Efficient Attributes Proof on CNF Relation for eID System

Nan Guo[1]*, Wenwei Jiang[1], Jiayi Ouyang[1], Tianhan Gao[2] and Bin Zhang[1]
[1] Computer Science and Engineering College, Northeastern University, Shenyang 110819 China
guonan@mail.neu.edu.cn, jiangwenwei01@163.com
ouyangjiayi04@163.com, zhangbin@mail.neu.edu.cn
[2] Software College, Northeastern University, Shenyang 110819 China
gaoth@mail.neu.edu.cn

Abstract

Electronic identity system is a user-centric identity management system which emphasizes the balance between security and privacy during authentication and authorization. Users can selectively disclose any combination of attributes and prove relations over them instead of being identified. Several attributes proof protocols are proposed to prove simple and complex logic over multiple attributes. However, the cryptographic building blocks to construct these protocols generally of large size of public parameter and signature, and also the complexity w.r.t. the concrete number of pairing and exponentiation operation is too large to be practical in source-limited devices. In this paper, we focus on reducing the complexity and the signature size in attributes proof procedure. We firstly demonstrate the application scenario, and then give an extended Boneh-Lynn-Shacham short signature scheme to construct an efficient pairing-based credential. We also describe the protocol of attributes proof on CNF relation, where the complexity is linear with the number of clauses instead of the attributes embedded in the credential or specified in the security policy.

Keywords: electronic identity system, privacy, attributes proof, short signature

1 Introduction

With the development of online and offline applications, electronic identity system has become more and more important to daily life. The common to online or offline service is that users have to be authenticated by the service provide before using services [16]. A user has to provide his/her information to the service provide during authentication, generally revealing his/her sensitive information such as birth date, credit card number and social security number. Privacy leak to adversaries and service provider insiders could put users to be traced, make users' transactions to be linked, and then profile them in a misbehavior way. Due to such problem, electronic identity system highlights the importance of the privacy protection. Users can just reveal the necessary information to the service provider to satisfy minimum information disclosure principle.

To meet the privacy protection requirement, group signature [10], attribute-based signature [13, 14, 18, 20] and blind signature [4] are building blocks to construct the eID system. In [1, 2, 4, 7, 8],Camenisch and Lysyanskaya adopt the group signature based on bilinear pairings and zero knowledge proof techniques to construct an anonymous credential system. This system shows good feature of anonymity and unlinkability. IBM [15] proposed a credential system called Idemix which is applied in identity management. Brands [4] uses the blind signature scheme to construct a private credential system. Users can selectively demonstrate the possession of attributes to a service provider. There are three types of

Research Briefs on Information & Communication Technology Evolution (ReBICTE), Vol. 2, Article No. 11 (September 15, 2016)

*Corresponding author: Computer Science and Engineering College, Northeastern University, Shenyang, Liaoning, 110819, China.

disclosure: no disclosure, partial disclosure and disclose all its properties. However, the blind signature scheme cannot handle multi-showing association. In addition, the credential assumes that the user has a master key and dependents on the PKI.

In the real application scenario while authentication, it is common that users just need to prove that they belong to a statistical group, while do not need to disclose the actual identity. For example, CNF complex logic relation [17, 19] consists of some OR clauses. Each clause represents the proof for possession of one of multiple attributes. In [5, 12], the user can prove that he is either a staff or a teacher when using a copy machine in a laboratory. In this case, the privacy property of anonymity and unlinkability can be obtained by using attribute-based credentials. However, the existing credential systems [1, 2, 4, 7, 8] use cryptography algorithms with linear complexity with the number of attributes which are either embedded in the credential or specified in the security policy. Precisely, the number of exponentiation and pairing will be increased in linear. Considering that the eID is generally applied in the source-limited devices such as smart card and smart phone, the existing attributes proof on CNF relation is far away from being practical.

In [6], the cryptography accumulator is used to solve linear complexity problem in attributes proof. Given an attribute from a finite set, the exponentiation operation is transformed into multiplication, thus output a constant size value. However, the accumulator scheme is not that suit for the string type attributes. Moreover, the verification of the validity of accumulator also needs some extra pairings. The public key size is also getting larger with the number of attributes increasing.

In order to reduce the complexity of the CNF attributes proof, the paper proposes a credential system based on the extended BLS short signature [3]). With the comparison with the accumulator scheme, the proposed protocol can prove AND relation over attributes in constant complexity with respect to the number of exponentiation and pairing, while the complexity of OR relation proof is reduced to depend on the number of OR clauses in stead of attributes embedded in the credential or specified in the security policy.

2 Preliminaries

In this section, we give the cryptographic preliminaries about bilinear maps and zero-knowledge proof which are the building blocks to construct a credential system and attributes proof protocols. The notation in the paper is as follows.

(1) G_1 and G_2 are two (multiplicative) cyclic groups of prime order p;

(2) g_1 is a generator of G_1 and g_2 is a generator of G_2.

2.1 Bilinear maps [9, 11]

A bilinear pairing e is defined if the mapping $e : G_1 \times G_2 \to G_T$ satisfies the following properties:

- **Bilinear** for all $u \in G_1$,$v \in G_2$ and a, $b \in Z$, $e\left(u^a, v^b\right) = e\left(u, v\right)^{ab}$.
- **Non-degenerate** $e\left(g_1, g_2\right) \neq 1$.
- **Computability** there exists an efficient algorithm to compute $e(g_1, g_2) \in G_T$.

2.2 Zero-knowledge proof

Zero-knowledge proof can ensure that the verifier does not get the information in the process of interaction. We use the method of Camenisch-Stadler notation [5] to show the expression of knowledge proof about the discrete logarithm problem. Precisely, the prover randomly selects a number x from the Z_p, which is used as the private key. We also set $y = g^x$ as the public key. Running the zero-knowledge proof protocol, the prover can testify the proof statement without revealing the value of x. We denote as $PK\{\alpha : y = g^\alpha\}$.

The process of the zero-knowledge proof protocol is described as follows.

Step 1. The prover randomly selects a value t from Z_p.

Step 2. The prover computes $T = g_1^t$ as a commitment, and send T to the verifier.

Step 3. The verifier randomly selects a value c from Z_p as a challenge, and sends c to the prover.

Step 4. The prover computes $s = xc + t \ (mod\ p)$ as a response, and sends s to the verifier. Finally, the verifier tests the equation $g^s = y^c T$. If true outputs $success$; otherwise, output $failure$.

3 The attribute model

The attribute model is the basic component to the eID system and attributes proof procedures. In this section, the classification, the formalized definition, and the structure of attributes are given.

3.1 Classification of attribute

Considering the data type of attributes in eID systems, it mainly consists of string, integer, and finite set. For example, $first\ name$ and $last\ name$ is of string, $credit\ number$ is of integer, and $gender$ is of finite set. According to the surveyed data sets for electronic identity cards and driver's license cards, only a minority of attributes are generic string or integer, whereas most attributes are taken from a finite set of discrete values.

Considering the privacy property of attributes in eID system, it consists of strong attributes and weak attributes. A strong attribute uniquely identifies an individual in a population, whereas a weak attribute can be applied to many individuals in a population. Whether an attribute is strong or weak depends upon the size of the population and the uniqueness of the attribute. The value of a strong attribute should be kept secret and not open in the clear to any parties other than the issuer and the user herself, while the value of a weak attribute is of less sensitive and can be disclosed as necessarily required to the relying parties. Generally, weak attributes are practical enough for attribute-based authentication or group authentication for privacy sensitive services. Weak attributes have lower sensitive to the privacy, such as $gender$ and $nationality$.

3.2 Formalized definition of attribute

In this paper, we mainly considering the dominated attribute type of finite set. An attribute is defined as a $\langle label, value, credential\_reference\rangle$ tuple. The $label$ represents the unique name of an attribute. The attribute's $value$ comes from a finite set. The $credential\_reference$ represents the authority-issued credential's identifier which makes attributes verifiable. For example, one person can enjoy the free tickets with his ID-card only if his minority is blind or social benefit is unemployed or the type is kids_card, i.e., she needs to prove she possesses these certified attributes $\langle minority, blind, sig_{ID}\rangle$, $\langle social\_benefit, unemployed, sig_{ID}\rangle$, and $\langle type, kids\_card, sig_{ID}\rangle$, where the $credential\_reference$ is represented by the signature on the value of attribute, i.e, sig_{ID}, under the ID-card issuer's private key.

3.3 Structure of attribute-based credential

According to the attribute definition as a tuple $\langle label, value, credential\_reference\rangle$, we denote the attributes with the corresponding cryptographic information as the structure of attribute-based credential based on the Identity Mixer. $AttributeOrder$ is to control the sequences of the attributes. $credential\_reference$ associates the attributes and the corresponding signature.

$Attributes\{$
    $Attribute : (label_1, type_1),$
    $\ldots$
    $Attribute : (label_n, type_1)$
$\}$
$Implementation\{$
    $AttributeOrder : (label_1, \cdots, label_n)$
$\}$
$Elements\{$
    $credential\_reference : (CR_1, \cdots, CR_n)$
    $values : (value_1, \cdots, value_n)$
$\}$

4 The construction of eID system

A credential system usually consists of three entities which are the issuer, the prover, and the verifier. In practical, the issuer generally is an institution has authority and credibility. The prover is an individual user, the verifier represents service providers. Based on this, our credential system involves the roles of the issuer, recipient, the prover and the verifier. Also, our credential system consists of two protocols, issuing protocol a proving protocol. Besides, based on the attribute knowledge, we give the process of CNF proof. The eID App is deployed in individuals' end device like smart phone, which is the personal identity management tool for authentication, access control, and other security applications. It integrates the individual's identifiers, attributes and credentials all together and acts as the trusted third-party to provide identity verification to the online and offline services.

4.1 The scenario of eID system and CNF relation proof over attributes

A scenario of CNF relation proof over attributes is depicted as 1. Given a movie streaming service which offers subscribers the service of movie, an individual Alice who possesses an eID App on her smart phone is requesting for a particular movie. The movie streaming service provider, short for SP, allows admission only to subscribers with valid identity certificate and the age over 16 years old. The eID App acts as an intermediate to check if Alice's attributes satisfy the SP's policy as follows.

Step 1. Alice requests for a particular movie from the SP.

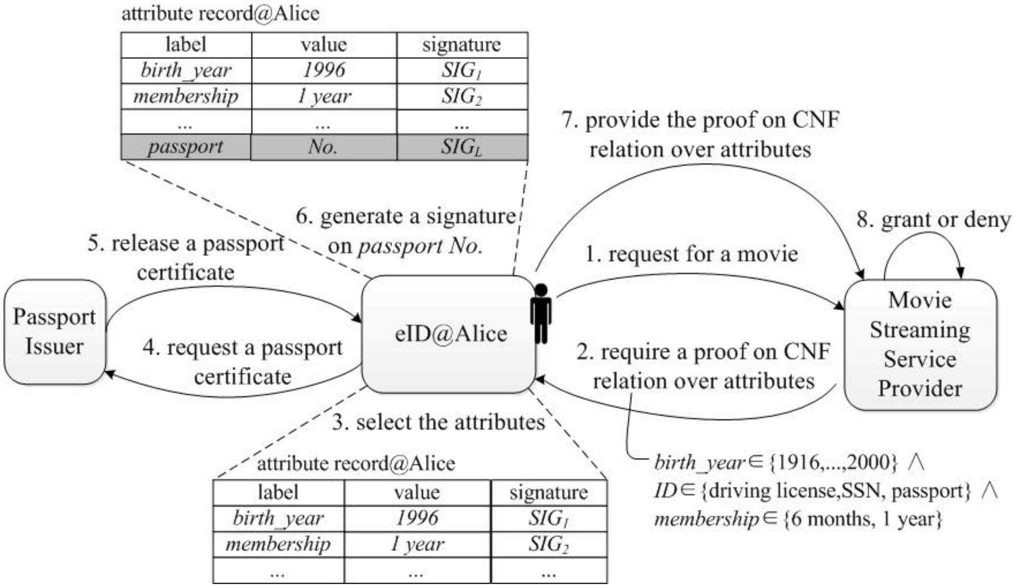
Step 2. As the SP receives the request, it redirects Alice to the eID App with the policy to generate an attributes proof. In this case, suppose the requested movie is accessed by users who has subscribed 6 *months* or 1 *year* membership, hold a valid identity certificate such as *driving license*, *SSN*, or *passport*, and also been over 16 years old, the relation over attributes is represented as a CNF formular, i.e., $birth\_year \in \{1916, \cdots, 2000\} \wedge ID \in \{driving\ license, SSN, passport\} \wedge membership \in \{6\ months, 1\ year\}$.

Step 3. The age and membership is respectively represented by the attribute of *birth_year* and *membership* which have been already created in the attributes record. However, not any identity certificate is found.

Step 4. On behalf of Alice, the eID App requests a passport certificate from the authority, i.e., ID Issuer.

Step 5. By some way the ID Issuer authenticates Alice and then releases a passport certificate if succeed.

Step 6. The eID App checks the validity of the passport certificate, and then generates a signature on it under the private key of the eID system if succeed. The attributes record is also updated with this newly


attribute record@Alice
7. provide the proof on CNF relation over attributes
8. grant or deny
6. generate a signature on *passport No.*
5. release a passport certificate
1. request for a movie
Passport Issuer
eID@Alice
Movie Streaming Service Provider
4. request a passport certificate
2. require a proof on CNF relation over attributes
3. select the attributes
attribute record@Alice
birth_year∈{1916,...,2000} ∧
ID∈{driving license,SSN, passport} ∧
membership∈{6 months, 1 year}

attribute record@Alice

label	value	signature
birth_year	*1996*	SIG_1
membership	*1 year*	SIG_2
...	...	...
passport	*No.*	SIG_L

attribute record@Alice

label	value	signature
birth_year	*1996*	SIG_1
membership	*1 year*	SIG_2
...	...	...

Figure 1: The scenario of CNF relation proof over attributes in eID system

created entry.

Step 7. According to the CNF relation over attributes derived from the SP's policy, the eID App firstly retrieves the related attributes, i.e., $birth\_year$, $membership$, and ID, and then generates a proof.

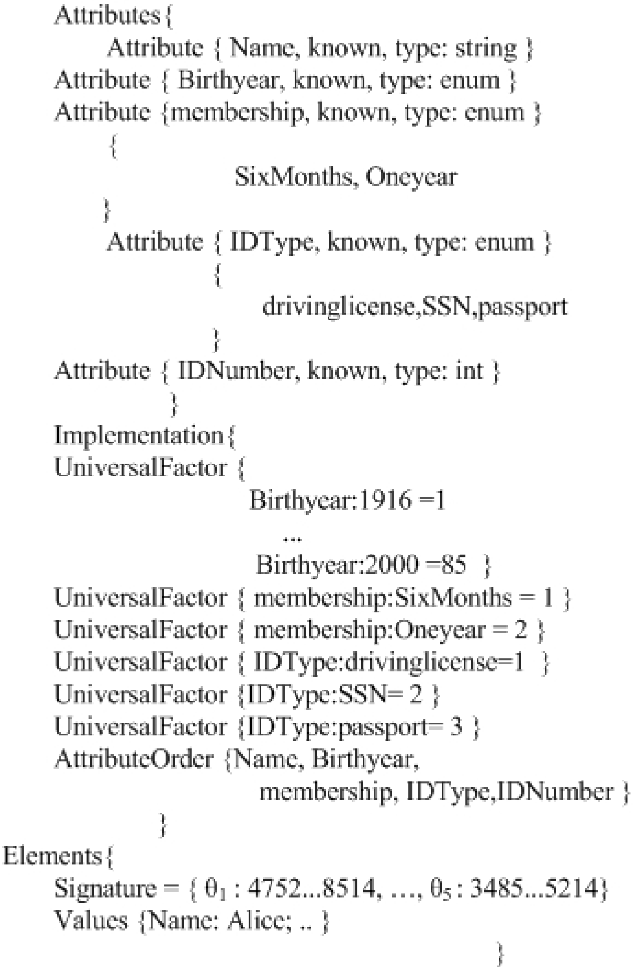
Step 8. The SP checks the validity and ownership of attributes through the verification of the proof. If succeed, it will grant the movie to Alice; otherwise, deny the request.

During the above attributes proof procedure in the eID system, the user's identity is kept secure and privacy-preserving.

Security Property. For confidentiality, the value of an attribute is kept secret within the issuer and the user herself; it will never be disclosed to any service providers and even to the eID service provider either, instead who can only determine if some of the user's attributes satisfy the particular policy or not. For integrity, the valid proof can be only generated on the issuer-certified attribute and by the user herself; any identity forgery cannot succeed. For availability, the attributes record is maintained only at the end user but backup on the eID server in an encryption fashion.

Privacy Property. In some identity sensitive scenarios, the attributes proof procedure meets such privacy requirements as anonymity and minimum disclosure of information. For anonymity, neither any service provider nor the eID service provider can identify the user; they cannot either link the user's multiple service requests all together even if they collude. For minimum disclosure of information, the user can selectively disclose any attribute or prove any relation over multiple attributes as necessarily required and keep others hidden.

The structure of attribute-based credential in the eID system is depicted as Fig 2. Due to the tuple <label,value,credential_reference>, we partition the attribute credential model into the attribute information part, implementation of attribute part and the credential's structure part. The attribute information



```
        Attributes{
            Attribute { Name, known, type: string }
        Attribute { Birthyear, known, type: enum }
        Attribute {membership, known, type: enum }
            {
                        SixMonths, Oneyear
            }
            Attribute { IDType, known, type: enum }
                    {
                            drivinglicense,SSN,passport
                    }
        Attribute { IDNumber, known, type: int }
                    }
        Implementation{
        UniversalFactor {
                            Birthyear:1916 =1
                              ...
                             Birthyear:2000 =85  }
        UniversalFactor { membership:SixMonths = 1 }
        UniversalFactor { membership:Oneyear = 2 }
        UniversalFactor { IDType:drivinglicense=1  }
        UniversalFactor {IDType:SSN= 2 }
        UniversalFactor {IDType:passport= 3 }
        AttributeOrder {Name, Birthyear,
                             membership, IDType,IDNumber }
                    }
    Elements{
        Signature = { θ1 : 4752...8514, ..., θ5 : 3485...5214}
        Values {Name: Alice; .. }
                                                    }
```

Figure 2: The specific example of attribute credential model

contains attribute's name, issuance mode and each type of attribute. The implementation of attribute part includes the attribute encoding and the order of attributes. As shown in Fig 2, adopt universal attributes set, marked *index* $\{1,...,n\}$, and serial encoding method. The credential's structure consist of the signatures and the values of the attributes.

Attribute-based credentials are used for logic relation proof. In our scheme, the credentials consist of three parts: the ids which could represent you are the holders. the ids are different for the every credential holders. We use M to describe each id. Each $M = g_1^r$ where the r is chosen by the every user. And only the credential owner knows its value. The second parts are the sets of all the values of attributes. We adopt $ATTR$ to represent all the values of user's attributes certified by the Issuer. Attributes encoding using index form, given specific attribute types, attribute values and the index of universal attributes set, we mark *index* $\{1,...,n\}$. It contains all the types of attributes values. And the third parts are the corresponding signatures. The signatures are based on the attribute values, i.e, $\{\theta_1,...,\theta_N\}$.Our credential *Cred* which is formed as $(M,ATTR,\{\theta_1,...,\theta_N\})$.

Algorithm 1 credential_generation

```
Input
  $x \in G_1$ : The private key generated by KeyGen() algorithm
  $v$ : The public key generated by KeyGen() algorithm
  $M$ : The ownership binder of credential
  $ATTR\{m_1,...,m_N\} \in \{0,1\}^*$: The set of attributes

Output
  $\sigma_i\{\sigma_1,...,\sigma_N\}$ : The credential of attributes $i$
  $//Initialization$
  Dim $p, g_1, g_2, m$ As Element;
  Dim $G_1, G_2, G_T, Z_p$ As Field;
  Dim $e$ As Pairing; $//e : G_1 * G_2 \to G_T$
  if $(PK\{\alpha : M = g^{\alpha}\})$ then
      for $i = 1$ $to$ $N$ do
          $m \leftarrow ATTR[i].getAttribute$;
          $h_i \leftarrow H(m)$; $//H : \{0,1\}^* \to G_1$
          $\sigma_i \leftarrow (h_i \cdot M)^x$;
      end for
  end if
```

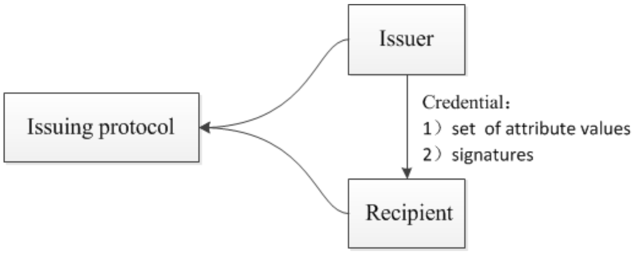


Figure 3: The issuing process

4.2 Protocols of Attributes Proof on CNF Relation

4.2.1 Issuing protocol

As shown in Fig 3, the issuing protocol is carried out between the issuer and a recipient who requests the credential. The credential is made of the set of attributes values, the corresponding signatures, and some cryptographic information that allows the owner of the credential to create a proof of possession. Due to the discrete logarithm problem, it is sufficient for the issuer to know the commitment M instead of the actual id r.

For CNF relation over attributes, it is to prove that several OR clauses are all satisfied the policy. We can construct the math model as follows.

Input: To the specific CNF formula φ : $(a_{11} \vee a_{12} \vee ...) \wedge (a_{21} \vee a_{22} \vee ...) \wedge ...$, every attribute is predefined in the universal sets, we use n to indicate the maximum index of attributes. It could be described as $\{1,2,...,n\}$. $a_{11}, a_{21}, ...$ all of these attributes stand for the verifier hold the index of these attributes. We also use U to represent the index of the prover's attributes. Besides, we set every OR clause as: $V_1 = (a_{11} \vee a_{12} \vee ...)$, $V_2 = (a_{21} \vee a_{22} \vee ...)$.

Output: If the attributes satisfied the CNF formula, output $true$; otherwise output $false$.

The above process of the formal definition could be described as: $Proof_{CNF}\,(n,\varphi,U) \to \{true, false\}$.

For each single OR clause, the prover needs to prove that one of the attributes is embedded in the credential through the form of a signature. We define two universal attributes sets $ATTR$ and TA, where $ATTR$ is a component of credential made up of the attribute values and the TA is the attribute set for the specific formula. The credential is defined as $Cred = (M,ATTR,\sigma)$, where M is a commitment as the binder of the prover. The issuing process consists of three steps as follows.

Step 1. Given the public parameters $(p,G_1,G_2,G_T,e,H,g_1,g_2,v)$ where a full-domain hash function is defined as $H:\{0,1\}^* \to G_1$, the recipient randomly selects a value $r \in_R Z_p$ and computes a commitment $M \leftarrow g_1^r \in G_1$.

Step 2. Given the signing key x and the attribute's value $m \in \{0,1\}^*$, the issuer computes $h \leftarrow H(m), \sigma \leftarrow h^x \cdot M^y \in G_1$, then output σ as a signature on an attribute.

The pseudocode is described in Algorithm 1.

4.2.2 Proving protocol

Before we introduce the proving protocol, we give some definitions as follows.
1. We use a_{ij} to stand for the ith attribute get jth value.
2. In universal set TA, marked as $Index\{1,\cdots,n\}$ where i from 1 to k, and j range from 1 to l, $kl = n$.
OR clauses are the basis element of the CNF formula. To each OR clause V_i, the prover should prove one of the attributes in the universal set TA. We adopt the OR relation attribute proof as the basic operation. Similarly, the proving process could be made up of five steps:

Step 1. The preparation for proving process, the public parameters $(p,G_1,G_2,G_T,e,H,g_1,g_2,v)$ are given. We also use a full-domain hash function $H:\{0,1\}^* \to G_1$. We carry on the bilinear pairing operation $e\,(g_1,v)$, marked as V_c.

Step 2. For each V_i, according to the issuing protocol, the prover input the corresponding signature σ. Then choose a random value $r^{'}$ from Z_p. Set the blinded signature $\sigma' \leftarrow \sigma^{r^{'}}$ and send the σ' to the verifier.

Step 3. To the verifier, firstly, according to m_i from universal attribute sets TA pre-compute the digest $h_{ij} \leftarrow H\,(a_{ij})$. Then, executing the bilinear pairing operation $V_{ij} = e(h_{ij},v)$. For every V_i, the verifier get the result $V_{i1},...,V_{il}$.

Step 4. For each clause, if the prover's attribute equal to the policy, the verifier according to $\sigma^{'}$, set $V_s = \left(\sigma^{'},g_2\right)$.

Step 5. The prover want to prove his/her attributes satisfy the corresponding policy. Based on the OR clause, the prover and the verifier carry out zero-knowledge proof:
$PK\left\{(\alpha_1,\alpha_2,\cdots,\alpha_l,\beta): V_s = V_{i1}^{\alpha_1},\cdots,V_{il}^{\alpha_l}V_c^{\beta}\right\}$. If it is true then output $success$; otherwise output $failure$.

In Fig 4 , proving protocol is executing between the prover and the verifier. In our eID system ,related to the issuing process, we define the participant holding the certificate as the prover. Many relevant attributes require multiple certificates, for example, in complex logic relation CNF, we should meet the one of the multiple attributes combination. Only the proof involving these certificates, we claim the prover meet the requirement.

In more concrete terms, when the prover's proof which including the attribute−related cryptographic information is transmitted to the verifier, the verifier based on the policy and PK to verify the result. At the same time, due to the blind signature technology, the verifier is unable to get the actual information of the prover. The pseudocode is showed in Algorithm 2. The complexity of the CNF relation proof is originally $O(LN)$. For balancing privacy and efficient, once the literals in each OR clause are reduced to k where $k \ll N$, the complexity will be linear in $O(kN)$; at the meanwhile, the advantage of inferring

Algorithm 2 proof_ generation

```
Input
    $x \in G_1$ : The private key generated by KeyGen() algorithm
    $v$ : The public key generated by KeyGen()algorithm
    $M$ : The ownership binder of credential
    $ATTR\{m_1,...,m_N\} \in \{0,1\}^*$: The set of attributes
    $TA\{a_{11},a_{12}...,a_{NL}\} \in \{0,1\}^*$: The set of attributes
    $\sigma_i\{\sigma_1,...,\sigma_N\} \in G_1$ : The credential of attributes $i$

Output
    $(T,s)$: The commitment and response
    $//Initialization$
    Dim $p,g_1,g_2,m[N],a[N][L],c,x[N][L],t[N][L],V_c,,V_s[N],V[i][j]$ As Element;
    Dim $G_1,G_2,G_T,Z_p$ As Field;
    Dim $V_c$ As Element;
    Dim $e$ As Pairing; $//e : G_1 * G_2 \to G_T$
    $V_c \leftarrow e(g_1,v)$;
    for $i = 1$ $to$ $N$ do
        $m[i] \leftarrow ATTR[i].getAttribute$;
        for $j = 1$ $to$ $L$ do
            while $((m[i] == a[i][j])$ do
                $r \leftarrow Z_p.Rand(1,p-1)$;
                $\sigma^{'} \leftarrow \sigma_i^r$;
                $V_s[i] \leftarrow e(\sigma^{'},g_2)$;
            end while
            $//generate\ the\ blind\ signature$
            $a[i][j] \leftarrow TA[i][j].getAttribute$;
            $h \leftarrow H(a[i][j])$;
            $V[i][j] \leftarrow e(h,v)$;
        end for
    end for
    $//generate\ proof$
    for $i = 1$ $to$ $N$ do
        for $j = 1$ $to$ $L$ do
            $t[i][j] \leftarrow Z_p.Rand(1,p-1)$;
        end for
        $t[0][0] \leftarrow Z_p.Rand(1,p-1)$;
        $T[i] \leftarrow \prod_{j=1}^{L} V[i][j]^{t[i][j]} \cdot V_c^{t[0][0]}$; $//commitment$
        $//PK\{x_i : M = V_i^{x_i}\}$
        for $j = 1$ $to$ $L$ do
            $x[i][j] \leftarrow Z_p.Rand(1,p-1)$;
            //suppose challenging $c \varepsilon Z_p$ are given
              $s[i][j] \leftarrow x[i][j] \cdot c + t[i][j]$; $//response$
        end for
        $x[0][0] \leftarrow Z_p.Rand(1,p-1)$;
        $s[0][0] \leftarrow x[0][0] \cdot c + t[0][0]$; $//response$
        $proof \leftarrow (T,s)$;
    end for
```

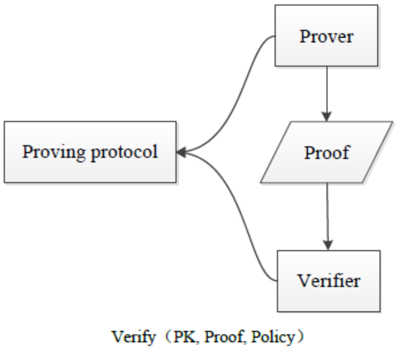


Figure 4: The proving process

which attribute in OR clause is satisfied is $1/k$ and the advantage of inferring which combination of attribute in CNF relation are satisfied is $1/k^N$.

5 Conclusion

Attributes proof is the key function in eID systems. The paper demonstrates the application scenario of the proposed eID system and attributes proof procedure. Considering that eID is convenient to be applied in the resource-limited devices such as smart card and smart phone, to reduce the complexity of attributes proof on CNF relation will make eID system more practical. Due to weak attributes occupying the majority proportion, lowering the complexity of weak attributes proof is thus to improve the overall efficiency. The paper proposed a pairing-based credential system based on the extended BLS short signature scheme, in which the size of public parameters and signature is short and the concrete number of exponentiation and pairing is small. The complexity of the proposed attributes proof on CNF relation protocol is linear with the number of OR clause in the CNF relation instead of the number of attributes embedded in the credential or specified in the security policy, thus to increase the efficiency.

Acknowledgments

This work was supported by the China Natural Science Foundation [Grant number 61402095] and [Grant number 61300196].

References

[1] N. Akagi, Y. Manabe, and T. Okamoto. Anonymous credentials light. In *Proc. of the 12th International Conference on Financial Cryptography and Data Security (FC'08), Cozumel, Mexico*, volume 5143 of *Lecture Notes in Computer Science*, pages 272–286. Springer Berlin Heidelberg, January 2008.

[2] F. Baldimtsi and A. Lysyanskaya. Anonymous credentials light. In *Proc. of the 2013 ACM SIGSAC conference on Computer & Communications Security (CCS'2013), Berlin, Germany*, pages 1087–1098. ACM, November 2013.

[3] D. Boneh, B. Lynn, and H. Shacham. Short signatures from the weil pairing. *Journal of Cryptology*, 17(24):297–319, September 2004.

[4] S. Brands. Rethinking public key infrastructure and digital certificates. *Journal of Urban Technology*, 8:143–145, 2000.

[5] J. Camenisch and T. Groß. Efficient attributes for anonymous credentials. In *Proc. of the 15th ACM conference on Computer and Communications Security (CCS'08), Alexandria, VA, USA*, pages 345–356. ACM, October 2008.

[6] J. Camenisch, M. Kohlweiss, and C. Soriente. An accumulator based on bilinear maps and efficient revocation for anonymous credentials. In *Proc. of the 12th International Conference on Practice and Theory in Public Key Cryptography (PKC'09), Irvine, CA, USA*, volume 5443 of *Lecture Notes in Computer Science*, pages 246–263. Springer Berlin Heidelberg, March 2009.

[7] J. Camenisch and A. Lysyanskaya. Efficient non-transferable anonymous multi-show credential system with optional anonymity revocation. In *Proc. of the 2001 International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT'01), Innsbruck, Austria*, volume 2045 of *Lecture Notes in Computer Science*, pages 93–118. Springer Berlin Heidelberg, May 2001.

[8] J. Camenisch and A. Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. In *Proc. of the 24th Annual International Cryptology Conference (CRYPTO'04), Santa Barbara, California, USA*, volume 3152 of *Lecture Notes in Computer Science*, pages 56–72. Springer Berlin Heidelberg, August 2004.

[9] J. Camenisch and M. Stadler. Proof systems for general statements about discrete logarithms. Technical Report 260, ETH Zurich, 1997.

[10] B. Dan, X. Boyen, and H. Shacham. Short group signatures. In *Advances in Cryptology - Proc. of the 24th Annual International Cryptology Conference (CRYPTO'04), Santa Barbara, California, USA*, volume Lecture Notes in Computer Science of *3152*, pages 227–242. Springer Berlin Heidelberg, August 2004.

[11] B. Dan, C. Gentry, L. B, and et al. Aggregate and verifiably encrypted signatures from bilinear maps. In *Advances in Cryptology — Proc. of the 2003 International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'03), Warsaw, Poland*, volume 2656 of *Lecture Notes in Computer Science*, pages 416–432. Springer Berlin Heidelberg, March 2003.

[12] D. G. Feng and C. Chen. Research on attribute-based cryptography. *Journal of Cryptologic Research*, 1(1):1–12, 2014.

[13] E. Ghadafi. Stronger security notions for decentralized traceable attribute-based signatures and more efficient constructions. In *Proc. of The Cryptographers' Track at the RSA Conference 2015 (CT-RSA'15), San Francisco, CA, USA*, volume 9048 of *Lecture Notes in Computer Science*, pages 391–409. Springer Berlin Heidelberg, April 2015.

[14] J. Herranz, F. Laguillaumie, B. Libert, and C. Ràfols. Short attribute-based signatures for threshold predicates. In *Proc. of The Cryptographers' Track at the RSA Conference 2012 (CT-RSA'12), San Francisco, CA, USA*, volume 7178 of *Lecture Notes in Computer Science*, pages 51–67. Springer Berlin Heidelberg, February-March 2012.

[15] IBM Research Security Team - Zurich. Specification of the Identity Mixer Cryptographic Library, v. 2.3.0*. Research Report RZ 3730, April 2010. `http://domino.research.ibm.com/library/cyberdig.nsf/papers/EEB54FF3B91C1D648525759B004FBBB1/$File/rz3730_revised.pdf`.

[16] S. Kiyomoto and T. Tanaka. Anonymous attribute authentication scheme using self-blindable certificates. In *Proc. of the 2008 IEEE International Conference on Intelligence and Security Informatics (ISI'08), Taipei, Taiwan*, pages 215–217. IEEE, June 2008.

[17] S. Sadiah, T. Nakanishi, K. Watanabe, and N. Funabiki. A proposal of extended anonymous credential scheme with efficient proof of age inequality. *Technical Report of IEICE. ISEC*, 113(217):21–28, September 2013.

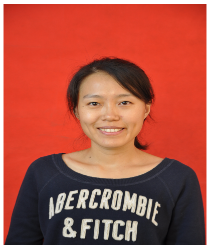
[18] J. Su, D. Cao, B. Zhao, X. Wang, and I. You. ePASS: An expressive attribute-based signature scheme with privacy and an unforgeability guarantee for the internet of things. *Future Generation Computer Systems*, 33(2):11–18, 2014.

[19] A. Sudarsono, T. Nakanishi, and N. Funabiki. Efficient proofs of attributes in pairing-based anonymous credential system. In *Proc. of the 11th International Symposium on Privacy Enhancing Technologies (PETS'11), Waterloo, ON, Canada*, volume 6794 of *Lecture Notes in Computer Science*, pages 246–263. Springer Berlin Heidelberg, July 2011.

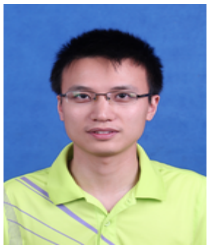
[20] Y. Zhang and D. Feng. Efficient attribute proofs in anonymous credential using attribute-based cryptography.

In *Proc. of the 14th International Conference on Information and Communications Security (ICICS'12), Hong Kong, China*, volume 7618 of *Lecture Notes in Computer Science*, pages 408–415. Springer Berlin Heidelberg, October 2012.

Author Biography



Nan Guo received the BE in Computer Science and Technology, the ME and the PhD in Computer Application Technology, from Northeastern University, China, in 1999, 2001, 2005 respectively. She joined Northeastern University, China, in September 2005. She has been an associate professor since 2008. She has been a visiting scholar at department of Computer Science, Purdue University, USA, from August 2010 to August 2011. Her research interests are security and privacy in social network and digital identity management.



Wenwei Jiang received the BE degree in Computer Science and Technology, from City Institute, Dalian University of Technology in 2012. Currently he is taking a master's course in Computer Application Technology, from Northeastern University, China.

Jiayi Ouyang received the BE degree in Optical Information Science and Technology, from HeFei University of Technology in 2012. Currently he is taking a master's course in Computer Technology from Northeastern University, China.

Tianhan Gao received the BE in Computer Science and Technology, the ME and the PhD in Computer Application Technology, from Northeastern University, China, in 1999, 2001, 2006 respectively. He has been an associate professor since January 2010. He has been a visiting scholar at department of Computer Science, Purdue University, USA, from February 2011 to February 2012. His research interests are MIPv6/HMIPv6 security, wireless mesh network security, Internet security, security and privacy in ubiquitous computing, and virtual reality.

Bin Zhang was born in 1964. He is a professor in the College of Computer Science and Technology at Northeastern University, Shenyang, China. He is a senior member of China Computer Federation (CCF). He received his Ph.D. degree from Northeastern University in 1997. His current research interests include service oriented computing and information retrieval.