https://doi.org/10.52326/ic-ecco.2022/SEC.03

Attribute-Based Encryption for Weighted Threshold Access Structures

Alexandra-Ina BUTNARU [1], ORCID: 0000-0001-8180-7727
[1] Faculty of Computer Science, "Alexandru Ioan Cuza" University of Iași, 16 General Henri Mathias Berthelot Street, Iași, alexandraibutnaru@gmail.com

***Abstract*— Access control is a fundamental security component in a system, especially in rapidly developing domains such as the cloud or the Internet of Things. The nature of these domains renders formerly acclaimed access control techniques inefficient in environments that are distributed and need highly scalable solutions. Attribute-based access control offers a multitude of advantages, especially through its cryptographic implementation, attribute-based encryption. Weighted threshold access structures are structures that closely cover real-life scenarios and have high applicability in practice as access control policies.**

***Keywords*— access control; attribute-based access control; attribute-based encryption; weighted threshold access structures**

I. INTRODUCTION

Some of the most powerful and fast-paced technologies today, for example the internet of things (IoT) and the cloud, have created a need for stronger, faster, more flexible security solutions than the ones we have employed up until now. It is more common than ever for multiple applications to make use of the same data, a phenomenon which has led to a shift in the industry towards data-centric security and therefore data-centric access control.

The most widely used paradigms in the past, such as mandatory access control (MAC), discretionary access control (DAC) and more recently role-based access control (RBAC), are user-centric and cannot take into consideration contextual information like the time of the day or parameters like the relationship between the user and the resource for which access is requested. Furthermore, they have proven not to be enough when using technologies which employ a large number of devices or users and require speed, strong security and resource efficiency. Even RBAC, the most flexible and expressive out of the access control models mentioned, has been criticized for leading to role explosion and becoming unmanageable in large scale systems [1].

One of the main advantages attribute-based access control has over other access control solutions is the finer granularity it offers due to directly basing authorization on attributes that the requesting party holds.

An additional argument is the overall industry need for better security and the needs that new, distributed, big data technologies (IoT, wireless sensor networks, the cloud) have created, for example when device storage and power is severely limited or the system has highly unsafe components (like public servers and networks). As the number of data breaches and the number of records exposed increases, the money spent by companies to recover data encrypted by ransomware attacks goes well into billions each year.

II. PRELIMINARIES

A. Attribute-Based Access Control

One of the core concepts in information security is access control and attribute-based access control is a more powerful and flexible solution than other historically acclaimed techniques specifically because of the use of attributes.

Over time, two major solutions have developed in order to realize the ABAC concept: the first one consists of standardized languages like the eXtensible Access Control Markup Language (XACML) [2] (which is used to define access policies and requests to information), the second one entails using mathematics rather than software for enforcing access control and is called attribute-based encryption [3]. The cryptographic approach has a considerable advantage in flexibility over the other because the encrypted data contains the access control, meaning that it is not reliant on infrastructure and it also can be stored on any, secure or not, public or private servers .

Even though businesses have shied away from using ABAC until recently, the European Telecommunications Standards Institute (ETSI) has released two specifications ([4], [5]) on Attribute-Based Encryption (ABE) for access control and declared it a key enabler technology. By enforcing access control at a mathematical level, ABE is the better solution in terms of security than the ones enforcing it through techniques that rely on software.

The National Institute of Standards and Technology (NIST) has recommended certain key sizes for elliptic curve cryptography (ECC) and showed the cryptographic security each of them offers in comparison to the well-established RSA [6], further reinforcing the superior efficiency of ABE solutions.

B. Attribute-Based Encryption

Sahai and Waters introduced the concept of attribute-based encryption [7] in 2005 as a technique that allows ciphertexts to be encrypted under a collection of attributes and secret keys to contain authorization policies so that a user's private key will only be able to decrypt data that has been encrypted with the attributes that match their policy.

Up until recently attribute-based encryption solutions have mostly been either impractical, not secure or very limited in regards to the policies they can express. Lattice-based solutions, although secure, are infeasible in practice because of the expansion of both the decryption key and the ciphertext. As we do not have at the present time a secure multilinear map candidate, the preferable solution is using just one bilinear map over an elliptic curve in order to construct such schemes [8].

The main advantages of using ABE are inherently the ones offered by ABAC: decreased key size with an increased security and speed of computation, even on smaller, mobile devices with low computational power, and more flexible and expressive access control policies that closely model real-world situations.

Another crucial security aspect of attribute-based encryption is collusion-resistance: an adversary that holds multiple keys should only be able to access data if at least one individual key grants access.

One more advantage ABE has over other cryptographic protocols is its capability to provide data protection (e.g. compliance with General Data Protection Regulation - GDPR) by using attributes that do not disclose sensitive information like a person's name.

A. CP-ABE

Ciphertext-policy attribute-based encryption is the variant of ABE where messages get encrypted with policies under sets of attributes and the central authority distributes decryption keys to which the users' attributes are associated. A user will only be able to decrypt a ciphertext if his attributes satisfy the ciphertext's corresponding policy. The feature is called implicit authorization.

B. KP-ABE

Key-policy attribute-based encryption associates access control policies to the users' private keys, while the ciphertexts are encrypted under finite sets of attributes. A ciphertext's attributes must satisy a user's key's policy in order for the user to have access to the requested resource.

A key-policy attribute based encryption (KP-ABE) scheme consists of four probabilistic polynomial-time (PPT) algorithms [3]:

- Setup(λ): this is a PPT algorithm that takes as input the security parameter λ and outputs a master key MSK and a set of public parameters PP;
- Encrypt(m,A,PP): this is a PPT algorithm that takes as input a message m, the public parameters PP and a non-empty set of attributes A ⊆ U and outputs a ciphertext E;
- KeyGen(□,MSK): this is a PPT algorithm that takes as input the master key MSK and an access structure □ (given as a Boolean circuit) in order to output a decryption key D (for the entire Boolean circuit □);
- Decrypt(E,D): this is a deterministic polynomial-time algorithm that takes as input a ciphertext E and a decryption key D as described above and outputs the decrypted message m or the special symbol ⊥.

C. Access Control Structures

$\mathbb{Z}$ denotes the set of integers. A positive integer a > 1 is a prime number if its only positive divisors are 1 and a. Two integers a and b are called *congruent modulo n* (denoted $a \equiv b \bmod n$) if n divides $a-b$ (n is also an integer).

Let $\{P_1, \ldots, P_n\}$ be a set of elements called parties or participants, and $2^{\{P_1,\ldots,P_n\}}$ the set of all subsets of $\{P_1, \ldots, P_n\}$. A collection A is monotone if $\forall$ □, □: if □ $\in$ A and □ $\subseteq$ □ then □ $\in$ A. An access structure (respectively, monotonic access structure) is a collection (respectively, monotone collection) A of non-empty subsets of $\{P_1, \ldots, P_n\}$, i.e., $A \subseteq 2^{\{P_1,\ldots,P_n\}} \backslash \{\emptyset\}$. The sets in A are called the authorized sets, and the sets not in A are called the unauthorized sets.

In the following we refer to monotonic access structures whenever access structures are mentioned.

D. Weighted Threshold Access Structures

Weighted threshold access structures are a concept introduced by Shamir [9] that mirror scenarios where an authority wishes to share a secret between multiple parties so that particular subsets of those parties can recover the secret. The fragments of the secret that the aforementioned parties get are called shares. The secret can only be reconstructed if the weights of the parties surpass a threshold established by the authority sharing the secret [10].

Let U be the set of all attributes, a weight function $\omega: U \to N$, a threshold $T \in N$. Define $\omega(A) = \sum_{u \in A} \omega(u)$ and $\Gamma = \{A \subset U: \omega(A) \geq T\}$. Then Γ is called a weighted threshold access structure on U.

E. Bilinear maps

Number.

Let G and G_T be two multiplicative cyclic groups of prime order *p*. Let g be a generator of G and e be a bilinear map, $e: G \times G \to G_T$. The bilinear map e has the following properties:

1. Bilinearity: for all $u, v \in G$ and $a, b \in Z_p$, we have $e(u^a, v^b) = e(u, v)^{ab}$.
2. Non degeneracy: $e(g, g) \neq 1$.

We say that G is a bilinear group if the group operation in G and the bilinear map $e: G \times G \to G_T$ are both efficiently computable.

III. OUR CONTRIBUTION

The number of IoT and cloud applications has skyrocketed over the last years with even more promising (and daunting) predictions. Domains such as these have exhibited a need for faster, more flexible and more scalable security and access control solutions mainly because of their unprecedented size and distributed nature. ABAC, through its cryptographic implementation (ABE), is able to offer finer granularity, more expressive policies and higher security (through its data-centric approach) than previously widely-used solutions like RBAC.

As weighted threshold access structures are structures with many applications in practice, especially in expressing policies in attribute-based access control, finding an efficient solution to implementing them has been imperative.

We can achieve very good security, speed and resource efficiency by using bilinear maps over elliptic curves in order to construct schemes applicable for this type of access structures. In the following we present the cryptographic scheme in its two variants.

F. Scheme WAS_ABE_1

k represents the number of attributes. The compartment gate's threshold is the global threshold t, and *q* is the sum of all attributes' weights.

In order to share a value , where *p* is a prime, on a Boolean circuit $\boldsymbol{C}$ we are going to use a secret sharing procedure Share(*y*,$\boldsymbol{C}$):

- Assign *y* to the output wire of the circuit (the output wire of the (*t*,*q*)-gate);
- Choose uniformly at random and define the polynomial mod *p*. Then, assign to the input wires of the (*t*,*q*)-gate the shares f(1),…,f(*q*) in this order from left to right.

Now we introduce our scheme as follows:

Setup(λ, *n*): the algorithm uses the security parameter λ to choose a prime *p*, two multiplicative groups G and G_T of prime order *p*, a generator g of G, and a bilinear map $e: G \times G \to G_T$. Then, it chooses $y \in Z_p$ and, for each attribute i.j, chooses $r_{i.j} \leftarrow Z_p$. Finally, the algorithm outputs the public parameters $PP = (p, G, G_T, g, e, n, Y = e(g, g)^y, (T_{i.j} = g^{r_{i.j}} | \, i, j))$ and the master key $MSK = (y, r_{i.j} | i, j)$.

Encrypt(m, A, PP): the encryption algorithm encrypts a message $m \in G_T$ by a non empty set A of attributes as follows:

- $s \leftarrow Z_p$;
- Output $E = (A, E' = mY^s, (E_{i.j} = T_{i.j}^s = g^{r_{i.j}s} | i, j \in A), g^s)$.

KeyGen($\boldsymbol{C}$, MSK): he decryption key generation algorithm generates a decryption key D for the WAS defined by the Boolean circuit $\boldsymbol{C}$ as follows:

- $S \leftarrow Share(y, C)$;
- Output D, where $D(i.j) = g^{S(i.j)/r_{i.j}}, \forall 1 \leq i \leq k, 1 \leq j \leq \omega_i$.

Decrypt(E,D): given E and D as above, the decryption works as follows:

Compute $F_A(i.j)$ for all attributes i.j by

$$F_A(i.j) = \begin{cases} \{e(g,g)^{S(i.j)s}, & if\ i \in A \\ \perp, & otherwise \end{cases}$$

and ⊥ means "undefined";

- If the (*t*,*q*)-gate is satisfied (i.e., the global threshold is satisfied), then use the Lagrange interpolation formula to derive $O = e(g, g)^{ys}$ from the F_A-values. If the gate is not satisfied, then the value will be ⊥;
- $m = E'/O$.

If we do not multiply the attributes, but distribute multiple decryption keys to each one (as many as the attribute's weight) we can view the resulting scheme as more efficient as it keeps the number of circuit leaves.

G. Scheme ABE_WAS_2

The Setup, Encrypt and KeyGen algorithms perform identically with the ones in the first scheme.

Decrypt(E, D): given E as the result of the Encrypt algorithm and D as the result of the KeyGen algorithm, the decryption works as follows:

- Compute $F_A(i.j)$ for all decryption keys j of all attributes i by

$$F_A(i.j) = \begin{cases} \{e(g,g)^{S(i.j)s}, & if\ i \in A \\ \perp, & otherwise \end{cases}$$

and ⊥ means "undefined";

- If the (*t*,*q*)-gate is satisfied (i.e., the global threshold is satisfied), then use the Lagrange interpolation formula to derive $O = e(g,g)^{ys}$ from the F_A-values. If the gate is not satisfied, then the value will be $\perp$;
- $m = E'/O$.

H. Security

We prove the security of the second scheme. For the proof is analogous as the two schemes behave similarly.

Theorem 1. In the selective model and under the decisional bilinear Diffie-Hellman assumption the scheme is secure.

Proof. The main proof idea is by contradiction: we defined WAS_ABE_2 by transforming any weighted access structure into a CAS structure. So, if the WAS_ABE_2 scheme is not secure, the CAS_ABE scheme should not be either. Therefore, let us develop this main idea.

Suppose that, in the selective model, the WAS_ABE_2 scheme is not secure. Consider then an adversary **A** which has against this scheme a non-negligible advantage when it is applied to WASs. We define an adversary **A**' against the corresponding CAS_ABE scheme and we show that it has a non-negligible advantage against this scheme, which is a contradiction. If $\boldsymbol{\mathcal{C}}$ stands for a Boolean circuit in the WAS_ABE_2 scheme, then from a technical perspective the CAS_ABE scheme equivalent circuit is identical. To be able to differentiate between the two cases, let $\boldsymbol{\mathcal{C}'}$ denote the CAS_ABE scheme circuit. The adversary **A**' will:

- announce the set **A** of attributes that he wishes to be challenged upon;
- receive the public parameters PP during the Setup phase;
- query the decryption key generation oracle for a Boolean circuit $\boldsymbol{\mathcal{C}}$ representing some CAS. The adversary **A**' is granted access during Phases 1 and 2. The adversary can obtain the decryption key (D',P') as it is described in the CAS_ABE scheme by querying the oracle with $\boldsymbol{\mathcal{C}}$ (**A**) = 0. Consequently **A**' can compute $F_A = e(g,g)^{S(i.j)s}$, for all (i.j). At this point we can notice looking at the WAS_ABE_2 scheme that the F_A-values computed by **A**' are the same F_A-values that would have been computed by **A** if **A** had interrogated the key decryption oracle of the WAS_ABE_2 scheme with the circuit $\boldsymbol{\mathcal{C}}$. Taking note that $\boldsymbol{\mathcal{C}}$(**A**) = 0, the secret sharing would have happened in the same manner at the logic gates;
- submit two messages, m_0 and m_1, of equal length in the Challenge phase and receive **A**'s corresponding ciphertext and one of the two messages, say m_b, where $b \leftarrow \{0,1\}$.

A' can obviously compute in the worst case the same information adversary **A** can. As a consequence, adversary **A**' has a greater probability of correctly guessing b than **A**. We hence arrive at the contradiction that **A**' has a non-negligible advantage against the CAS_ABE scheme.

I. Implementation

The main challenges of implementing schemes like the ones presented above are working with elliptic curves and bilinear maps.

An implementation of the weighted threshold access structure scheme *ABE_WAS_1* has been written using Java 14 and is publicly available at https://github.com/alexandraib/abe-was. For bilinear map support, Ben Lynn's PBC library [11] offers an abstract interface and easy-to-use APIs to a cyclic group with a bilinear pairing. The implementation uses curve $y^2 = x^3 + x$ over the field F_q for some prime q=3 mod 4 and a billinear nondegenerate map $e: G \times G \rightarrow G_T$ where G and G_T are both cyclic groups of prime order r with q+1=rh and h is a multiple of 12 (in order to have $q \equiv -1 \; mod \; 12$).

The implementation uses r on 160 bits and q on 512 bits. It was tested on Windows 10.

IV. CONCLUSIONS

Weighted threshold access structures are highly important in practice due to their accuracy in the representation of situations where particular subsets of parties must be authorized and others must not, according to their importance.

These access structures are decidedly important because of the abundance of real-life situations that they can model and the way attribute-based access control can express them in order to regulate access and authorization.

In this paper we propose an attribute-based encryption scheme for weighted threshold access structures. To the best of our knowledge this is the first scheme dealing with this problem.

We have proof that our scheme is secure under the assumption of decisional bilinear Diffie-Hellman.

A remaining issue that requires solving and that would improve the scheme would be finding a procedure that condenses the partial secrets attached into a single attribute.

ACKNOWLEDGMENT

Sincere thanks go towards Prof. dr. Ferucio Laurenţiu Ţiplea from "Al. I. Cuza" University of Iaşi for the

suggestion of writing this paper and his patience, support and feedback throughout its development.

REFERENCES

[1] A. H. Karp, H. Haury, and M. H. Davis, "From abac to zbac: the evolution of access control models," Journal of Information Warfare, vol. 9, no. 2, pp. 38–46, 2010.

[2] S. Godik and T. Moses, "Oasis extensible access control markup language (xacml)," OASIS Committee Secification cs-xacml-specification-1.0, 2002.

[3] V. Goyal, O. Pandey, A. Sahai, and B. Waters,"Attribute-based encryption for fine-grained access control of encrypted data," in Proceedings of the 13th ACM conference on Computer and communications security, 2006, pp. 89–98.

[4] E.T.S.I., ETSI TS 103 458: "CYBER; Application of Attribute Based Encryption (ABE) for PII and personal data protection on IoT devices, WLAN, cloud and mobile services - High level requirements", 2018.

[5] E.T.S.I, ETSI TS 103 532: "CYBER; Attribute Based Encryption for Attribute Based Access Control", 2018.

[6] E. Barker, E. Barker, W. Burr, W. Polk, M. Smid et al., Recommendation for key management: Part 1: General. National Institute of Standards and Technology, Technology Administration . . . , 2006, pp. 54–55.

[7] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in Annual international conference on the theory and applications of cryptographic techniques. Springer, 2005, pp. 457–473.

[8] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in Annual international cryptology conference. Springer, 2001, pp. 213–229.

[9] A. Shamir, "How to share a secret," Communications of the ACM, vol. 22, no. 11, pp. 612–613, 1979.

[10] A. Beimel, T. Tassa, and E. Weinreb, "Characterizing ideal weighted threshold secret sharing," in Theory of Cryptography Conference. Springer, 2005, pp. 600–619.

[11] B. Lynn, "Pbc library manual 0.5. 11," 2006.