A^{3}BAC: Attribute-Based Access Control Model with Anonymous Access

Runnan Zhang[1], Gang Liu[1(✉)], Shancang Li[2], Yongheng Wei[1], and Quan Wang[1]

[1] School of Computer Science and Technology, Xidian University, Xi'an 710071, China
gliu_xd@163.com
[2] Department of Computer Science and Creative Technologies, University of the West of England, Bristol BS16 1QY, UK

Abstract. Researchers believe that anonymous access can protect private information even if it does not store in authorization organization. The current solution supports anonymous access by using a certificate instead of a subject identity or Attribute-Based Encryption. In a solution using a certificate, access may be linked to the certificate, which poses a risk of re-identification. The encryption of objects based on attributes limits the types of objects. The ABAC with anonymous access proposed in this paper called A^{3}BAC inherits the features of the ABAC model, such as fine-grained authorization, policy flexibility, and unlimited object types. By combining HABS, it strengthens the identity-less of ABAC, so that the access does not involve a unique identification, reducing the risk of subject identity re-identification. It is a secure anonymous access framework.

Keywords: Access control · ABAC · Anonymous access · HABS

1 Introduction

In Internet of Things (IoT) and Mobile Internet of Things (MIoT), there are many problems like forensics [1] and consensus algorithm [12]. However, anonymous access control is rarely studied in the MIoT environment [7]. Some scholars have studied the access control technology supporting anonymous authorization [9–11]. The methods mainly use attributes to encrypt objects to support anonymous authorization in the access control model.

Ahuja R. and Mohanty S. K. [3] proposed a scalable Attribute-Based Encryption (ABE) scheme in the cloud environment. The scheme generates a hierarchical attribute private key for users through hierarchical authority and hierarchical Ciphertext Policy Attribute-Based Encryption (CP-ABE) algorithm. Yuen T. H., Liu J. K., Man. H. A., et al. [2] proposed an anonymous ABAC model based on an anonymous certificate to support k-times anonymous authorization for cloud services. Given the serious harm of privacy disclosure in the Personal Health Records (PHR) system, Pussewalage H. S. G. and Oleshchuk V. A. [4] proposed an anonymous ABAC scheme based on ABE and proxy re-encryption to protect privacy.

D. Wang et al. (Eds.): SPNCE 2020, LNICST 344, pp. 441–447, 2021.
https://doi.org/10.1007/978-3-030-66922-5_30

The above method supports anonymous access by using a certificate instead of a subject identity or ABE. There are some defects in these methods. In some schemes, the certificate of the subject is unique, so the subject's access may be linked to its certificate. The attacker can re-identify the identity of the subject through the access linked to the subject. Once the re-identification is successful, it will cause unexpected privacy disclosure [5]. The encryption of objects based on attributes limits the types of objects. It is friendly to access control of objects that can be moved from server to client, such as video files. It is unfriendly to access control of objects that cannot be moved to the client, such as web services, which needs additional mechanisms to provide access control and increases the complexity of its implementation. And using an ABE-based algorithm to access objects often needs to download the object or generate tokens, which increases the load of the network.

In this paper, we propose an ABAC model that supports anonymous access called A^3BAC. By combining HABS (homomorphic attribute-based signatures) and transferring some functions of ABAC to the attribute authority and audit institution, we strengthen the identity-less of ABAC, so that its authorization is no longer dependent on identity. A^3BAC does not use unique certificates and is friendly for all types of objects. It inherits the features of fine-grained access control, flexible policy, and unlimited object type of ABAC. It overcomes the defect that the ABAC framework does not support anonymous access, and provides an audit that improves security. Extending the ABAC framework, giving it anonymous access and audit support, can expand ABAC's scope of application and improve its practicality [14].

Our Contribution:

1. This paper proposes an ABAC framework that supports anonymous access and fine-grained control of attributes by subjects.
2. The workflow of A^3BAC and the HABS algorithm that each entity needs to execute is elaborated.

2 Preliminary

2.1 ABAC Model

While there is currently no single agreed-upon model or standardization of ABAC, there are commonly accepted high-level definitions and descriptions of its function. One such high-level description is presented in [6]:

Attribute-Based Access Control: An access control method where subject requests to perform operations on objects are granted or denied based on assigned attributes of the subject, assigned attributes of the object, environmental conditions, and a set of policies that are specified in terms of those attributes and conditions.

Most researchers agree with the ABAC framework shown in Fig. 1 [15]. policy Enforce Point (PEP) performs preliminary verification of access requests and executes access control decisions; Policy Decision Point (PDP) evaluates access requests based on acquired attributes and policies; Policy Information Point (PIP) manages attributes of all subjects, objects, and environment; Policy Attribute Point (PAP) manages and maintains a policy library. In ABAC, when a subject accesses an object, it sends an access request

to an authorized organization. The process of evaluating access requests by the ABAC framework used by most researchers is shown in Fig. 1.

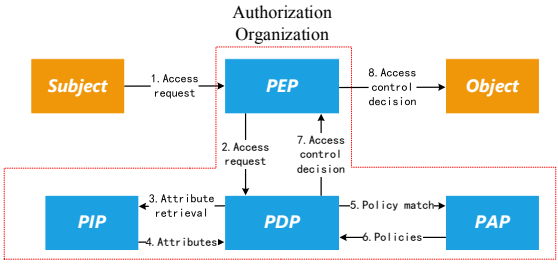

Fig. 1. ABAC framework

The access request used by the ABAC framework shown in Fig. 1 must contain the unique identifier of the subject and object. The basis for retrieving attributes is the unique identifier of the subject and object contained in the access request.

2.2 HABS Algorithm

HABS is an anonymous certification scheme based on the Attribute-Based Signatures (ABS). The ABS is designed for the user to sign a message with fine-grained control over identifying information, and it does not support the properties required for anonymous certification [8]. HABS has a clear identification of missing properties to serve anonymous certification objectives. HABS supports a flexible selective disclosure mechanism at no extra computation cost [13], which is inherited from the expressiveness of ABS for defining access policies.

HABS relies on four procedures based on the inspector, subject, issuer and verifier.

The *system initialization procedure* derives a global parameter containing the inspector's public key and pairs of public and private keys for the subject, inspector, and issuer. HABS.Setup and HABS.KeyGen is executed by the trusted third party.

The *credential issuing procedure* issues a certified credential for the subject based on its attributes. The HABS.Issue algorithm is executed by the issuer. The HABS.Obtain algorithm is executed by the subject.

The *verifying procedure* enables the verifier to check that a subject is authorized to access an object with respect to some access policy. As such, the verifier has first to send a random message to the subject. Second, the user signs the received message based on his credential. In a nutshell, the subject signs the received message based on the subset of his attributes that satisfy the signature predicate. The user finally sends his signature to the verifier who checks the resulting signature. The HABS.Show algorithm is executed by the subject. The HABS.Verify algorithm is executed by the verifier.

The HABS supports the *inspection procedure* performed by a separate and trusted entity referred to as the inspector. It relies on two algorithms namely HABS.trace and HABS.judge needed to identify the subject and give proof of judgment. They are executed by the inspector.

HABS supports a flexible selective disclosure mechanism that allows the subject to sign the message with a subset of its attributes. The verifier only obtains the subset of attributes after successful verification and does not reveal the identity of the subject. In addition, this scheme ensures the unlinkability between sessions while maintaining the anonymity of the subject.

3 A^{3}BAC Framework

A^{3}BAC is an access control framework with anonymous access and audit capabilities. A^{3}BAC does not use unique certificates and is friendly for all types of objects. It inherits the characteristics of fine-grained access control, flexible policy and unlimited object type of ABAC.

3.1 A^{3}BAC Framework

A^{3}BAC shown in Fig. 2 is a direct extension of the ABAC framework, which inherits many advantages of ABAC, such as fine-grained access control, flexible policy and unlimited object. The functions of PDP, PEP, and PAP in the A^{3}BAC framework are consistent with the corresponding modules in the framework shown in Fig. 1. The main functions of other modules are as Fig. 2.

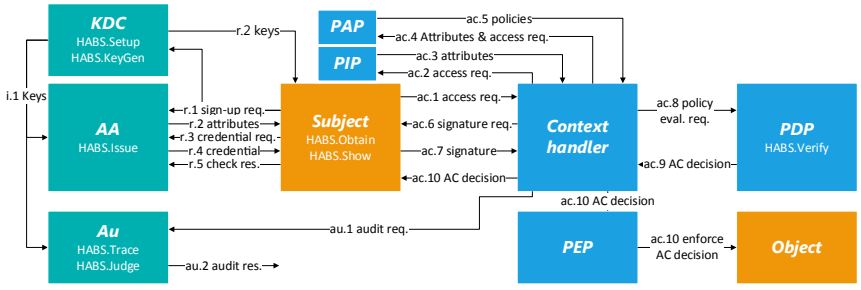

Fig. 2. A^{3}BAC framework; the green module is provided by a trusted third party, the blue module is provided by the authorization organization; the yellow module is the subject and object; req. means request; res. means result; eval. means evaluation. (Color figure online)

- PIP: is only responsible for managing object attributes and environmental attributes.
- Context handler: repackage and redirect requests based on context.
- Key Distribution Center (KDC): A trusted third party responsible for generating and distributing keys.
- Attribute Authority (AA): AA is responsible for managing all subject attributes. Subject obtains a certificate generated by AA.
- Audit Authority (Au): In A^{3}BAC, Au is responsible for auditing subject access.

3.2 Workflow of A^{3}BAC

The ABAC model has four workflows, which are initialization, registration, anonymous access and audit. Before the system starts running, the initialization workflow is executed:

(i.1) is HABS *system initialization procedure.*
After the system is initialized, the system is started. Registration, anonymous access, and audit are all performed at runtime. When the subject enters the system for the first time, it executes the registration workflow:
(r.1-5) is HABS *credential issuing procedure.*
When the system is running, the subject anonymously accesses the object:
(ac.1-5) is the same as the ABAC except that they do not contain subject attributes.
(ac.6) The context handler generates the signature predicate γ and the random message for the policies in the policy set, and packages them into signature requests and send signature requests to the subject one by one.
(ac.7) After receiving the signature request, the subject selects the appropriate subset of attributes and runs HABS.Show to sign the random message in the signature request and sends the signature to the Context handler.
(ac.8) After receiving the signature, the Context handler packages the signature with the corresponding policy, object attributes, and environment attributes into a policy evaluation request and sends it to the PDP.
(ac.9) The PDP evaluates the policies based on the attributes in the requests. Then, PDP combines the evaluation results of these policies to form an access control (AC) decision and return it to the Context handler.
(ac.10) The Context handler sends the access control decision to the subject. If the decision is deny, the workflow ends. If the decision is permit, it is forwarded to PEP and PEP enforces the access control decision.
When abnormal access of the subject is discovered, the audit workflow can be started:
(au.1) The context handler sends an audit request contains the signature and access request to Au.
(au.2) Au evaluates the audit request. If the request is denied, the audit decision will be output directly. If the audit decision is permit, the HABS.Trace and HABS.Judge algorithm is executed based on the information in the audit request to expose the corresponding subject identity. Then Au packages the subject identity and the evidence as to the audit decision and outputs it.

Obviously, the subject identity is not involved in A^{3}BAC's access workflow. The subject-related information obtained by the context handler or PDP in the access workflow is the signature. The signature only involves the predicate, the random message and a subset of subject attributes. Thus, A^{3}BAC's access workflow is identity independent, and A^{3}BAC's access is anonymous.

4 Analysis

Kaaniche N. and Laurent M. [7] proved the correctness, unforgeability, anonymity and the anonymity removal of the HABS algorithm. The security feature of A^{3}BAC is the

Table 1. Solution comparison

Literature	Anonymous access	Fine-grained AC	Policy flexibility	Audit	Restricted object type
[2]	Y	Y	N	N	N
[3]	Y	Y	N	N	Y
[4]	Y	Y	N	N	Y
[9]	Y	Y	N	Y	N
[10]	Y	N	N	Y	N
[11]	Y	Y	N	N	Y
A^3BAC	Y	Y	Y	Y	N

same as the HABS algorithm. And A^3BAC also has these properties. The comparison between A^3BAC and the existing solution is shown in Table 1.

The main problem of the existing solution is that it cannot balance anonymous access, auditing, restricted object type, fine-grained access control, and policy flexibility. Anonymous access, auditing involves security issues; restricted object type, fine-grained access control and policy flexibility involve availability. In other words, it is difficult for the access control framework to balance security and availability. A^3BAC has both these characteristics by combining HABS and ABAC.

5 Conclusion

Existing anonymous access solutions have the problems of subject re-identification and constraints on the types of objects. The ABAC with anonymous access proposed in this paper called A^3BAC inherits the features of the ABAC model, such as fine-grained authorization, policy flexibility, and unlimited object types. By combining HABS, it strengthens the identity-less of ABAC, so that the access does not involve a unique identification, reducing the risk of subject identity re-identification. The A^3BAC inherits HABS anonymity, unforgeability, unlinkability, anonymous removal, auditing and other features, ensuring the security of the model.

In future work, we will implement and evaluate A^3BAC. A^3BAC will be deployed and its performance will be tested. We will also evaluate the anonymity guarantee provided by A^3BAC to the subject.

References

1. Li, S., Choo, K.R., Sun, Q., et al.: IoT forensics: Amazon echo as a use case. IEEE Internet Things **6**(4), 6487–6497 (2019)
2. Yuen, T.H., Liu, J.K., Man, H.A., et al.: k-times attribute-based anonymous access control for cloud computing. IEEE Trans. Comput. **64**(9), 2595–2608 (2015)

3. Ahuja, R., Mohanty, S.K.: A scalable attribute-based access control scheme with flexible delegation cum sharing of access privileges for cloud storage. IEEE Trans. Cloud Comput. **PP**(99), 1 (2017)
4. Pussewalage, H.S.G., Oleshchuk, V.: A patient-centric attribute based access control scheme for secure sharing of personal health records using cloud computing. In: IEEE, International Conference on Collaboration and Internet Computing, pp. 46–53 (2017)
5. Sweeney, L.: k-anonymity. Int. J. Uncertainty Fuzziness Knowl.-Based Syst. **10**(05), 557–570 (2008)
6. Hu, V., Ferraiolo, D., Kuhn, R., et al.: Guide to attribute based access control (ABAC) definition and considerations. ITLB (2014)
7. Li, S., Da Xu, L., Zhao, S.: 5G internet of things: a survey. J. Ind. Inf. Integr. **10**, 1–9 (2018)
8. Maji, H.K., Prabhakaran, M., Rosulek, M.: Attribute-based signatures. In: International Conference on Topics in Cryptology: CT-RSA, pp. 376–392. Springer-Verlag (2011)
9. Backes, M., Camenisch, J., Sommer, D.: Anonymous yet accountable access control. In: [ACM WPES] Proceedings of ACM Workshop on Privacy in the Electronic Society, p. 40 (2005)
10. Yao, X., Liu, H., Ning, H., et al.: Anonymous credential-based access control scheme for clouds. IEEE Cloud Comput. **2**(4), 34–43 (2015)
11. Zhang, Y., Li, J., Chen, X., et al.: Anonymous attribute-based proxy re-encryption for access control in cloud computing. Secur. Commun. Netw. **9**(14), 2397–2411 (2016)
12. Li, S., Zhao, S., Yang, P., et al.: Distributed consensus algorithm for events detection in cyber-physical systems. IEEE Internet Things J. **6**(2), 2299–2308 (2019)
13. Kaaniche, N., Laurent, M.: Attribute-based signatures for supporting anonymous certification. In: Computer Security – ESORICS 2016 (2016). Li, S., Xu, L.D., Zhao, S., et al.
14. Servos, D., Osborn, S.L.: Current research and open problems in attribute-based access control. ACM Comput. Surv. **49**(4), 1–45 (2017)
15. Wang, X.M., Fu, H., Zhang, L.C.: Research progress on attribute-based access control. Acta Electronica Sinica **38**(7), 1660–1667 (2010)