Anonymity in Attribute-Based Access Control: Framework and Metric

Runnan Zhang, Gang Liu, Hongzhaoning Kang, Quan Wang, Bo Wan, and Nan Luo

***Abstract*—Anonymous access is an effective method for preserving privacy in access control. This study assumes that anonymous access control requires both frameworks and policies. Numerous solutions have been proposed for anonymous access at the framework level. In this study, these solutions are analyzed and quantified using a unified attribute-based access control (ABAC) anonymous access reference framework. Anonymous access at the framework level is the first line of defense, and inappropriate policies may undermine subject anonymity. An anonymity metric is proposed at the policy level to prevent authorization authority from re-identification using specific attributes and policies. The anonymity metric evaluates the risk of re-identifying a subject due to inappropriate access requests, as well as subject attribute assignment schemes and policies. This study is the first to focus on anonymity at the policy level in ABAC. Furthermore, a formal definition of anonymity suitable for ABAC is proposed. The feasibility of the proposed anonymity metric is verified through simulations.**

***Index Terms*—Anonymity metric, anonymous access, attribute-based access control, re-identification.**

I. Introduction

ANONYMITY, such as statistical disclosure control (SDC) [23], payment [9], and access control, is critical for protecting the privacy of subjects. In access control, anonymous access control (AAC) not only protects the subject's private information, such as the subject's name, birthday, and race, but also protects the subject's privacy behavior, which is the access request. Most AAC studies have focused on ensuring that access requests do not reveal the subject's identity. However, the re-identification risk resulting from the access control policy is not considered. In this approach, AAC and access control models are weakly coupled, which is advantageous. However, the disadvantage of this method is that the authorization authority (AA) may obtain subject-related information by formulating a special access control policy to re-identify the subject. Thus, AAC only avoids the re-identification risk at the framework level but not at the policy level.

Attribute-based access control (ABAC) models have garnered considerable attention and have been implemented in Amazon Web Services [1]. The difference between ABAC and the conventional access control model is that the authorization process of the former is attribute based rather than identity based. The attribute-based authorization process refers to an identity-less feature and attribute combination, which implies an identity-less authorization process [5]. Several studies have used this feature to achieve anonymous access, which include frameworks such as hierarchical attribute-set-based encryption (HASBE) [24], ciphertext-policy attribute-based encryption (CP-ABE) [7], and attribute-based signature (ABS) [15]. Anonymous access is achieved at the framework level. Therefore, the access request does not contain the subject's identity, and unified logic exists in these anonymous-access solutions. In this study, these solutions are analyzed and quantified using a unified ABAC anonymous access reference framework. This framework reveals the conditions that should be satisfied in ABAC to achieve anonymous access, in which the access request does not contain the identity of the subject.

In ABAC, anonymous access at the framework level allows the access request to conceal the identity of the subject, which is a prerequisite for anonymous access. However, the access request should contain information related to the subject such that the AA can make a reasonable access decision. In ABAC, the attributes of the subject contain the required information. The policy determines the attributes required by the AA when an access request is processed. If the AA develops a specific policy, then the subject can be easily re-identified using the method. For example, a subject is assigned an attribute with a unique value that should be included in the access request. This attribute replaces the identity, and the subject can be easily re-identified. Therefore, the AA should enforce policies that satisfy anonymity constraints, which is the second condition for anonymous access in ABAC. However, this policy is based on access control requirements, which may be in contrast to anonymous access. Unreasonable constraints on the policy result in the unavailability of access controls. This study proposes an anonymity metric that allows security administrators to evaluate

Manuscript received 8 September 2021; revised 25 February 2023; accepted 21 March 2023. Date of publication 24 March 2023; date of current version 12 January 2024. This work was supported in part by the Key Research and Development Program of Shannxi Province under Grants 2023-YBGY-205, 2021GY-086, in part by the Natural Science Basic Research Program of Shaanxi under Grant 2022JZ-40, and in part by the National Natural Science Foundation of China under Grant 61901362. *(Corresponding author: Gang Liu.)*

Runnan Zhang and Hongzhaoning Kang are with the School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi 710071, China (e-mail: zhangrunnan_xd@qq.com; kanghzn@stu.xidian.edu.cn).

Gang Liu is with the School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi 710071, China, also with the Guangzhou Institute of Technology, Xidian University, Xi'an, Shaanxi 710071, China, and also with the Key Laboratory of Smart Human-Computer Interaction and Wearable Technology of Shaanxi Province, Xi'an, Shaanxi 710071, China (e-mail: gliu_xd@163.com).

Quan Wang, Bo Wan, and Nan Luo are with the Key Laboratory of Smart Human-Computer Interaction and Wearable Technology of Shaanxi Province, Xi'an, Shaanxi 710071, China, and also with the School of Computer Science and Technology, Xidian University, Xi'an, Shaanxi 710071, China (e-mail: qwang_xd@163.com; wanbo@xidian.edu.cn; nluo@xidian.edu.cn).

Digital Object Identifier 10.1109/TDSC.2023.3261309

the anonymity and usability of policies such that policies can be formulated reasonably.

This paper discusses anonymous access at both the framework and policy levels. The contributions of this study are as follows:

1) A formal definition of anonymity suitable for ABAC is proposed.
2) At the framework level, anonymous access solutions for ABAC are analyzed and then quantified into a unified ABAC anonymous access reference framework. This framework specifies the features required for an anonymous-access framework.
3) An anonymity metric for the subject, access request, and policy is proposed at the policy level for the first time in ABAC.

The metrics of the various factors of an access control model and, in particular, the evaluations of policy-related factors, are critical. Because of the increase in the number of subjects and objects as well as the requirement for fine-grained access control, both the complexity of policies and the size of policy libraries are gradually increasing. Therefore, policy management has become more complex. Most studies focused on policy management through model improvement. However, the development of policy management tools is critical. Owing to the maturation of machine learning, semi-automated or automated policy management tools have been further developed. The k-nearest neighbor algorithm [6] has been used to facilitate security administrators in managing policy. Furthermore, mining algorithms [26] have been proposed for automatically generating policies. Any optimization or mining policy algorithm contains policy preference objective(s) and can be used as policy metrics. Administrators can select the appropriate policies. Metrics are crucial for ABAC because the same authorization can be expressed as different policies. Herein, a metric for anonymity in ABAC is proposed. To evaluate the various features of ABAC quantitatively, additional metrics should be introduced.

II. Related Work

A. Anonymous Access Control

AAC is an application-independent, privacy-preserving method. Many AAC methods achieve anonymous access using an "entity" that replaces the identity of the subject. The AA does not disclose the identity of the subject; thus, the subject's private information is protected. Yuen et al. [30] proposed an attribute-based anonymous-authentication method, where the subjects obtained unique credentials through authentication. This certificate does not contain the identity of the subject, and the cloud service is authorized based on credentials. Furthermore, the authors imposed additional constraints on the credential, which can only be used k times to improve privacy preservation. Using a credential instead of an identity in AAC is a simple and effective strategy for achieving anonymous access. However, the credentials used for AAC are unique. This technique allows an adversary to re-identify the credential owner based on information such as that related to access. This method is similar to background knowledge attack in SDC. The use of non-unique credentials instead of unique credentials has been advocated to maintain the anonymity of participants. Hampiholi et al. [11] proposed an anonymous transaction model to replace subject identity with attributes to complete transactions between the subject and database. However, in this study, attributes and access requests were not analyzed comprehensively. Notably, the improper use of attributes can result in re-identification.

AAC is achieved using unique credentials. AAC implemented by unique credentials is vulnerable to background knowledge attacks. Non-unique credentials for anonymous access are similar to k-anonymity. Sweeney et al. [23] proposed k-anonymity to publish a large dataset that renders the subject anonymous in SDC. K-anonymity requires at least k subjects to be assigned the same attribute for each combination of attributes. Furthermore, it imposes constraints on the data; however, the access control model cannot enforce the same constraints on the attributes. This study was inspired by k-anonymity, and a policy anonymity metric was proposed to generate policies with strong anonymity.

B. Evaluation and Metric of Access Control

The evaluation of the features of an access control model is a critical research topic. However, the feature quantitative evaluation of access models or policies are rarely reported. Servos et al. [20] classified existing studies pertaining to ABAC and evaluated them quantitatively by reviewing their results. Aftab et al. [2] analyzed, compared, and qualitatively evaluated ABAC and RBAC. In this study, a qualitative evaluation of 12 issues in ABAC and RBAC was conducted, and the results were evaluated via comparison with other studies. Ouaddah et al. [17] formulated a model to quantitatively evaluate access control, where the evaluation was performed based on six perspectives. The evaluation methods from the aforementioned studies include an analysis of existing studies, which are not suitable for applications because of the significant workload involved. Although qualitative evaluation is the crux of model evaluation, quantitative evaluation methods have been proven to be effective. Qi et al. [19] analyzed the advantages and disadvantages of RBAC, ABAC, and hybrid models and proposed a quantitative evaluation of decision efficiency, granularity, and flexibility. The evaluation process does not require human participation (reviewing the literature and refining documents) and can be applied in practice; however, formal definitions of decision efficiency, granularity, and flexibility were not provided. Quantitative evaluation methods for granularity and flexibility are applicable only to specific models. The quantitative evaluation method for decision efficiency is related to model implementation. The discrimination rate [22] is an attribute-centric metric based on information theory for privacy. This method is only suitable for runtime because the anonymity of the access requests is measured. Because the method does not involve the measurement of policy anonymity, policy creators cannot use it to formulate appropriate policies.

Policy mining and optimization involve a quantitative evaluation of policy features. Owing to the revival of artificial intelligence research, researchers have focused on automated policy generation and optimization. A common feature of these algorithms is their objective to determine the direction of policy

generation or optimization. The objectives of these algorithms are the metrics of the policy characteristics. For example, Xu and Stoller proposed a policy-mining algorithm that generates policies to minimize the weighted structural complexity (WSC) for mining the ABAC policy from the RBAC policy [25], logs [27], and ABAC policy [26]. The original WSC is a metric pertaining to the number of relationships between elements in the RBAC model [16]. Bui et al. [8] proposed a mining algorithm for the relationship-based access control policy, which similarly targets WSC minimization. In addition to WSC, other policy features, such as algorithm objectives, have been proposed. John et al. [14] proposed an algorithm for mining an ABAC policy in a cooperative cloud environment. The objective of the algorithm is to maximize the number of permissions included in each policy. Hadj et al. [10] achieved ABAC policy redundancy using the k-nearest neighbor algorithm. In their study, the similarity of attributes between policies was defined as the distance between them, and the objective of their algorithm was to eliminate similar policies. However, policies with similar attributes may express different permissions, which implies that the rules are neither "similar" nor "redundan" for permissions. Zhou et al. [33] proposed an algorithm for predicting access-control decisions based on partial policy sets, and the objective of the algorithm was to improve prediction accuracy. Zhang et al. [32] classified policies based on the attribute distance in an extensible access control markup language to improve evaluation efficiency. However, the objectives of current policy-mining or optimization algorithms do not involve anonymity.

The quantitative evaluation of access control features can be classified into the following two categories: 1) Measures the features of the access control model. Measurement methods are generally devised by reviewing the literature to obtain evaluation results, which are expensive and subjective. 2) Measures of features of access control policies. These measures are policy mining and policy optimization goals. However, these policies cannot formalize the definition of the features, thus rendering them unreliable. Herein, a formal definition of anonymity and a metric based on a formal definition are introduced. The anonymity metric can be used as an objective in policy mining or optimization algorithms, thus allowing security administrators to further understand and improve their policies.

III. Preliminaries

Although no consensus has been reached regarding ABAC standardization, the accepted high-level definitions and descriptions of its function are available. A high-level description is presented in the "Guide to Attribute Based Access Control (ABAC) Definition and Considerations" published by the National Institute of Standards and Technology [13].

ABAC: An access control method in which a subject requests to perform operations on objects is granted or denied based on the assigned attributes of the subject, assigned attributes of the object, environmental conditions, and a set of policies specified in terms of those attributes and conditions.

Yuan et al. [29] proposed an ABAC framework (Fig. 1). Most studies regarding ABAC are based on this framework. In this framework, authorization proceeds as follows:

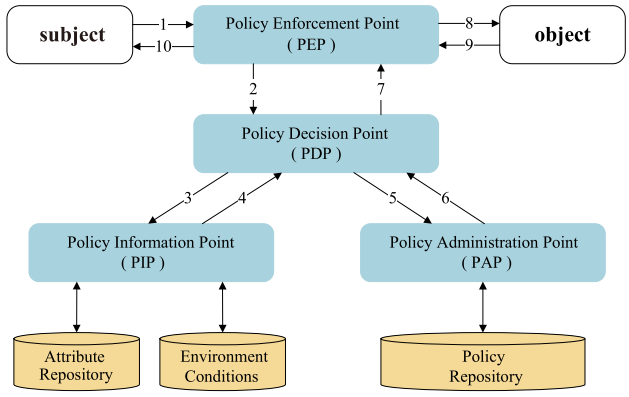


Fig. 1. ABAC framework.

1) The subject sends a request to the PEP; the request contains an identifier of the subject (such as an ID), an identifier of the object, the operation, and the context of the subject (such as location);
2) The PEP forwards the request to the PDP;
3) The PDP forwards the request to the PIP;
4) The PIP returns the corresponding attributes;
5) The PDP queries the appropriate policies based on attributes related to the request;
6) The PAP returns the policies;
7) The PDP performs an access control decision based on the policies and attributes and sends it to the PEP;
8) The PEP enforces access control decision;
9) Access results are returned to the subject.

In the ABAC framework shown in Fig. 1, the access request arrives at the PIP through the PEP and PDP, and the PIP searches for subject-related attributes based on the subject identifier contained in the request. The AA (including the PEP, PDP, PAP, and PIP in ABAC, which are marked in blue) makes access control decisions based on attributes associated with the access request. The AA displayed in Fig. 1 can directly or indirectly identify subjects using attribute combinations and implement identity-less ABAC. Therefore, identity-less ABAC cannot ensure anonymity. To achieve anonymity, identity-less ABAC should be constrained, which renders it difficult for the AA to identify the subjects.

IV. Definition of Anonymity and Identity-Free ABAC

In the conventional access control model, the AA stores information regarding numerous subject privacy behaviors owing to its responsibilities. Moreover, whether the AA is for commercial purposes, curiosity, or auditing, the privacy behavior of the subject can be obtained. We assumed an attack scenario in which the AA re-identified the subject who sent the request through the available information and conducted targeted attacks on various subjects. Because the AA has access to information regarding the subject's attribute assignment scheme, request, and policy in the access control process, it can re-identify the subject who sent the request. Thus, the subject may be attacked.

A. Formal Definition of Anonymity

In ABAC, subject identity is typically considered an attribute (such as an ID) and each subject identity attribute exhibits a unique value that identifies its principal identity. However, the subject may have many attributes with unique ABAC values. Although these attributes may not be an identification, they have the same features as the ID. The AA may re-identify the subjects by regarding attribute(s) with a unique value(s) as an ID. Therefore, a definition for the identifier was proposed to prevent the re-identification of subjects who are more suitable for ABAC.

Definition IV.1. Subject identifier. In ABAC, when attributes with particular values are assigned to only one subject, they are known as subject identifiers.

According to Definition 4.1, a subject may have more than one identifier. Identifiers can be classified into two categories. The first is an explicit identifier. If a bias exists from the universal set of subjects to the range of an attribute, the attribute is known as an explicit identifier. Explicit identifiers (such as IDs) are typically used by administrators to identify one or more subjects in a policy. If the identifier is not explicit, then it is implicit. Implicit and explicit identifiers differ primarily in terms of two aspects. First, implicit identifiers may consist of one or more attributes, whereas explicit identifiers consist of only one attribute. Second, bijection from the universal set of subjects to the implicit identifier does not occur. Thus, some of the attribute values used by implicit identifiers may be possessed by more than one subject.

To define anonymity in ABAC formally, the following symbols are used.

S is the universal set of subjects, s is the subject, and $s \in S$; m attributes are denoted as $attr_i$. Each attribute $attr_i$ corresponds to domain $Domain_i$, which is a range of $attr_i$. The value $\perp$ signifies that the subject is not assigned an attribute. In real applications, the symbol $attr_i$ is not used as an attribute name; instead, a meaningful name is used. For example, consider three attributes, i.e., age, gender, and birthday. Alice's attributes may be $age = 20, sex = female, birthday = \perp$. Here, $birthday = \perp$ indicates that Alice is not assigned the birthday attribute. Anonymity is not only related to the subject, but also to access requests and policies. Hereinafter, if not specified, the term attribute is equivalent to the subject attribute.

The attribute assignment of subject s is denoted as a vector $\overrightarrow{attr_s} = (value_1, value_2, value_3, \ldots, value_m)$, where $value_i \in Domain_i$.

Definition IV.2. Construction of attributes vector. The conversion of zero or more $none - \perp$ values to $\perp$ in the attribute vector $\overrightarrow{attr}$ generates a new attribute vector $\overrightarrow{attr'}$, known as the construction of attributes vector, denoted as $\overrightarrow{attr'} \sqsubseteq \overrightarrow{attr}$.

All access requests generated by each subject are constructed from its attribute vectors. For example, Alice can perform an access request as follows:

$$\overrightarrow{req} = (\perp, female, \perp) \sqsubseteq \overrightarrow{attr_{Alice}} = (20, female, \perp)$$

In ABAC, anonymity is exprressed as follows:

Definition IV.3. The subject space of $\overrightarrow{attr}$ is expressed as

$$S_{\overrightarrow{attr}} = \{s | \overrightarrow{attr} \sqsubseteq \overrightarrow{attr_s}, s \in S\} \quad (1)$$

1) $|S_{\overrightarrow{attr}}| = 0$, no subject can construct such an attribute vector, and it is not related to anonymity;
2) $|S_{\overrightarrow{attr}}| = 1, s \in S_{\overrightarrow{attr}}$ loses anonymity for attribute vector $\overrightarrow{attr}$, $\overrightarrow{attr}$ is an identifier of the subject;
3) $|S_{\overrightarrow{attr}}| > 1$, subjects in subject set $S_{\overrightarrow{attr}}$ maintains anonymity for the attribute vector $\overrightarrow{attr}$.

Definition 4.3 is based on the study conducted by Pfitzman et al. [18], who defined "the anonymity of a subject from an attacker's perspective means that the attacker cannot sufficiently identify the subject within the set of all possible subjects." Obtaining an identifier is sufficient to identify a subject. Loss of subject anonymity (re-identification) occurs when any entity (except the subject) acquires the identifier of the subject. Otherwise, the participants remain anonymous.

When $|S_{\overrightarrow{attr}}| > k, k > 1$, k-anonymity is achieved. Anonymity differs between access control and SDC. Two problems exist when enforcing k-anonymity for all access requests. 1) Policy problem. If an object only allows access by $k-1$ subjects, then $\exists \overrightarrow{attr}, |S_{\overrightarrow{attr}}| = k - 1$. Otherwise, at least one policy does not conform to the principle of least privilege. 2) Runtime problem. In SDC, k-anonymity can be satisfied by accumulating data, because it publishes data within a certain period. However, we cannot accumulate access requests in access control and send them to the AA to satisfy k-anonymity, because the access requests are processed in real time. Enforcing k-anonymity on access control is difficult because its side effects cause the policies to violate the least privilege set and cause access request processing to lose its real-time nature. Therefore, a quantitative evaluation method is proposed to achieve anonymity that allows security administrators to recognize the anonymity of a policy regardless of whether it follows k-anonymity. This phenomenon enhances the security administrators' ability to improve policy anonymity.

According to Definition 4.3, to ensure the anonymity of the subjects in ABAC, access requests must not be able to identify a subject. This can be achieved using two techniques. First, the anonymous access framework allows subjects to generate access control requests, which renders the identity of the subject in access control requests no longer mandatory. In another method, the anonymity metric of the policy, which allows the subject or administrator to evaluate the risk of the subject being re-identified, is defined. Depending on the policy availability, the policy might not guarantee anonymity. For example, if a request exists, only a specific user can construct a request. According to Definition 4.3, this request can be re-identified. However, for most policies, particularly those that can be organized into various forms, the anonymity metric can facilitate the adjustment of policies via administrators or algorithms. Anonymity metrics can facilitate the generation of strong anonymous policy sets via administrators, policy generation algorithms, and policy optimization algorithms. The framework and policies of access control should support anonymity such that access can be

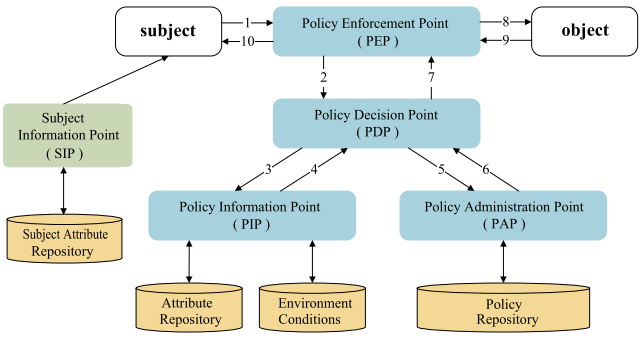


Fig. 2. Identity-free ABAC reference framework.

anonymous. Otherwise, the anonymity of the policy is strong; however, the access request should include the identity of the subject because of the limitations of the framework. This process renders the access no anonymous.

B. Identity-Free ABAC

By analyzing the characteristics of multiple existing ABAC frameworks that support anonymous access, such as HASBE [24], CP-ABE [7], and ABS [15], an ABAC anonymous access reference framework, namely, identity-free ABAC, was proposed. This framework highlights the features of the ABAC framework, which supports anonymous access. If an ABAC framework exhibits the same characteristics as those of this framework, the it is identity-free and supports anonymous access.

Three features of ABAC support anonymous access are as follows:

1) AA (marked in blue) does not manage subject attributes;
2) The subjects can construct access requests by themselves, and the request cannot include an explicit identifier;
3) The AA can make access control decisions based on access requests that do not contain an explicit identifier.

If an ABAC framework satisfies these three features, then the framework supports anonymous access. The crux of an ABAC framework that supports anonymous access is that it allows the AA to make access control decisions based on access requests that do not contain explicit identifiers. This affects not only the access control decision-making process, but also the form of access requests and the management of subject attributes. Furthermore, by removing the AA's ability to manage subject attributes, the relationship between subject attributes and subjects can be obtained directly. This phenomenon eliminates the link between the access request and subject. Access requests cannot contain explicit identifiers because Constraint 3. The access request is constructed by the subject, not the subject attribute management agency (such as the SIP in Fig. 2), which prevents the subject identity from being associated with the subject attributes or access request.

The models shown in Figs. 2 and 1 differ in terms of two aspects at the framework level. First, the access request without an explicit identifier is used in Steps 1, 2, and 3. The other steps shown in Fig. 2 are consistent with those shown in Fig. 1. Second, the SIP, instead of the PIP, manages the subject attributes. The PAP obtains the name and domain of the subject attribute from the SIP to ensure the validity of the policy. The PIP manages the environment and object attributes. Access requests contain only attribute-related information that enables the PDP to make access control decisions and do not contain the explicit identifier of the subject. The reference framework shown in Fig. 2 satisfies the three constraints of identity-free ABAC. The AA may not re-identify the subject using an explicit identifier.

V. Access Anonymity Metric

If the subject attribute management agency and AA collude to re-identify a subject, then an identity-free ABAC cannot ensure access anonymity. An adversary can calculate a set of subjects based on the attributes of an access request. The adversary may directly or indirectly re-identify the subjects that send an access request by analyzing the information related to the access request [4]. Thus, the subject can be re-identified. The access anonymity metric (AAM) quantitatively evaluates the re-identification risk. In this section, the AAM of the subject attribute assignment scheme, access request, and policy is presented.

A. AAM of Access Request

The AAM of the access request measures the risk of the subject being re-identified when it sends an access request. The basis of the anonymity metric for an access request is the definition of anonymity. According to Definition 4.3, after receiving access request $\overrightarrow{req}$, the adversary calculates a set of subjects $S_{\overrightarrow{req}}$ that can construct access request $\overrightarrow{req}$. However, the adversary cannot reveal the subject that sends access request $\overrightarrow{req}$. The adversary may assume that subject s in the subject set $S_{\overrightarrow{req}}$ sent the access request $\overrightarrow{req}$. If s sends access request $\overrightarrow{req}$, then s is re-identified; otherwise, the subject that sends access request $\overrightarrow{req}$ remains anonymous.

Random variable X represents the subject estimated by the adversary. Furthermore, $0 < P(X = s) \leq 1$, $\sum_{s \in S_{\overrightarrow{req}}} P(X = s) = 1, s \in S_{\overrightarrow{req}}$. If identifying the subject sending the request based on the adversary's guesses is difficult, then it implies that the anonymity of the subject is stronger, and vice versa. Notably, the quantitative evaluation of the anonymity of the access request is the uncertainty of random variable X.

$$\mathcal{A}_{req}(\overrightarrow{req}) = -\sum_{s \in S_{\overrightarrow{req}}} P(X = s) \log P(X = s), \qquad (2)$$

where $\mathcal{A}_{req}(.)$ is the AAM function of the access request. When $|S_{\overrightarrow{req}}| = 1$, $\mathcal{A}_{req}(\overrightarrow{req}) = 0$ the subject is re-identified. The anonymity of the access request increases with $\mathcal{A}_{req}(.)$.

The AAM of the access request assumes the same form as the information entropy because the information entropy of random variable X is a metric of the uncertainty of X[21]. Thus, the less information an adversary obtains (increased uncertainty), the more difficult it is to identify the subject that sends the access request. The "information" obtained by the adversary is the set of subjects $S_{\overrightarrow{req}}$ who may send the access request. The difficulty of re-identification increases with the amount of information entropy. The quantitative evaluation of the anonymity function

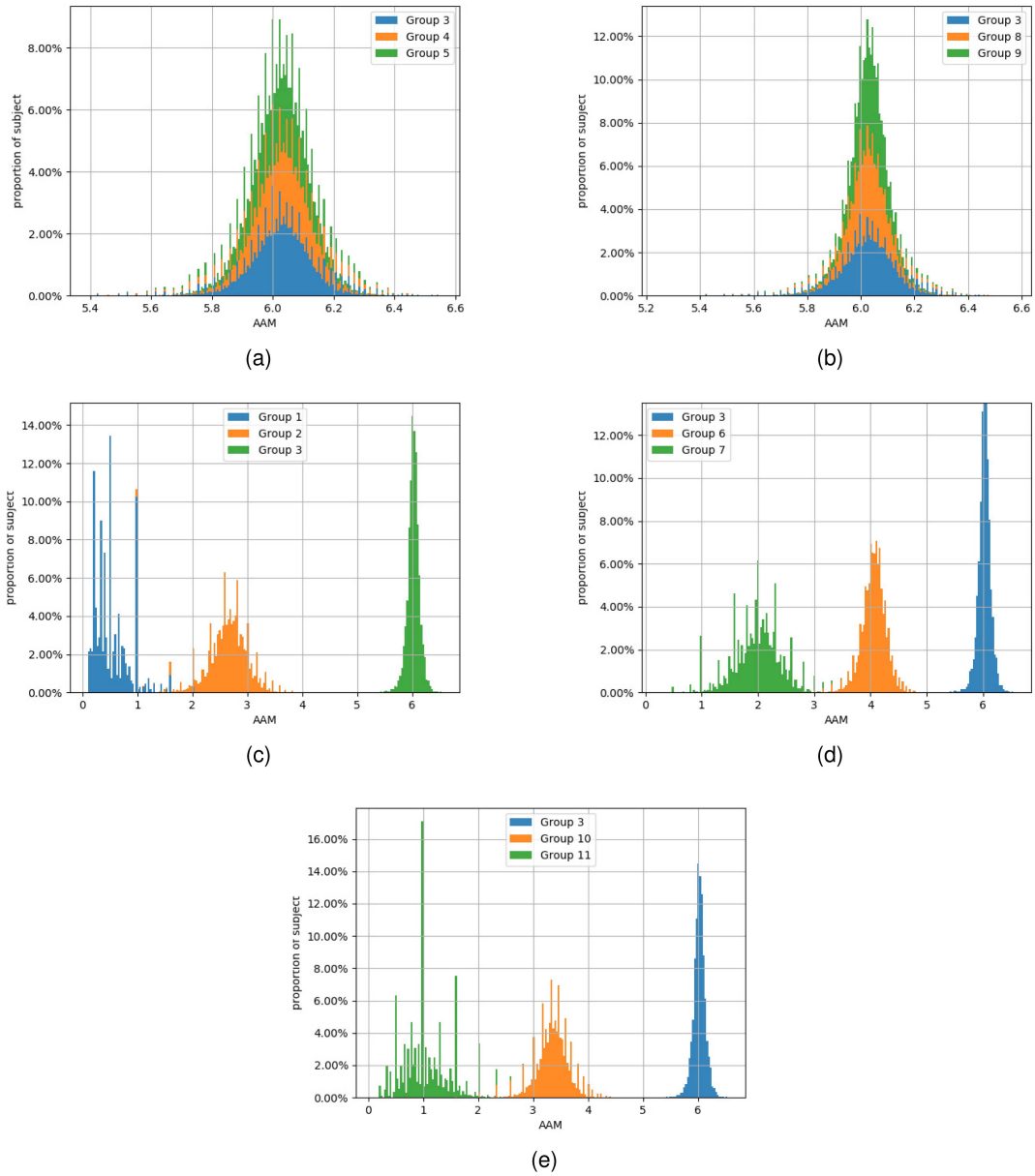


Fig. 3. AAM distribution of subject assignment under the effects of five corresponding iterms such as (**a**) number of attributes,(**b**) number of policies, (**c**) number of subjects, (**d**) number of values in attributes, and (**e**) number of values in policies.

of the access control request has the same form as that of information entropy.

B. AAM of Subject Assignment and Policy

Both subject attribute assignments and policies affect the anonymity of subjects. The request constructed by the subject is limited by the subject attribute assignment and policies.

1) AAM of Subject Assignment: The AAM of the subject assignment measures the risk that the access requests constructed by the subject attribute vector $\overrightarrow{attr_s}$ re-identify the subjects. The AAM of access requests is the basis for the AAM of the subject assignment scheme.

Definition V.1. Subject accesses request construction space. A set that contains all access requests $\overrightarrow{req}$ constructed from the subject attribute vector $\overrightarrow{attr_s}$ is known as the subject access request construction space Req_s.

$$Req_s = \{\overrightarrow{req} | \overrightarrow{req} \sqsubseteq \overrightarrow{attr_s}\} - \overrightarrow{\emptyset} \tag{3}$$

where $\overrightarrow{\emptyset} = (\perp, \perp, \perp, \ldots, \perp)$, is an m-dimensional vector.

$$\mathcal{A}_s(subject) = \sum_{i=1}^{|Req_s|} \omega_i \mathcal{A}_{req}(\overrightarrow{req_i}), \tag{4}$$

where $\overrightarrow{req_i} \in Req_s, \sum_{i=1}^{|Req_s|} \omega_i = 1$. $\mathcal{A}_s(.)$ increases with the anonymity of the subject. Introducing weights enhances the flexibility of the AAM used in the subject assignment scheme, thus allowing it to consider factors other than quantitative evaluation of anonymity (such as access frequency). Weights increase the

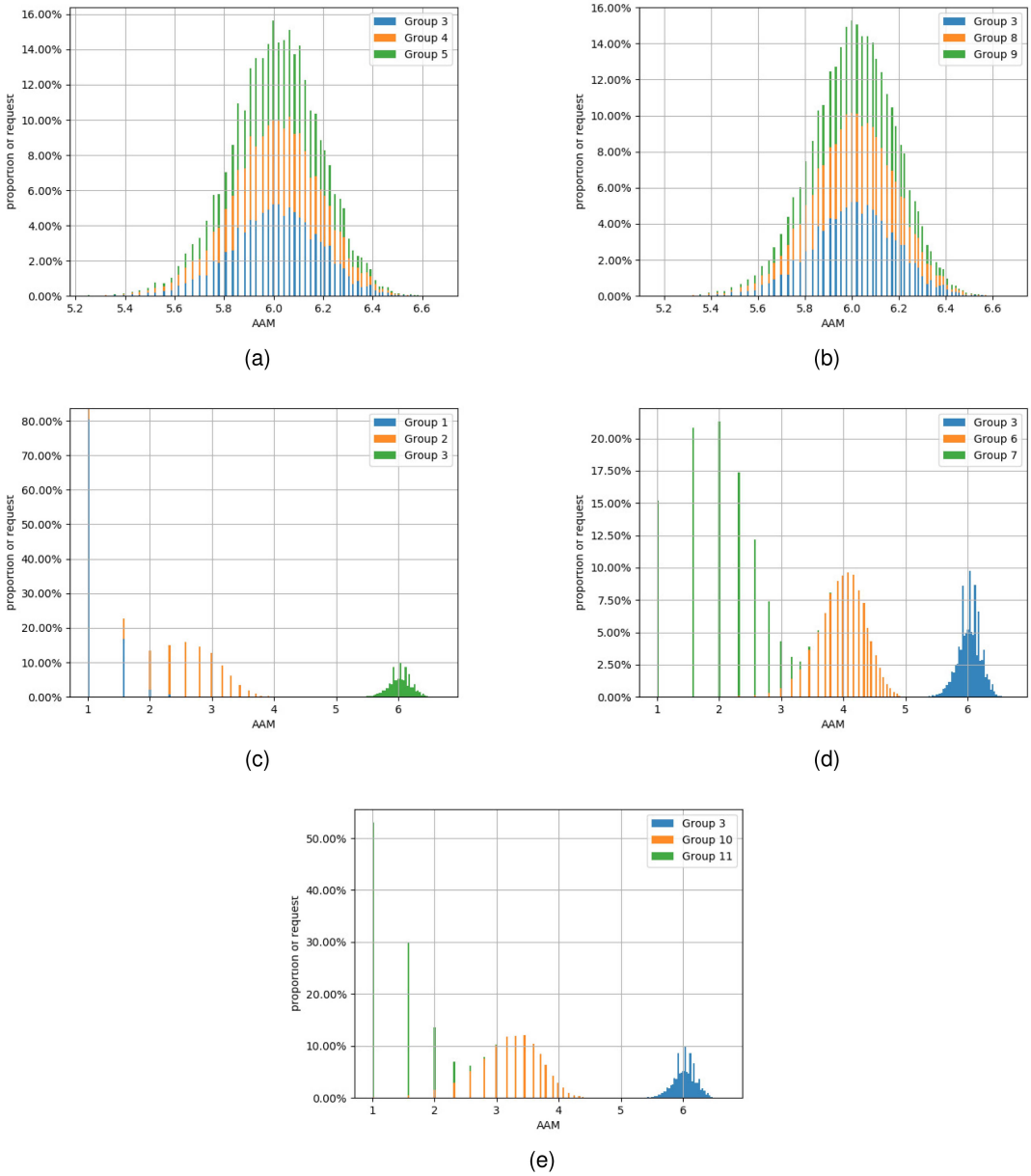


Fig. 4. AAM distribution of access request under the effects of five corresponding iterms such as (**a**) number of attributes,(**b**) number of policies, (**c**) number of subjects, (**d**) number of values in attributes, and (**e**) number of values in policies.

accuracy of the AAM used in the subject assignment scheme when prior knowledge of access is available. For example, if the policy is known, the the weight of the access request without the corresponding policy is assigned a value of zero to avoid disturbing quantitative evaluation.

In ABAC, administrators have considerable flexibility in selecting attributes when formulating attribute assignment schemes. The AAM of the subject attribute assignment scheme can help administrators select better attributes and improve the anonymity of the subject attribute assignment scheme.

2) AAM of Policy: The AAM of the policy measures the risk of the re-identification of subjects that may trigger the policy. The basis of the policy's AAM is the access request related to the policy. However, policy privacy considerably affects the access requests constructed by the subject. In this study, policy privacy is categorized into the following three types based on the information that the policy exposes to the subject:

1) The policy is private. The subjects have no policy-related knowledge. Therefore, the policy and access requests received by the AA are irrelevant. The AA receives random-access requests.
2) The policy is protected. The subjects understand the attributes involved in the policy but do not understand the corresponding conditions. For example, if the policy is (age $>$ 18 years, sex $=$ male,...), then the policy shows (age, sex,...) to the subjects. Access requests may contain a set of attributes that satisfy the policy.

3) The policy is public. The subjects understand the complete policy and can evaluate the policy by themselves. We believe that access requests contain a set of attributes that conforms to the corresponding constraints.

Currently, few studies have been conducted regarding policy privacy, particularly those regarding the privacy preservation of policies. Harbach et al. [12] proposed a privacy policy by introducing trusted third parties as the PDP. The policy protects the privacy of both the subjects and the AA and requires that a third party be trusted by both the subjects and AA. In anonymous-access ABAC, the privacy of the policy directly affects the construction of access requests.

Public and protected policies are identical in terms of the AAM. Access requests and policies are relevant to both public and protected policies, and the same quantitative evaluation of anonymity can be used. However, if the policy is private, then access requests and policies are irrelevant. No effective policy anonymity evaluation method has been devised for privacy policies.

Herein, the policy is represented as a disjunctive normal form and the rule is represented as a fundamental product:

$$\begin{aligned} policy &= rule_1 \vee rule_2 \vee rule_3 \vee \ldots \vee rule_p, \\ rule &= clause_1 \wedge clause_2 \wedge clause_3 \wedge \ldots \wedge clause_m, \\ clause_i &= attr_i \subseteq ValueSet_i, \end{aligned} \tag{5}$$

where $ValueSet_i$ is the constraint of $attr_i$, and $ValueSet_i \subseteq Domain_i$.

If $attr_i$ is unconstrained, then $ValueSet_i = Domain_i$; if $attr_j$ is constrained, then $ValueSet_j \subset Domain_j$. For protected policies, the subjects are aware that $attr_i$ does not assume the value $\perp$, i.e., $\perp \notin ValueSet_i$and$ValueSet_i \subset Domain_i$. From the perspective of the subject, all attributes in the privacy policy correspond to domains, i.e., $ValueSet_i = Domain_i$. For public policy, subjects are aware of the domains of all attributes. For all policy privacy types, the rule specifies a space in which an access request vector can be applied.

Definition V.2. Rule space. The rule space Req_r is a set of access request vectors that contain all access request vectors that apply to the rule from the perspective of the subject.

$$Req_r = \prod_{i=1}^{m} ValueSet_i - \overrightarrow{\emptyset} \tag{6}$$

The AAM of the policy can be classified into two types. The first is the AAM of the rule, and the second is the AAM of the policy based on the rule.

$$\mathcal{A}_r(rule) = \sum_{i=1}^{|Req_r|} \omega_i \mathcal{A}_{req}(\overrightarrow{req_i}) \tag{7}$$

where $\overrightarrow{req_i} \in Req_r, \sum_{i=1}^{|Req_r|} \omega_i = 1$. $\mathcal{A}_r(rule)$ increases with its anonymity. In $\mathcal{A}_r$, the weights are rule-related information, such as the frequency of access requests related to the rule. In particular, security administrators can adjust the weights based on actual scenario changes and ensure the anonymity of the same rule in various scenarios. Thus, the weights provide superior scenario adaptability for rule anonymity evaluation and yield accurate evaluation results.

Access requests related to a policy are a union of the rule spaces Req_r which is included in the policy. Therefore, the AAM should consider all rules included in the policy.

$$\mathcal{A}_p(policy) = \frac{1}{p} \sum_{i=1}^{p} \mathcal{A}_r(rule_i) \tag{8}$$

Any policy expressed by the disjunctive normal form weakens the connection between the rules. When one rule changes, the anonymity of the other rules does not change. Equation (8) is based on the following properties.

Anonymity evaluation of access requests, subject attributes, and policies reveals that the adversary obtains the set of subjects associated with them through access requests, subject attributes, or policies. Regardless of whether the access request, subject attribute, or policy is attribute based, an anonymity evaluation can include access control evaluations based on a set of subjects. For the current anonymous access schemes, such as CP-ABE [31], modified ABAC model [3], ABS [30], and anonymous certificate [28], determining the set of subjects related to access requests, subject attributes, and policies using certain analysis methods is critical. Anonymity evaluation can be performed using a generalized anonymity evaluation method.

C. Samples

To visualize the access anonymity evaluation metrics proposed herein, a specific scenario was considered to analyze the anonymity of requests, subjects, and policies using the proposed access anonymity evaluation method.

Users, tenants, and ABAC were used in an existing movie cloud with authentication and access control capabilities. In this case, the tenant purchases movies stored in the cloud and sells them to the user while providing them with several corresponding attribute credentials. Each attribute's credentials are independent. The tenant registers access control policies to the authentication and access control module provided by the cloud to achieve user-access control. The user accesses the movies stored in the cloud by selecting one or more of their attribute credentials in the access request. The movie cloud receives the access request sent by the user and performs an access control decision based on the subject attributes, objects, and operations. The access control process on the cloud in this scenario satisfies the three characteristics of identity-free ABAC. Therefore, its identity is free and the user can access it anonymously.

We assume two tenants, three users (subjects) and three movies (objects). Tenant 1 provides a buyout service, in which users can watch unlimited movies once they purchase a category. Tenant 2 provides a subscription service, in which users are provided with movie viewing services through various levels of VIPs. The attributes of the subjects and objects are listed in Tables I and II, respectively.

The process of access control in this scenario is as follows:

1) Determine whether the request contains attribute credentials for multiple tenants. If this is the case, then reject the request. This process protects the rights of each tenant

TABLE I
SUBJECT ATTRIBUTES

Subject user name	$\boldsymbol{Attr_1}$ category 1 credential	$\boldsymbol{Attr_2}$ category 2 credential	$\boldsymbol{Attr_3}$ category 3 credential	$\boldsymbol{Attr_4}$ VIP level credentials
Alice	Y	$\perp$	Y	1
Bob	Y	Y	$\perp$	1,2
Candy	$\perp$	Y	Y	1,2,3

TABLE II
OBJECT ATTRIBUTES

Object movie	$\boldsymbol{Attr_1}$ category	$\boldsymbol{Attr_2}$ VIP level
1	1,2	1
2	2,3	2
3	1,3	3

and ensures that the user can only use the corresponding attribute credentials in the service of a specific tenant.

2) Determine whether the credentials in the request expire, and reject the credentials if they are. This process verifies the validity of the attribute credentials obtained by the user from the tenant and ensures that the user uses the attribute credentials within the validity period.
3) Match policies based on attributes and make access control decisions.

1) AAM of Access Request: To ensure the anonymity of requests in this scenario, two sample access requests were provided, and their AAMs were computed. The requests shown below do not provide an object or an operation. However, the calculation of the AAM is not affected.

If subject Alice constructs a request $\overrightarrow{req_1} = (Y, \perp, Y, \perp)$, the the request contains attribute credentials $Subject.Attr_1 = Y$, $Subject.Attr_2 = \perp$, $Subject.Attr_3 = Y$, and $Subject.Attr_4 = \perp$. Based on Table I, only Alice can construct the request; therefore, $S_{\overrightarrow{req_1}} = \{Alice\}$. Thus, the following expression is obtained:

$$\begin{aligned}\mathcal{A}_{req}(\overrightarrow{req_1}) &= -P(X = Alice)logP(X = Alice)\\ &= -1 \cdot log1 = 0\end{aligned}$$

The anonymity of request $\overrightarrow{req_1}$ is 0. The initiator of this request can be directly identified as user Alice, i.e., request $\overrightarrow{req_1}$ is not anonymous.

If subject Bob constructs a request, then $\overrightarrow{req_2} = (\perp, Y, \perp, \perp)$. Based on Table I, the request can be constructed from a user with attribute credentials $Subject.Attr_2 = Y$, such that $S_{\overrightarrow{req_2}} = \{Bob, Candy\}$. If the attacker does not know the information related to the request and the user, then the probability of each user performing the request is equal, i.e., $P(X = Bob) = P(X = Candy) = 0.5$. Thus,

$$\begin{aligned}\mathcal{A}_{req}(\overrightarrow{req_2}) = &-P(X = Bob)logP(X = Bob)\\ &-P(X = Candy)logP(X = Candy)\\ = &-0.5 \cdot log0.5 - 0.5 \cdot log0.5 \approx 0.301\end{aligned}$$

If the opponent learns in advance that the frequency of Bob and Candy watching Category 2 movies is $3:7$, i.e., $P(X = Bob) = 0.3, P(X = Candy) = 0.7$, then,

$$\mathcal{A}'_{req}(\overrightarrow{req_2}) = -0.3 \cdot log0.3 - 0.7 \cdot log0.7 \approx 0.265$$

Here, $\mathcal{A}'_{req}(\overrightarrow{req_2}) < \mathcal{A}_{req}(\overrightarrow{req_2})$ when the attacker learns the relationship between the request and the user, i.e., the anonymity of request $\overrightarrow{req_2}$ decreases. Thus, the request anonymity evaluation can be used to visualize the anonymity level of the request and the change in request anonymity due to the change in information.

2) AAM of Subject Assignment: Prior to calculating subject anonymity, the following constraint exists for access control requests in this scenario: the requests are not allowed to contain the attribute credentials of various tenants; that is, the requests can only contain the attribute credentials issued by one category of tenants.

Considering user Alice as an example, $\overrightarrow{attr_{Alice}} = (Y, \perp, Y, 1)$. Based on Definition 4.2, Alice can construct the access request as follows:

$$\begin{aligned}\overrightarrow{attr_{Alice}} = \{&(Y, \perp, Y, 1), (\perp, \perp, Y, 1), (Y, \perp, \perp, 1), (Y, \perp, Y, \perp),\\ &(\perp, \perp, \perp, 1), (\perp, \perp, Y, \perp), (Y, \perp, \perp, \perp)\}.\end{aligned}$$

According to previous constraints, requests $(Y, \perp, Y, 1)$, $(\perp, \perp, Y, 1)$, and $(Y, \perp, \perp, 1)$ are not permitted. To protect their own anonymity to the maximum extent, the users will not use $(Y, \perp, Y, \perp)$ as a request based on the aforementioned evaluation results. Thus, the following expression is obtained:

$$\begin{aligned}\omega(Y, \perp, Y, 1) = \omega(\perp, \perp, Y, 1) = \omega(Y, \perp, \perp, 1)\\ = \omega(Y, \perp, Y, \perp) = 0.\end{aligned}$$

Assume that Alice uses Tenant 1 and 2 services with a frequency of $6:4$, and Category 1 and Category 2 movies are viewed with the same frequency, i.e., $(Y, \perp, \perp, \perp) : (\perp, \perp, Y, \perp) : (\perp, \perp, \perp, 1) = 3:3:4$. If the security administrator sets the weight of access requests based on the frequency ratio, then the following expression can be obtained:

$$\begin{aligned}\omega(Y, \perp, \perp, \perp) = \omega(\perp, \perp, Y, \perp) = 0.3\\ \omega(\perp, \perp, \perp, 1) = 0.4\end{aligned}$$

Based on (4), the subject anonymity of Alice can be expressed as follows:

$$\begin{aligned}\mathcal{A}_s(Alice) = &\omega(Y, \perp, \perp, \perp)\mathcal{A}_{req}(Y, \perp, \perp, \perp)\\ &+ \omega(\perp, \perp, Y, \perp)\mathcal{A}_{req}(\perp, \perp, Y, \perp)\\ &+ \omega(\perp, \perp, \perp, 1)\mathcal{A}_{req}(\perp, \perp, \perp, 1)\\ \approx &0.3 \cdot 0.301 + 0.3 \cdot 0.301 + 0.4 \cdot 0.477\\ \approx &0.371\end{aligned}$$

3) AAM of Policy: In this scenario, various tenants have distinct policies depending on their service models to ensure the approperiate service delivery. Considering Tenant 2 as an example, its policy is that the user can watch all movies at its

VIP level (including the same level). The policy is expressed based on the table of subject and guest attributes as follows:

$$policy = \begin{cases} (s.attr4 \subseteq \{3\} \wedge o.attr2 \leq 3 \wedge op = r) \vee \\ (s.attr4 \subseteq \{2,3\} \wedge o.attr2 \leq 2 \wedge op = r) \vee \\ (s.attr4 \subseteq \{1,2,3\} \wedge o.attr2 \leq 1 \wedge op = r) \end{cases}$$

The policy can be decomposed into three rules, and a portion of each rule concerning the subject attributes is expressed as follows:

$$rule_1 : Subject.Attr_4 \subseteq \{3\}$$
$$rule_2 : Subject.Attr_4 \subseteq \{2,3\}$$
$$rule_3 : Subject.Attr_4 \subseteq \{1,2,3\}$$

Next, considering $rule_2$ as an example, calculate its AAM. Based on Definition 5.2, the rule space of $rule_2$ is expressed as:

$$\begin{aligned} Req_{rule_2} &= \{Y, \perp\} \times \{Y, \perp\} \times \{Y, \perp\} \times \{2,3\} \\ &= \{(Y,Y,Y,2), (Y,Y,\perp,2), (Y,\perp,Y,2), (\perp,Y,Y,2), \\ &\quad (Y,\perp,\perp,2), (\perp,Y,\perp,2), (\perp,\perp,Y,2), (\perp,\perp,\perp,2), \\ &\quad (Y,Y,Y,3), (Y,Y,\perp,3), (Y,\perp,Y,3), (\perp,Y,Y,3), \\ &\quad (Y,\perp,\perp,3), (\perp,Y,\perp,3), (\perp,\perp,Y,3), (\perp,\perp,\perp,3)\} \end{aligned}$$

According to the previous constraint, for Tenant 2, only $(\perp,\perp,\perp,2)$ and $(\perp,\perp,\perp,3)$ satisfy $rule_2$. Therefore, these two requests do not have a weight of 0. The remaining requests have weights of 0. We assume that the security administrator sets the weights based on the request frequency and that these two requests exhibit the same frequency. Therefore, the AAM of $rule_2$ is expressed as follows:

$$\begin{aligned} \mathcal{A}_r(rule_2) &= \omega(\perp,\perp,\perp,2)\mathcal{A}_{req}(\perp,\perp,\perp,2) \\ &\quad + \omega(\perp,\perp,\perp,3)\mathcal{A}_{req}(\perp,\perp,\perp,3) \\ &\approx 0.5 \cdot 0.301 + 0.5 \cdot 0 \approx 0.151 \end{aligned}$$

In practice, a situation exists in which a user's VIP expires and becomes invalid. This phenomenon affects the AAM of the policy rules. Assuming that Candy's VIP expires, although no user with VIP3 exists, a person can subsequently subscribe to the VIP3 service. Therefore, Tenant 2 does not modify its original policy. At this stage, the set of requests in the $rule_2$ rule space does not change. However, the weight of requests $(\perp,\perp,\perp,3)$ is set to 0 because no credentials with VIP3 are apparent. Therefore, the AAM of $rule_2$ is changed to the following:

$$\mathcal{A}'_r(rule_2) = \omega'(\perp,\perp,\perp,2)\mathcal{A}'_{req}(\perp,\perp,\perp,2) = 1 \cdot 0 = 0$$

As Candy's VIP expires, only Bob can construct $(\perp,\perp,\perp,2)$, and the anonymity of $rule_2$ is weakened. Thus, when the attributes of the subject in the scenario change, they affect the AAM changes in the policy rules. Because the subject and object attributes exhibit the same status in ABAC, based on (8), the AAM of the policy is the average of the sum of the AAM of all rules in the policy. Therefore, when the subject and object attributes change, the AAM of the policy changes accordingly and can be measured anonymously using the anonymity evaluation method of the proposed policy.

TABLE III
SIMULATION SCHEME

Sim. scheme	Subject num	Attr num	Values in attr	Policy num	Attr in policy
Group 1	1k	10	5	10	4
Group 2	10k	10	5	10	4
Group 3	100k	10	5	10	4
Group 4	100k	8	5	10	4
Group 5	100k	12	5	10	4
Group 6	100k	10	7	10	4
Group 7	100k	10	10	10	4
Group 8	100k	10	5	20	4
Group 9	100k	10	5	30	4
Group 10	100k	10	5	10	5
Group 11	100k	10	5	10	6

VI. SIMULATION

The evaluation of policies (including attribute data) from organizations is ideal for use in simulations. However, we are not aware of any suitable or publicly available policies from organizations. Xu et al. [26] used sample and randomly generated policies to test the effectiveness of their mining algorithms. The sample policies in [26] are not based directly on specific real-world case studies, but are intended to be similar to policies available in the application domains for which they are named. However, the AAM cannot be tested using sample policies as the latter are small-scale policies. Compared with the sample policy, the randomly generated policies have a larger scale. Large-scale random generation policies can effectively validate the AAM. The simulation groups contained the subject attribute assignment schemes, policies, and access requests. We analyzed the simulation results, identified the factors affecting anonymity, and proposed suggestions to improve anonymity.

A. Simulation Scheme

In practice, the difference between various attributes is the attribute type. Attribute types can be summarized as Boolean, numeric, or set. The range of the numeric type may be real numbers, integers, or a set of integers. The range of the set is any subset of itself, and the range of the Boolean only contains "true" and "false." Clearly, only the numerical range may be an infinite set. In practice, because the number of subjects is finite, the values used in the numerical range are finite sets. The attribute type affects only the number of values in the domain. Therefore, in the simulation, the attributes only differ in the number of values (for example, five values in the range, where the range is 1,2,3,4,5), and the attribute types do not differ from one another. The simulation results are presented in Table III.

In Table III, Columns 2, 3, 4, 5, and 6 show the numbers of corresponding items (such as 100 k subjects). In any group, the probability that an attribute is assigned $\perp$ is 0.2. For example, in the case of $attr_i$, $P(\perp) = 0.2$, $P(value_i) = 0.8/(|Domain_i| - 1)$.

This policy is protected in the simulation. Constraints do not affect policy anonymity in protected policies. Therefore, the

TABLE IV
SIMULATION RESULTS

Groups	Subject anonymity statistics			Access request anonymity statistics				policy anonymity statistics		
	μ	σ	$m_{0.5}$	$\|Req\|$	μ	σ	$m_{0.5}$	μ	σ	$m_{0.5}$
Group 1	0.3869	0.3546	0.3333	3029	0.3383	0.5377	0.0000	0.3533	0.0215	0.3554
Group 2	2.6443	0.3798	2.6667	6242	2.5972	0.6409	2.5850	2.5934	0.0062	2.5927
Group 3	6.0250	0.1106	6.0268	6250	6.0256	0.1776	6.0224	6.0239	0.0018	6.0238
Group 4	6.0287	0.1082	6.0309	6250	6.0249	0.1805	6.0224	6.0258	0.0021	6.0254
Group 5	6.0209	0.1023	6.0224	6250	6.0246	0.1783	6.0224	6.0216	0.0014	6.0216
Group 6	4.0533	0.2249	4.0660	24010	4.0450	0.3658	4.0875	4.0424	0.0025	4.0413
Group 7	1.9724	0.4537	2.0000	98355	1.8765	0.7786	2.0000	1.8914	0.0073	1.8926
Group 8	6.0280	0.0876	6.0296	12500	6.0237	0.1804	6.0224	6.0248	0.0012	6.0250
Group 9	6.0316	0.0732	6.0327	18750	6.0265	0.1785	6.0444	6.0285	0.0019	6.0282
Group 10	3.3489	0.3133	3.3596	31248	3.3210	0.4842	3.3219	3.3212	0.0042	3.3209
Group 11	0.9465	0.5310	1.0000	127221	0.8422	0.7525	1.0000	0.8764	0.0125	0.8798

policies used in the simulation indicate only the attributes used. Each group generates valid requests and evaluates the anonymity of the requests based on the subject attribute assignment scheme and policy. Groups 1, 2, and 3 compared the effect of the number of subjects on anonymity; Groups 3, 4, and 5 compared the effect of the number of attributes on anonymity; Groups 3, 6, and 7 compared the effect of the number of values in attributes on anonymity; Groups 3, 8, and 9 compared the effect of the number of policies on anonymity; and Groups 3, 10, and 11 compared the effect of the number of values in the policy on anonymity. In all groups, the number of subjects was not less than 1,000, because if few subjects are present in the group, then access requests for re-identifing subjects will be extremely high. However, the simulation results for Group 1 (Table IV) show that at least 50% of the access requested the re-identification of subjects. Reducing the number of subjects will not allow us to understand the effects of various elements on anonymity.

B. Simulation

The simulation results are presented in Table IV.

In Table IV, the number of valid requests for Groups 3, 4, and 5 is the same, because the number of valid requests is limited.

$$MaxReq = P \times NVA^{NAP}, \tag{9}$$

where $MaxReq$ is the maximum number of valid requests, P the number of policies, NVA the number of values in the attribute, and NAP the number of policy attributes. For all the groups, σ represents the uniformity of the set of subjects that can construct valid requests. Here, $m_{0.5}$ is similar to μ. Meanwhile, the μ and $m_{0.5}$ of the AAM of the subject, request, or policy increase with the number of subjects. The μ and $m_{0.5}$ of the AAM of the subjects, requests, and policies decrease as the number of attributes and NAP increase. The number of attributes does not affect the AAM, which allows administrators to use rich attributes in the subject attribute assignment scheme, thus rendering the latter more flexible. The number of policies does not affect the AAM, which allows administrators to focus on a single policy. Thus, the anonymity of the policies does not affect each other. This feature facilitates multiple administrators in developing multiple policies simultaneously. The AAM reflects the effects of various factors on anonymity, which demonstrates its rationality.

Figs. 3 and 4 show the effects of various factors on the distribution of the AAM.

The following results were obtained based on the distribution of the AAM:

1) A positive correlation exists between the number of subjects and the anonymity of the subjects, requests, and policies.
2) The subjects can construct access requests by themselves, and the request cannot include an explicit identifier;
3) The number of attributes and the number of policies are independent of anonymity.

Result 1 inferred based on Figs. 3(c) and 4(c); result 2, based on Figs. 3(d), 3(e), 4(d), and 4(e); and result 3, based on Figs. 3(a), 3(b), 4(a), and 4(b). Based on Fig. 4, the weaker the anonymity, the greater is the distance between the bars; the stronger the anonymity, the smaller is the distance between the bars. Thus, if anonymity is strong, then the numerical difference in the AAM is insignificant, whereas if anonymity is weak, then the numerical difference in the AAM is significant. This feature is an advantage of the AAM, as it allows individuals with weak anonymity to be adjusted, which is more critical to the anonymity metric of the subject, request, or policy universal set than to individuals with strong anonymity. Thus, administrators and algorithms assign greater priority to individuals with weak anonymity than to those with strong anonymity. The distribution of the anonymity request evaluation was analyzed based on the simulation conditions provided in Table III.

Let rule r contain m′ $non-\perp$ and the corresponding m′ attributes in the request are $non-\perp$. The probability that s can construct request $\overrightarrow{req}$ is as follows:

$$p_{\overrightarrow{req}} = [P(non-\perp)]^{m'} \tag{10}$$

where $\overrightarrow{req} \in Req_r$. The probability that subject i can construct $\overrightarrow{req}$ is as follows:

$$P(|S_{\overrightarrow{req}}| = i) = C^i_{|S|} {p_{\overrightarrow{req}}}^i \cdot (1 - p_{\overrightarrow{req}})^{|S|-i} \tag{11}$$

During simulation, for any $\overrightarrow{req}, \forall s_i, s_j \in S_{\overrightarrow{req}}, p(s_i) = p(s_j) = \frac{1}{|S_{\overrightarrow{req}}|}$.

Therefore, $\mathcal{A}_{req}(\overrightarrow{req}) = -\sum_{i=1}^{|S_{\overrightarrow{req}}|} p(s_i) \log p(s_i) = \log |S_{\overrightarrow{req}}|$.

The theoretical distribution for Group 2 is shown in Fig. 3.

VII. Conclusion and Future Work

A. Conclusion

Herein, an anonymity metric known as the AAM is proposed to measure the risk of a subject being re-identified in ABAC. Anonymous access has become critical for protecting privacy in access control because it not only protects general privacy information, but also protects access to maintain privacy. To the best of our knowledge, this study is the first to prevent anonymity loss at the policy level. Furthermore, this study is the first to formalize the definition of anonymity in ABAC. An ABAC anonymous access reference framework was proposed herein by analyzing the characteristics of multiple existing ABAC frameworks that support anonymous access. This framework highlights the features of the ABAC framework that support anonymous access, which is the basis of anonymous access to ABAC. The simulation results revealed that the AAM can reasonably measure the anonymity of access requests, subject assignments, and policies. The AAM allows security administrators to rationally adjust policies and subject attribute assignments to protect subject privacy. A quantitative metric can be used as an objective or a constraint for the algorithm.

B. Future Work

The AAM measures independent access requests only. In the future, an evaluation method will be proposed for multiple associated access requests because not all implementations guarantee unlinkable access requests. The simulation in this study is simple, and the factors affecting anonymity are qualitatively analyzed in Section VI-B. In the future, we will perform more complex simulations and complete a quantitative analysis of factors affecting anonymity. The AAM are applicable to policy-mining algorithms for generating policies that satisfy anonymity requirements. In the future, a mining algorithm that uses a quantitative anonymity metric as a constraint should be developed since it is currently available. This algorithm will be able to automatically generate policies with strong anonymity.

References

[1] "Amazon web services (AWS) identity and access management (IAM)," 2023. [Online]. Available: https://docs.aws.amazon.com/iam/index.html

[2] U. M. Aftab, Z. Qin, Zakria, S. Ali, Pirah, and J. Khan, "The evaluation and comparative analysis of role based access control and attribute based access control model," in *Proc. 15th Int. Comput. Conf. Wavelet Act. Media Technol. Inf. Process.*, 2018, pp. 35–39.

[3] R. Ahuja and S. K. Mohanty, "A scalable attribute-based access control scheme with flexible delegation cum sharing of access privileges for cloud storage," *IEEE Trans. Cloud Comput.*, vol. 8, no. 1, pp. 32–44, First Quarter 2020.

[4] A. Basu, T. Nakamura, S. Hidano, and S. Kiyomoto, "k-anonymity: Risks and the reality," in *Proc. IEEE TrustCom/BigDataSE/ISPA*, 2015, pp. 983–989.

[5] G. Batra, V. Atluri, J. Vaidya, and S. Sural, "Enabling the deployment of ABAC policies in RBAC systems," in *Proc. 32th IFIP Annu. Conf. Data Appl. Secur. Privacy*, 2018, pp. 51–68.

[6] Y. Benkaouz, M. Erradi, and B. Freisleben, "Work in progress: K-nearest neighbors techniques for abac policies clustering," in *Proc. ACM Int. Workshop Attribute Based Access Control*, 2016, pp. 72–75.

[7] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy*, 2007, pp. 321–334.

[8] T. Bui, D. S. Stoller, and J. Li, "Greedy and evolutionary algorithms for mining relationship-based access control policies," *Comput. Secur.*, vol. 80, pp. 317–333, 2019.

[9] D. M. Green and I. Miers, "Bolt: Anonymous payment channels for decentralized currencies," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2017, pp. 473–489.

[10] A. E. M. Hadj, Y. Benkaouz, B. Freisleben, and M. Erradi, "ABAC rule reduction via similarity computation," in *Proc. Int. Conf. Networked Syst.*, 2017, pp. 86–100.

[11] B. Hampiholi and G. Alpár, "Privacy-preserving webshopping with attributes," in *Proc. IEEE Symp. Privacy-Aware Comput.*, 2017, pp. 25–36.

[12] M. Harbach, S. Fahl, M. Brenner, T. Muders, and M. Smith, "Towards privacy-preserving access control with hidden policies, hidden credentials and hidden decisions," in *Proc. Annu. Int. Conf. Privacy Secur. Trust*, 2012, pp. 17–24.

[13] V. C. Hu et al., "Guide to attribute based access control (ABAC) definition and considerations (draft)," NIST special publication 800.162, pp. 1–54, 2013.

[14] C. J. John, S. Sural, and A. Gupta, "Optimal rule mining for dynamic authorization management in collaborating clouds using attribute-based access control," in *Proc. IEEE 10th Int. Conf. Cloud Comput.*, 2017, pp. 739–742.

[15] K. H. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-based signatures," in *Proc. Cryptographer's Track RSA Conf.*, 2011, pp. 376–392.

[16] I. Molloy et al., "Mining roles with multiple objectives," *ACM Trans. Inf. Syst. Secur*, vol. 13, 2010, Art. no. 36.

[17] A. Ouaddah, H. Mousannif, A. E. A. Kalam, and A. A. Ouahman, "Access control in the Internet of Things: Big challenges and new opportunities," *Comput. Netw.*, vol. 112, pp. 237–262, 2017.

[18] A. Pfitzmann and M. Hansen, "A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management," 2010.

[19] H. Qi, X. Di, and J. Li, "Formal definition and analysis of access control model based on role and attribute," *J. Inf. Sec. Appl.*, vol. 43, pp. 53–60, 2018.

[20] D. Servos and L. S. Osborn, "Current research and open problems in attribute-based access control," *ACM Comput. Surv.*, vol. 49, pp. 65:1–65:45, 2017.

[21] E. C. Shannon, "A mathematical theory of communication," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 5, pp. 379–423, 1948.

[22] P. L. Sondeck, M. Laurent, and V. Frey, "Discrimination rate: An attribute-centric metric to measure privacy," *Ann. Telecommun.*, vol. 72, pp. 755–766, 2017.

[23] L. Sweeney, "k-anonymity: A model for protecting privacy," *Int. J. Uncertainty Fuzziness Knowl.-Based Syst.*, vol. 10, pp. 557–570, 2002.

[24] Z. Wan, J.-E. Liu, and H. R. Deng, "HASBE: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 743–754, Apr. 2012.

[25] Z. Xu and S. S.D., "Mining attribute-based access control policies from RBAC policies," in *Proc. 10th Int. Conf. Expo Emerg. Technol. Smarter World*, 2013, pp. 1–6.

[26] Z. Xu and D. S. Stoller, "Mining attribute-based access control policies," *IEEE Trans. Dependable Secure Comput.*, vol. 12, no. 5, pp. 533–545, sep./Oct. 2015.

[27] Z. Xu and D. S. Stoller, "Mining attribute-based access control policies from logs," in *Proc. IFIP Annu. Conf. Data Appl. Secur. Privacy*, 2014, pp. 276–291.

[28] X. Yao, H. Liu, H. Ning, L. T. Yang, and Y. Xiang, "Anonymous credential-based access control scheme for clouds," *IEEE Cloud Comput.*, vol. 2, no. 4, pp. 34–43, Jul./Aug. 2015.

[29] E. Yuan and J. Tong, "Attributed based access control (ABAC) for web services," in *Proc. IEEE Int. Conf. Web Serv.*, 2005, pp. 561–569.

[30] H. T. Yuen, K. J. Liu, H. M. Au, X. Huang, W. Susilo, and J. Zhou, "k -times attribute-based anonymous access control for cloud computing," *IEEE Trans. Comput.*, vol. 64, no. 9, pp. 2595–2608, Sep. 2015.

[31] Y. Zhang, J. Li, X. Chen, and H. Li, "Anonymous attribute-based proxy re-encryption for access control in cloud computing," *Secur. Commun. Netw.*, vol. 9, no. 14, pp. 2397–2411, 2016. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/sec.1509

[32] Y. Zhang and B. Zhang, "An ACO-based algorithm for efficient XACML policy evaluation," in *Proc. Int. Conf. Control Automat.*, 2017.

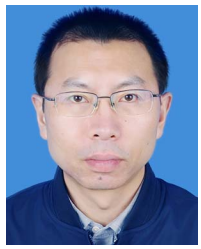
[33] L. Zhou, C. Su, Z. Li, Z. Liu, and P. G. Hancke, "Automatic fine-grained access control in SCADA by machine learning," *Future Gener. Comp. Syst.*, vol. 93, pp. 548–559, 2019.

Runnan Zhang received the BS degree from Zhongbei University, Taiyuan, Shanxi, China, in 2014. He is currently working toward the PhD degree with Xidian University. His current research interests include research on access control models and quantitative evaluation, conflict detection and resolution, and the optimization of security policies.

Gang Liu received the MS and PhD degrees in computer science and technology from Xian Jiaotong University, Xi'an, China, in 2001 and 2004, respectively. From 2005 to 2006, he was an associate professor with the Guangdong University of Petrochemical Technology, Guangdong, China. His research interests include high-speed computer network and multimedia technology. Since 2007, he has been a faculty member with the School of Computer Science and Technology with Xidian University. His major research interests include embedded system, information security and trusted computing.

Hongzhaoning Kang received the BS degree in software engineering from Shandong University, Jinan, Shandong province, in 2017. Now he is currently working toward the PhD degree in computer science and technology with Xidian University. His research interest includes zero trust and access control model.

Quan Wang received the BS, MS, and PhD degrees in computer science and technology from Xidian University, Xi'an, China. He is now a professor with the School of Computer Science and Technology, Xidian University, China. He serves as the dean with the School of Computer Science and Technology. His current research interests include input and output technologies and systems, image processing, and image understanding.



Bo Wan received the BS, MS, and PhD degrees from Xidian University, Xi'an, China. He is currently a professor with the Key Laboratory for Smart Human-Computer Interaction and Wearable Technology and the College of Computer Science and Technology, Xidian University. His current research interests include human-computer interaction and embedded system.

Nan Luo received the BS degree in computer science and technology, and the MS and PhD degrees in computer architecture from Xidian University, Xi'an, in 2009, 2012, and 2017. He is currently associated with the Computer Engineering Department, Xidian University. He is also a member of China Computer Federation(CCF) since 2012. His research interests include point cloud processing, 3-D scene reconstruction, and understanding.