q})$ (provable by recurrence reasoning). The tuple (m^*, R^*, s^*) is a valid

credential with probability at least $1 - \frac{1}{q}$ [39] and $\mathcal{M}^{\mathcal{A}}$ guesses it correctly with probability at least $\frac{1}{q_h}$ [39]. Finally, the overall successful probability is $\epsilon' = (1 - \frac{q_h q_i}{q})(1 - \frac{1}{q})(\frac{1}{q_h})\epsilon$ [39]. The time complexity of the algorithm $\mathcal{M}^{\mathcal{A}}$ is $t' = t + \mathcal{O}(q_i)T$ since the issuing oracle computes at most $2q_iT$ scalar multiplications.

□

6.4. Verification

Once the credential is issued, the user can be authenticated by a service provider. As for the issuing protocol, he performs an interactive protocol with the service provider at the end of which an access to the so-called service might be granted or denied. The user must prove knowledge of his secret key. This guarantees unforgeability property. The verification protocol is described in "Table 3".

Table 3. Verification protocol.

Prover	**Public**	**Verifier**
Secret m_0	$\boldsymbol{k, p, q, P, P_{pub}, E, \mathcal{H}}$	
Keeps secret $(\bar{R}, \bar{s})$ $PK\{(\mu) : P_0 = \mu.P\}$	$\xrightarrow{(R,s),h,PK}$	$s.P \overset{?}{=} h.P_{pub} + R$

6.5. Randomized Version

Many digital services involved in our life emphasize users' privacy. Blind signature is a well-known technique to address privacy concerns. When a user presents the same signature (R, s) multiple times, he could be traceable; R, s could be stored by the verifier even if this cannot be linked to issuance. The randomized version allows a user to derive a random signature from a valid one without invalidating it. The randomization feature is fundamental because it guarantees multi-show unlinkability. The prover generates a random factor r from which a random signature $(\hat{R}, \hat{s})$ is derived. The process of randomization is described in "Table 4".

Table 4. Randomization protocol.

Prover	**Public**	**Verifier**
Secret m_0	$\boldsymbol{k, p, q, P, P_{pub}, E, \mathcal{H}}$	
$r \in_R \mathbb{F}_q^*$ $\hat{s} \equiv s + r(mod\ q)$ $\hat{R} = R + r.P$	$\xrightarrow{(\hat{R},\hat{s}),h,PK}$	$\hat{s}.P \overset{?}{=} h.P_{pub} + \hat{R}$

6.6. Signature on a Block of Messages

A credential rarely contains one attribute. In this section, we consider a credential of l attributes $(m_1, ..., m_l)$. The issuing protocol on a block of messages is described in "Table 5". The verification protocol remains the same as in Section 6.4.

Table 5. Issuance protocol on a block of messages.

User	**Public**	**Issuer**
Secret m_0	$k, p, q, P, P_{pub}, E, \mathcal{H}$	**Secret x**
<u>Blinding</u>		
$\alpha, \beta \in_R \mathbb{F}_q^*$	$\xleftarrow{\bar{R}}$	$\bar{k} \in_R \mathbb{F}_q^*, \bar{R} = \bar{k}.P$
$R \equiv \alpha.\bar{R} + \beta.P$		
$h = \prod_{i=0}^{l} \mathcal{H}(P_i, R)$		
Where $P_i = m_i.P$		
$\bar{h} \equiv h\alpha^{-1}(mod\ q)$		
$PK\{(\mu) : P_0 = \mu.P\}$	$\xrightarrow{\bar{h},P_0,PK}$	<u>Signing</u>
<u>Verification</u>	$\xleftarrow{\bar{s}}$	$\bar{s} \equiv \bar{h}x + \bar{k}(mod\ q)$
$\bar{s}.P \overset{?}{=} \bar{h}.P_{pub} + \bar{R}$		
<u>Unblinding</u>		
$s \equiv \alpha\bar{s} + \beta(mod\ q)$		
Outputs (R,s) the		
blind version of $(\bar{R}, \bar{s})$		

6.7. Selective Disclosure

The fundamental principle of privacy master is data minimization. Selective disclosure is a way to achieve this. It is the ability of an individual to granularly decide what information to share. In our context, it is a very interesting feature that lets a user decide what attributes to disclose while being authenticated. How a user and verifier agreed on the attributes to disclose is beyond the scope of this paper. After being convinced about hidden and revealed attributes, the verifier grants access to the prover. More concretely, in the selective disclosure protocol, the prover decides to disclose the subset $m_{n+1}, m_{n+2}, ..., m_l$ while $m_0, m_1, ..., m_n$ remains secret $(n < l)$. The selective disclosure protocol is described in "Table 6".

Table 6. Selective disclosure protocol.

Prover	**Public**	**Verifier**
Secret $m_0, m_1, ..., m_n$	$k, p, q, P, P_{pub}, E, \mathcal{H}$	
$PK\{(s, R, \mu_0, ..., \mu_n) :$ $\prod_{i=0}^{n} \mathcal{H}(P_i, R).P_{pub} =$ $(\prod_{i=n+1}^{l} \mathcal{H}(P_i, R))^{-1}(s.P - R) \wedge$ $P_i = \mu_i.P,\ 0 \leq i \leq n\ \}$	$\xrightarrow{(R,s),h,PK}$	$s.P \overset{?}{=} h.P_{pub} + R$

7. Complexity Analysis

Designing an algorithm is good, but designing an efficient algorithm is better. Let P be a problem, with M_1 and M_2 two methods to solve P. Answer to the question: which of the methods M_1 and M_2 is more efficient is not trivial unless one knows their complexity. A complexity is a mathematical approximation to estimate the number of operations and/or memory required for an algorithm to solve a problem. A good algorithm must therefore use as little memory as possible and make the processor work as little as possible. In the remainder of this section, we adopt the following notations:

- M_s: Scalar multiplication over EC
- A_p: Point adding over EC

This section aims to compare complexities of schemes I2PA, U-prove, and Idemix. While tackling IoT devices, the case of Idemix could be left behind because a basic RSA operation (inversion or elevation to power on large numbers) is more expensive than a well-designed operation on an EC [9,40]. It should be noted that basic arithmetic operations such as addition and multiplication are negligible, in terms of resource consumption, compared to operations in elliptic curves or RSA base operations (power elevation and inversion). Given that Idemix does not provide, as far as we know, an EC-based implementation, we will focus on the memory usage. In the rest of this section, we consider a credential of *n attributes* to sign and verify. Results presented below are based on [5–7,21,41] and simplified schemes presented by Alpár Gergely [42,43].

7.1. Operations over the Curve Comparison

In this first part of comparative study, we focus on U-prove and I2PA schemes regarding operations on the elliptic curve. We do not consider the hash function because its fundamental property is that it should be very easy and quick to compute. We complete the list with the scheme ecHM12 [23] focusing on the verification (proving) protocol, since this scheme does not perform any operation in the curve during the issuance protocol. The comparison results are recorded in "Table 7".

Table 7. Operations over the curve analysis.

Protocol	U-prove	I2PA	ecHM12
Issuance	$(n+6).M_s + (n+4).A_p$	$(n+6).M_s + 2.A_p$	*None*
Verification	$2.M_s + 2.A_p$	$2.M_s + 1.A_p$	$19.M_s + 8.A_p$

"Table 7" shows that U-prove performs more operations in the curve than I2PA what should be the number of attributes to issue and verify. In addition, the issuance is far more expensive in U-prove than in I2PA especially when the number of attributes grows. ecHM12 is by far the greediest, seeing the number of operations required in the proving (verification) protocol. We can safely conclude, without going wrong, that I2PA is more efficient than U-prove in terms of computing time when system parameters are the same.

7.2. Memory Usage Comparison

In this second part of comparative study, we are interested in memory usage. We compare the number of bits in the issuing phase. This choice is justified by the fact that in the verification phase, the verifier has nothing to store. We adopt the following notation:

- T: Total of variables needed in this phase
- P: Total of variables to be stored permanently in this phase

In addition, we assume that all variables of a protocol have the same size and this size is the same as the security level. If in the Idemix scheme we work with 1024 bits, then in I2PA and U-prove we can work with 160 bits expecting the same security level [9]. Results obtained are recorded in "Table 8".

Table 8. Memory usage analysis.

Protocol	Idemix	U-prove	I2PA
User	T: n + 6 P: n + 4	T: n + 9 P: n + 4	T: n + 9 P: n + 4
Issuer	T: n + 9 P: n + 4	T: 2n + 8 P: 2n + 6	T: 10 P: 7

"Table 8" shows that at user side, I2PA and U-prove require around $1024(n+9)$ bits while Idemix requires around $1024(n+6)$ bits. However, at issuer side, I2PA presents interesting results (around 1600 bits which does not dependent on the number of attributes) compared to U-prove and Idemix that require respectively $160(2n+8)$ and $1024(n+9)$. We can also see that if implementation is based on RSA cryptosystem, U-prove presents less interesting results than Idemix. Finally, we can conclude that I2PA is very efficient compared to Idemix and U-prove, the two leaders, in terms of anonymous credentials.

7.3. Feature Comparison

In this last part of comparative study, we focus on the number of key features that need to be carefully considered when tackling privacy concerns in an IoT context. We consider three scenarios for a feature. It can be totally, partially or not at all supported (not provided). The following legend is adopted:

- ✓: Fully supported
- ⊘: Partially supported
- ×: Not supported

The "Table 9" is a summary of features comparison.

Table 9. Features analysis.

Protocol	Idemix	U-prove	I2PA
fully blind signature	⊘	✓	✓
Selective disclosure	✓	✓	✓
Randomization	✓	×	✓
Untraceability	✓	⊘	✓
Unlinkability	✓	✓	✓
Unforgeability	✓	✓	✓
small keys size	×	✓	✓
Bandwidth saving	×	✓	✓
small devices efficiency	×	✓	✓

Idemix issuance is not totally blind because two of the three parts of the resulting credential are known by the issuer. These two elements are A and e of the signature (A, e, v). In addition, this model is not optimal, with low-resource devices and bandwidth optimization. U-prove, meanwhile, does not support the randomization feature. Even if the issuing phase is blind, the issues related to non-traceability are not fully taken into account. Indeed, the triple (h', c', r') where (c', r') constitutes the signature, if presented several times, may be traceable. The problem can be seen in two angles. First, h', constituting the user's public key, is required in the verification phase. Since the latter only depends on the signed attributes and the secret key of the user, then the probability of having two credentials with the same public key is almost zero. Secondly, lack of randomization may be a problem. To guarantee non-traceability in U-prove, one needs several credentials [21,22]. In our schema, the triple (h, R, c) can become (h, R', c'), where (R', c') is a random version of (R, c) without invalidating the signature.

8. Performance Evaluation

This section describes measurements obtained for issuance and verification of 100 credentials with 10 fixed attributes. We roughly describe software, hardware, parameters, and curve used in this simulation.

8.1. Hardware Set-up

The hardware set-up used in this experimental evaluation consists of a Raspberry Pi and a smartphone. The Raspberry Pi (Figure 4) is used to deploy both issuers and verifiers. Below are some of its characteristics:

- Model Pi 3 B+
- 1 Go of SDRAM LPDDR2
- A 64-bit quad core processor clocked at 1.4 GHz
- Raspbian operating system.
- Dual Band 2.4 GHz and 5GHz IEEE 802.11. b/g/n/AC Wireless LAN
- Enhanced Ethernet performance over USB 2.0 (maximum throughput of 300 Mbps).

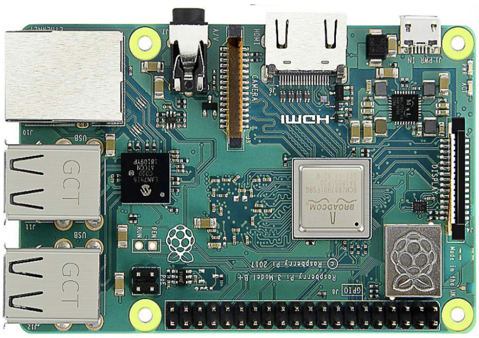
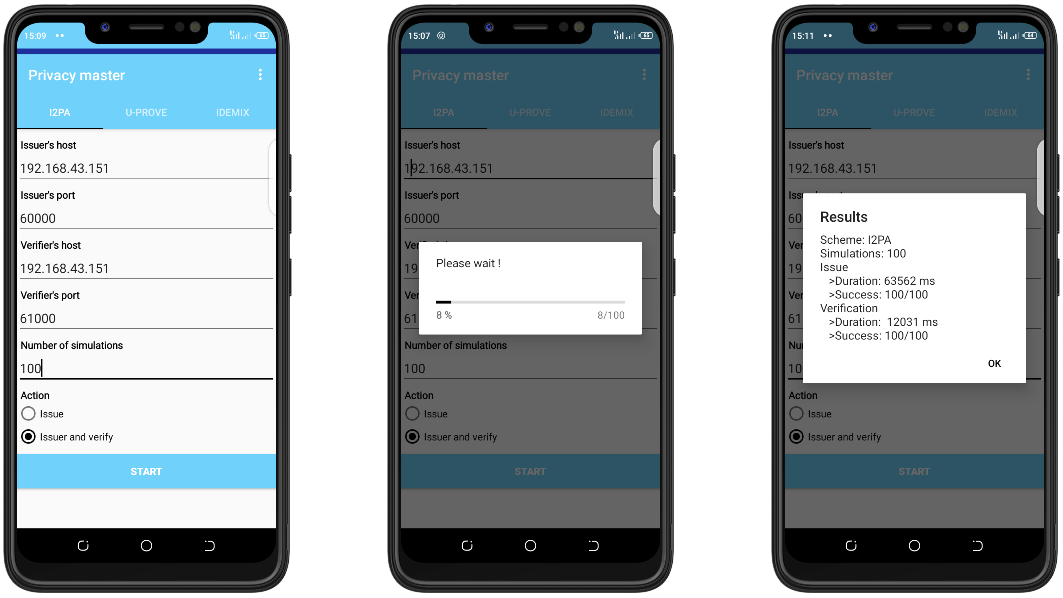


Figure 4. Raspberry Pi board.

The smartphone (Figure 5) acts as a user. It interacts with issuers to get credentials and with verifiers for verification of credentials purposes. Below are some of its main characteristics:

- model TECNO SPARK KB7j
- RAM 2 GB
- ROM 16 GB
- CPU 2.0 GH*4
- Battery 3500 mAh
- Memory 16 GB



Issuer's host
192.168.43.151
Issuer's port
60000
Verifier's host
192.168.43.151
Verifier's port
61000
Number of simulations
100
Action
○ Issue
● Issuer and verify

Please wait !
8 % 8/100

Results
Scheme: I2PA
Simulations: 100
Issue
>Duration: 63562 ms
>Success: 100/100
Verification
>Duration: 12031 ms
>Success: 100/100
OK

(a) Initial state **(b)** Processing state **(c)** Processing result

Figure 5. Screenshots of the Android application states.

8.2. Software Set-up

The software environment is made up of three major components: issuer, verifier, and user (Figure 6). Issuer and verifier are implemented with Java servers sockets while user is a mobile application developed with Android and Java client socket. We run both issuer and verifier servers on the same device (the Raspberry Pi).

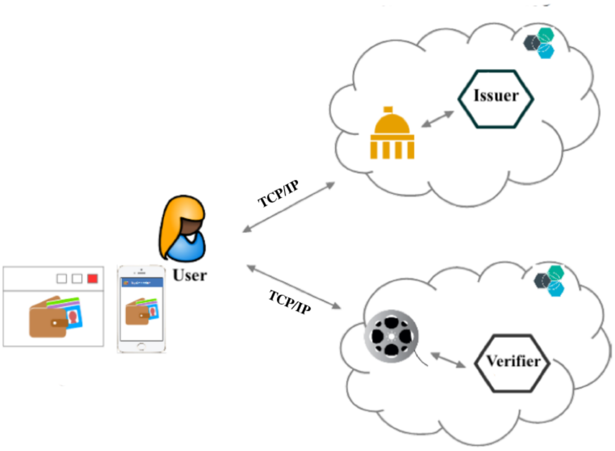


Figure 6. Software architecture.

8.3. Curve and Parameters

Edwards' curves are known to offer better performance among all elliptic curve families [13]. In this experience, we use the curve Curve25519 to implementation schemes I2PA and U-prove (refer to [44] for more details about this curve) with 160 bits key size while Idemix is implemented with 1024 bits key size. We also use extended homogeneous coordinates (EHCs) to speed up operations as recommended by Josefsson et al. [45]. In the EHCs representation, the affine couple (x, y) is represented as $(X : Y : Z : T)$ where $x = \frac{X}{Z}, y = \frac{Y}{Z}$ and $xy = \frac{T}{Z}$. The neutral point $(0, 1)$ is equivalent to $(0 : Z : Z : 0)$ for any nonzero Z. Coordinates $(X : Y : Z : T)$ and $(\lambda X : \lambda Y : \lambda Z : \lambda T)$ are equivalent for any nonzero λ. EHMs avoid inversion operations and, as a result, improve time computation.

8.4. Issuance

Results recorded from issuance are illustrated in the following graphs. In Figures 7 and 8, we illustrate respectively the variation of the duration by simulation and minimum, maximum, and average values.

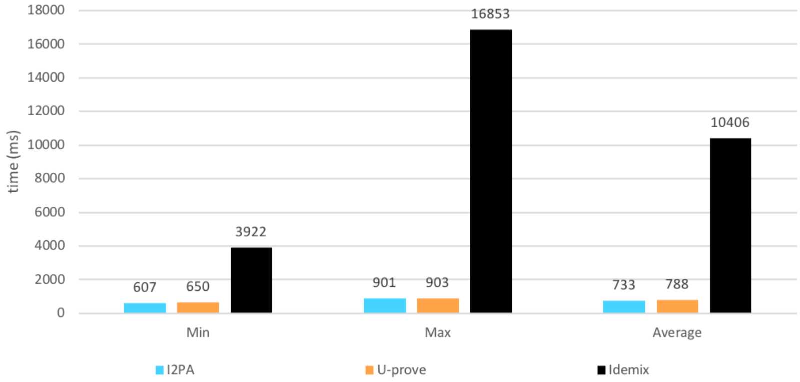


Figure 7. Issuances' metrics.

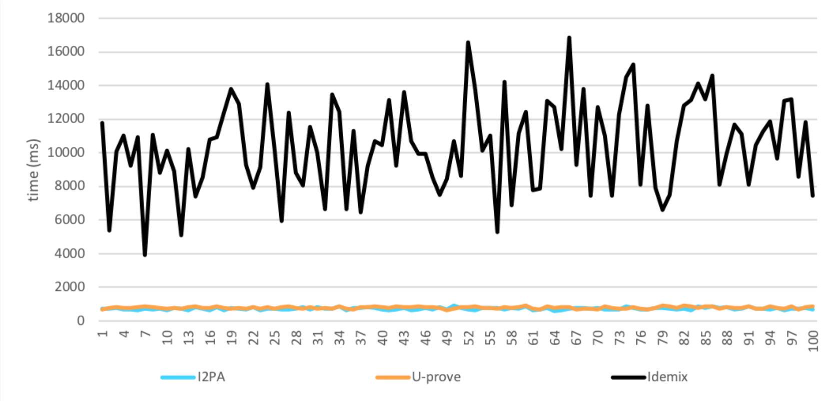


Figure 8. Variation of issuance duration by simulation.

Figures 7 and 8 show that I2PA and U-prove present similar performance on issuance. Idemix, meanwhile, has very low performance compared to I2PA and U-prove. The time it requires for issuance is at least six times greater than that required by I2PA and U-prove. Considering distribution of time, 50% of simulations have duration higher or equal to the average for I2PA , 49% for U-prove and 51% for Idemix. Finally, we can see, without any risk of being wrong, that I2PA is efficient compared to U-prove and very efficient compared to Idemix. Its curve remains almost below that of U-prove. For 100 simulations, it offers the best performance. The minimum, maximum, and average durations (Figure 7) always remain below the others.

8.5. Verification

As for the issuance, the verification is carried out on 100 credentials of ten fixed attributes. Figure 9 represents the minimum, maximum and average values while Figure 10 represents the variation of duration by simulation.

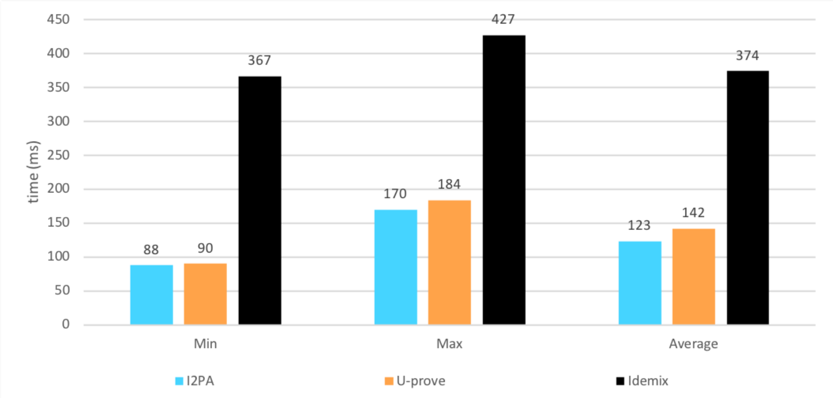


Figure 9. Verifications' metrics.

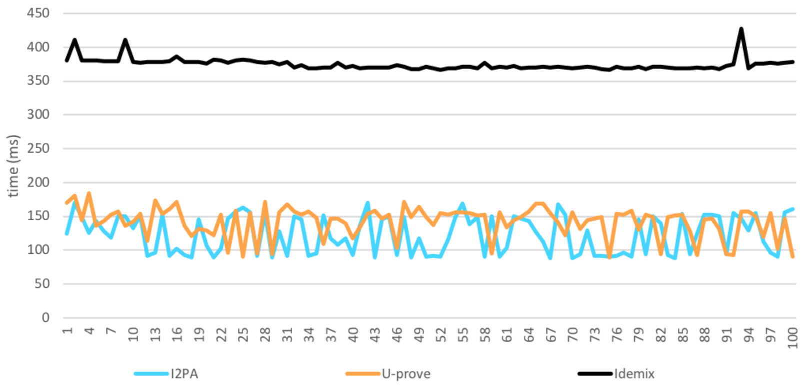


Figure 10. Variation of verification duration by simulation.

As for the issuance, Idemix presents less interesting results than I2PA and U-prove, which present very similar performance. Regarding distribution of time, 52% of simulations have duration higher or equal to the average for I2PA, 66% for U-prove and 41% for Idemix. Finally, we can safely conclude that I2PA presents very interesting results compared to U-prove. From the minimum duration to the maximum and the average, it presents best performance.

9. Conclusions

Until recently, authentication without identification was impossible. Anonymous credentials are a suitable way to achieve this. In this paper, we propose an efficient ABC scheme for IoT. We use elliptic curves without pairing, zero knowledge proof, blind signing, selective disclosure, and randomization. Our scheme guarantees anonymity, which is a fundamental aspect for privacy preservation. As stated in Section 3, our scheme is $(\epsilon', t', q_i, q_h)$-secure in the sense of unforgeability under chosen message attack in the random oracle model. Analysis presented in Section 7 and performance recorded in Section 8 show that our scheme is very suitable for an IoT environment with severely constrained resources. Future work would include performance study, predicate proof over attributes, inspection, and revocation protocols. We would also investigate on how to choose attributes to disclose efficiently in the selective disclosure protocol.

Author Contributions: Conceptualization, A.A.C. and O.N.; Formal analysis, I.S., A.A.C. and O.N.; Investigation, I.S., A.A.C. and O.N.; Methodology, I.S., A.A.C. and O.N.; Software, I.S., A.A.C. and O.N.; Project administration, A.A.C. and O.N.; Supervision, A.A.C. and O.N.; Validation, A.A.C. and O.N.; Writing—original draft, I.S., A.A.C. and O.N.; Writing—review and editing, I.S., A.A.C. and O.N.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Miranda, J.; Mäkitalo, N.; Garcia-Alonso, J.; Berrocal, J.; Mikkonen, T.; Canal, C.; Murillo, J.M. From the Internet of Things to the Internet of People. *IEEE Internet Comput.* **2015**, *19*, 40–47. [CrossRef]
2. Bradley, J.; Barbier, J.; Handler, D. L'internet of Everything, un Potentiel de 14,4 Trillions de Dollars. Available online: https://www.cisco.com/web/FR/tomorrow-starts-here/pdf/ioe_economy_report_fr.pdf (accessed on 12 March 2018).
3. Mbaye, A.; Ciss, A.A.; Niang, O. A Lightweight Identification Protocol for Embedded Devices. *arXiv* **2014**, arXiv:1408.5945.
4. Alpar, G.; Hoepman, J.H. A secure channel for attribute-based credentials: [Short paper]. In Proceedings of the 2013 ACM Workshop on Digital Identity Management, Berlin, Germany, 8 November 2013; pp. 13–18. [CrossRef]

5. Camenisch, J.; Van Herreweghen, E. Design and implementation of the idemix anonymous credential system. In Proceedings of the 9th ACM Conference on Computer and Communications Security, Washington, DC, USA, 18–22 November 2002; pp. 21–30.
6. Paquin, C.; Zaverucha, G. *U-Prove Cryptographic Specification v1.1*; Technical Report; Microsoft Corporation: Redmond, WA, USA, 2011.
7. Vullers, P.; Alpár, G. Efficient selective disclosure on smart cards using idemix. In *IFIP Working Conference on Policies and Research in Identity Management*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 53–67.
8. Alpár, G.; Batina, L.; Lueks, W. Designated attribute-based proofs for RFID applications. In *International Workshop on Radio Frequency Identification: Security and Privacy Issues*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 59–75.
9. Sinha, R.; Srivastava, H.K.; Gupta, S. Performance based comparison study of RSA and elliptic curve cryptography. *Int. J. Sci. Eng. Res.* **2013**, *4*, 720–725.
10. Bernal Bernabe, J.; Hernandez-Ramos, J.L.; Skarmeta Gomez, A.F. Holistic privacy-preserving identity management system for the internet of things. *Mob. Inf. Syst.* **2017**, *2017*, 6384186. [CrossRef]
11. Persiano, G.; Visconti, I. An Efficient and Usable Multi-show Non-transferable Anonymous Credential System. In *Financial Cryptography*; Springer: Berlin/Heidelberg, Germany, 2004; pp. 196–211. [CrossRef]
12. About IRMA. Available online: https://privacybydesign.foundation/irma-en/ (accessed on 3 March 2018).
13. Liu, Z.; Seo, H.; Xu, Q. Performance evaluation of twisted Edwards-form elliptic curve cryptography for wireless sensor nodes. *Secur. Commun. Netw.* **2015**, *8*, 3301–3310. [CrossRef]
14. De Fuentes, J.; González-Manzano, L.; Serna-Olvera, J.; Veseli, F. Assessment of attribute-based credentials for privacy-preserving road traffic services in smart cities. *Person. Ubiquitous Comput.* **2017**, *21*, 869–891. [CrossRef]
15. Lueks, W.; Alpár, G.; Hoepman, J.H.; Vullers, P. Fast revocation of attribute-based credentials for both users and verifiers. *Comput. Secur.* **2017**, *67*, 308–323. [CrossRef]
16. de Fuentes, J.M.; Gonzalez-Manzano, L.; Solanas, A.; Veseli, F. Attribute-Based Credentials for Privacy-Aware Smart Health Services in IoT-Based Smart Cities. *Computer* **2018**, *51*, 44–53. [CrossRef]
17. Hajny, J.; Dzurenda, P.; Malina, L. Attribute-based credentials with cryptographic collusion prevention. *Secur. Commun. Netw.* **2015**, *8*, 3836–3846. [CrossRef]
18. Camenisch, J.; Lysyanskaya, A. An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In *International Conference on the Theory and Applications of Cryptographic Techniques*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 93–118.
19. Brands, S.A. *Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy*; MIT Press: Cambridge, MA, USA, 2000.
20. Camenisch, J.; Lysyanskaya, A. Signature schemes and anonymous credentials from bilinear maps. In *Annual International Cryptology Conference*; Springer: Berlin/Heidelberg, Germany, 2004; pp. 56–72.
21. Alpár, G. Attribute-Based Identity Management : Bridging the Cryptographic Design of ABCs with the Real World. Available online: https://repository.ubn.ru.nl/bitstream/handle/2066/135177/135177.pdf (accessed on 16 september 2018).
22. Hanzlik, L.; Kluczniak, K. A short paper on how to improve U-Prove using self-blindable certificates. In *International Conference on Financial Cryptography and Data Security*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 273–282.
23. Dzurenda, P.; Hajny, J.; Malina, L.; Ricci, S. Anonymous Credentials with Practical Revocation using Elliptic Curves. In Proceedings of the 14th International Joint Conference on e-Business and Telecommunications—Volume 6: SECRYPT, Madrid, Spain, 24–26 July 2017; pp. 534–539. [CrossRef]
24. Hajny, J.; Dzurenda, P.; Malina, L. Privacy-PAC: Privacy-Enhanced Physical Access Control. In Proceedings of the 13th Workshop on Privacy in the Electronic Society, Scottsdale, AZ, USA, 3 November 2014; pp. 93–96. [CrossRef]
25. Koblitz, N. Elliptic curve cryptosystems. *Math. Comput.* **1987**, *48*, 203–209. [CrossRef]
26. Miller, V.S. Use of elliptic curves in cryptography. In *Conference on the Theory and Application of Cryptographic Techniques*; Springer: Berlin/Heidelberg, Germany, 1985; pp. 417–426.
27. Hu, X.; Zheng, X.; Zhang, S.; Li, W.; Cai, S.; Xiong, X. A High-Performance Elliptic Curve Cryptographic Processor of SM2 over GF(p). *Electronics* **2019**, *8*, 431. [CrossRef]

28. Hu, X.; Zheng, X.; Zhang, S.; Cai, S.; Xiong, X. A Low Hardware Consumption Elliptic Curve Cryptographic Architecture over GF(p) in Embedded Application. *Electronics* **2018**, *7*, 104. [CrossRef]
29. Lara-Nino, C.A.; Diaz-Perez, A.; Morales-Sandoval, M. Energy/Area-Efficient Scalar Multiplication with Binary Edwards Curves for the IoT. *Sensors* **2019**, *19*, 720. [CrossRef] [PubMed]
30. Liu, Z.; Seo, H. IoT-NUMS: Evaluating NUMS elliptic curve cryptography for IoT platforms. *IEEE Trans. Inf. Forensics Secur.* **2018**, *14*, 720–729. [CrossRef]
31. Chatzigiannakis, I.; Vitaletti, A.; Pyrgelis, A. A privacy-preserving smart parking system using an IoT elliptic curve based security platform. *Comput. Commun.* **2016**, *89*, 165–177. [CrossRef]
32. Aziz Ciss, A. Trends in Elliptic Curves Cryptography. *IMHOTEP Afr. J. Pure Appl. Math.* **2015**, *2*, 1–12.
33. Bernstein, D.J.; Lange, T. Faster addition and doubling on elliptic curves. In *International Conference on the Theory and Application of Cryptology and Information Security*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 29–50.
34. Bernstein, D.J.; Birkner, P.; Joye, M.; Lange, T.; Peters, C. Twisted edwards curves. In *International Conference on Cryptology in Africa*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 389–405.
35. Hisil, H.; Wong, K.K.H.; Carter, G.; Dawson, E. Twisted Edwards curves revisited. In *International Conference on the Theory and Application of Cryptology and Information Security*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 326–343.
36. Goldwasser, S.; Micali, S.; Rackoff, C. The Knowledge Complexity of Interactive Proof-Systems (Extended Abstract). In Proceedings of the 17th Annual ACM Symposium on Theory of Computing, Providence, RI, USA, 6–8 May 1985; pp. 291–304. [CrossRef]
37. Qu, H.; Shang, P.; Lin, X.J.; Sun, L. Cryptanalysis of A Privacy-Preserving Smart Metering Scheme Using Linkable Anonymous Credential. *IACR Cryptol. ePrint Arch.* **2015**, *2015*, 1066.
38. Hülsing, A. Digital Signature Schemes and the Random Oracle Model. Available online: https://www.win.tue.nl/applied_crypto/2016/20161115_ROM_Signatures.pdf (accessed on 24 February 2019).
39. Liu, J.K.; Baek, J.; Zhou, J.; Yang, Y.; Wong, J.W. Efficient online/offline identity-based signature for wireless sensor network. *Int. J. Inf. Secur.* **2010**, *9*, 287–296. [CrossRef]
40. Jansma, N.; Arrendondo, B. Performance Comparison of Elliptic Curve and RSA Digital Signatures. Available online: http://www.nicj.net/files/performance_comparison_of_elliptic_curve_and_rsa_digital_signatures.pdf (accessed on 24 February 2019).
41. Camenisch, J. *Direct Anonymous Attestation Explained*; Technical Report; IBM Research: Yorktown Heights/Albany, NY, USA, 2007.
42. Alpar, G. Cryptography Fact Sheet about Idemix's Basic Proof Techniques. Available online: https://privacybydesign.foundation/pdf/idemix_overview.pdf (accessed on 24 February 2018).
43. ALPAR, G. U-PROVE CRYPTOGRAPHY. Available online: http://www.cs.ru.nl/~gergely/objects/u-prove.pdf (accessed on 24 February 2018).
44. El Housni, Y. *Edwards Curves*; Working Paper or Preprint; HAL Id: hal-01942759; HAL: Paris, France, 2018.
45. Josefsson, S.; Liusvaara, I. *Edwards-Curve Digital Signature Algorithm (EdDSA)*; Technical Report. RFC: 2017; Volumle: 8032, pp. 1–60. Available online: http://www.rfc-editor.org/info/rfc8032 (accessed on 21 June 2019).