RESEARCH ARTICLE OPEN ACCESS

Anonymous Authentication in the SAML 2. 0 Protocol Based On Algebraic MAC

Shiwei Wen *

*(Department of computer science, Jinan university)

----------**********----------

Abstract:

In this paper, the Security Assertion Markup Language (SAML) protocol is analyzed and the existing privacy leakage problems are pointed out. A method for anonymous authentication in the SAML 2. 0 protocol proposed by adding a lightweight Attribute-based algebraic MAC (Message Authentication Code) anonymous credential solution. Due to the anonymity and security of the authentication solution, both the identity provider and the service provider can not be traced back to the real name of the pseudonym, so as to realize the protection of users' privacy data.Experiments show that the method we design is feasible and computationally efficient.

Keywords **—SAML，anonymous credential， algebraic MAC;privacy-preserving， federated identity management.**

----------**********----------

I. INTRODUCTION

With the rapid development of the Internet, cooperation between various network service providers is becoming more and more frequent. A service often needs multiple service providers to cooperate with each other. For example, in online shopping applications, users usually visit the online shopping mall to purchase goods and then pay via online banking. At this time, the online shopping mall and the bank form a business relationship. Generally, users need to use multiple account information managed by different service providers to complete such services. This interactive switching reduces the user experience to a certain extent. To solve this problem, Federated Identity Management (FIM) [1] came into being. It provides a loosely coupled cross-domain authentication and authorization model that enables service providers that have a partnership to trust each other to form a federation that shares their locally managed identity and security credentials. As a result, users can log in with only one of their account information at one service provider to interact with many other trusted service providers.

Currently, Common Federal Identity Management protocol specifications include the Security Assertion Markup Language (SAML) protocol [2], the OpenID Specification [3], the Windows CardSpace Specification [4], and so on. Among them, the SAML protocol has been widely used because of its strong security mechanism and flexibility. Many service providers have launched their own SAML products, such as Sun's OPENSSO [5], Microsoft's ADFS and WS-Federation [6], and Pingidentity's Pingfederate [7].

However, in federated identity management, the shareability of identity data also poses a risk of disclosure of user privacy. The identity provider can only passively provide the user identity attribute to the service provider, and the privacy data of some users may be exposed to the service provider, thus causing the hidden danger of the user's privacy. In order to solve this problem of privacy disclosure, this paper proposes an anonymous authentication method for privacy-preservable SAML protocol, which adopts the property-based lightweight anonymous credential scheme proposed by Chase et al. [8] Combined with the original SAML protocol, users can choose their

identity attributes to hide through Pedersen's commitment scheme. After the authentication is passed, a commitment is made to create a pseudonymous identity for the user that is not associated with the user, so as to achieve the privacy protection of the federated identity management.

The first section of this article describes the federated identity management and the SAML protocol. The second section introduces the theoretical basis and technical background of the scheme, including algebraic MAC algorithm, commitment algorithm and zero knowledge proof. The third section introduces the anonymous SAML The design framework of the federation authentication method; we have implemented and tested the program and introduced the test results in Section 4; In section 5 summarizes the current work and discusses the related research questions.

II. PRELIMINARIES

A. Security Assertion Markup Language

The Security Assertion Markup Language (SAML) standard is a framework created by the Organization for the Advancement of Structured Information Standards for the transmission of safety-related information between online business partners. In more detail, SAML defines a generic XML framework for exchanging authentication information between entities. The authentication information is represented in SAML assertion and can be transmitted in different trusted system domains.

B. Message Authentication Code

Message Authentication Code (MAC) is a cryptographic tool used to verify the integrity of a message. Integrity refers to the sender and the content of the message that have not been tampered with. MAC is a symmetric cipher algorithm, that is, the sender and the authenticator need to share a key. The sender of the message, when sending the message, uses this key to generate an authentication code (also called a tag) for this message and sends the message with the authentication code to the verifier at the same time. After the verifier receives the message, it can authenticate the message and the authentication code by using the corresponding authentication algorithm and key. If the authentication passes, it indicates that the message has not been tampered with. Otherwise, the message indicates that the message is forged or sent by others The contents of the process have been tampered with. Due to the efficiency of message authentication, it is widely used in numerous security protocols on the Internet

The security of a MAC scheme is usually defined by the EU-CMA model under selective message attack. Assuming that the attacker sends some messages and their corresponding authentication codes, the attacker still can not fake a new message The authentication code enables it to pass the verification of the authentication algorithm. Commonly used MAC schemes are generally based on block cipher or hash function design. Algebraic MAC is a MAC scheme constructed using group elements and group operations. Chase et al. [8] present two MAC schemes constructed in prime order group and prove that they satisfy EU-CMA security.

C. Commitment Scheme

In the digital world, promises made by users often require promises to be made to ensure their security and fairness. Commitment agreement runs between the sender and the receiver of the message, mainly including two phases, namely commitment generation phase and promised to open phase. During the commitment phase, the sender first chooses a random key K, encapsulates the content m to be committed, generates a commitment to m, and sends the commitment to the receiver. In the Commitment Open phase, the sender sends the key K to the receiver, and the receiver can open the encapsulated commitment to get m. In commitment, the promise can be compared to a locked box. In the first phase, the sender locks the promised content in the box, and then sends the box to the recipient. In the second stage, the receiver can open the box after obtaining the key to verify. A secure promise scheme guarantees that once a promise has been made, there is no other way for the receiver to open the promise before it gets the key (otherwise the promise is meaningless) and the sender can not change the promised one even after making the promise. This is also the nature of the promise

agreement that needs to be fulfilled, namely, hiddenness and binding. Hidden requirements that the receiver can not get any information about promised content m only from the promise, and binding requires that the sender can not deceive the promised content of the receiver.

D. Zero knowledge proof protocol

The zero-knowledge proof protocol was proposed by Goldwasser, Milcali and Rackoff in the 1980s [9] and has been widely used in various fields of computer science. Run a zero-knowledge proof protocol that proves to be able to prove to the verifier that an assertion holds without providing additional information. For example, the prover can prove to the verifier that he knows the answer to a mathematical question, without telling the verifier the content of the answer, thus completing the proof without revealing any knowledge. Zero-knowledge proof protocols need to satisfy three basic properties, namely, integrity, reliability and zero-knowledge. Integrity means that the verifier must be able to verify successfully when the prover truly knows the answer to the question. Reliability means that the prover can be verified by the prover if and only if the prover knows the answer to the question. , Then the verifier can not verify the pass; zero knowledge to ensure that the verifier can not get more information from the certificate.

The non-interactive zero-knowledge proof protocol is a variation in the zero-knowledge proof protocol, in which the prover and the verifier need not interact with each other to complete the proof process. In the scheme proposed by Blum, Feldman and Micali [10], when the prover and the verifier share a universal string, the prover only needs to send a message to the verifier, and the verifier can judge the prover.

III. CONSTRUCTION

This section mainly describes the design of the anonymous authentication method under the federal model of SAML protocol. We adopt the lightweight algebraic MAC anonymous credential scheme proposed by Chase et al. [8]. Construct a pseudonyms identity that can not be associated with it, In this section, we describe the security attributes required by the system. Then, we explain the basic steps of the system composition, how to use the algebraic MAC anonymous authentication scheme to construct the non-linkable Pseudonym [11] and how to apply it to the SAML protocol.

A. System decription

1) System Establish

The system is built by the identity provider, who executes two algorithms, the system parameter generation algorithm and the identity provider's key generation algorithm.

System parameter generation algorithm: Identity provider to run Setup (1k) algorithm to generate the system's public parameters params.

Key Generation Algorithm: The identity provider runs the CredKeygen (params) algorithm to generate a key that is used to subsequently issue the user's credentials.

2) Credential Issued:

According to the issuing algorithm proposed by Chase et al., The identity provider uses the private key to issue the algorithm BlindIssue and BlindObtain to the set of attributes provided by the user to obtain the algebraic message obtained from the identity provider's private key to the credential Authentication code and certificate, the certificate contains the blind value of the user's secret USK, the public attribute set S, and the certificate expiration Exp.

3) Pseudonym Generation

In the SAML protocol, the identifier uniquely identifies the entity, the pseudonym identifier that the user uses to access the service provider is generated by $\text{PersistentID}_{SERVICE} = {SID}^{USK}_{SERVICE}$, SID = H (ServiceName), and H is a hash function so that the resulting pseudonym identifier is based on the user USK is the secret value of the user. After the identity provider issues a certificate to the user, the user performs a pseudonym generation algorithm for the service name of each service to obtain a pseudonyms identifier, and executes the certificate presented by Chase et al. Show algorithm to prove that he has an algebraic MAC corresponding to the attributes in the voucher, and (S, EXP, USK) associate the set of user-exposed attributes with the pseudonym identifier to form a pseudonym identity information.

In more detail, the credential presentation algorithm Show outputs three Pedersen Commitment Values [13] (Cm0, Cm1, Cm2), respectively, to commit (S, EXP, SUB), so we use a noninteracting zero knowledge proof protocol to prove Its pseudonym is legal.

$$NIZK\{(USK, r_0, r_1, r_2) :$$

$$C_{m_0} = h^{r_0} g^{S} \wedge C_{m_1} = h^{r_1} g^{EXP} \wedge C_{m_2} = h^{r_2} g^{USK} \wedge PersistentID_{SERVICE} = SID^{USK}_{SERVICE}$$

4) Show algorithm

Show algorithm is a zero-knowledge interaction protocol between user and identity provider, which verifies the legitimacy interaction process of anonymous voucher provided by user.According to the Show Show algorithm proposed by Chase et al., Run the Show algorithm to generate voucher proof, identity provide Run ShowVerify algorithm to verify the certificate.

We made minor modifications to the existing SAML v2.0 persistence identifier federation protocol are shown in Figure 1.

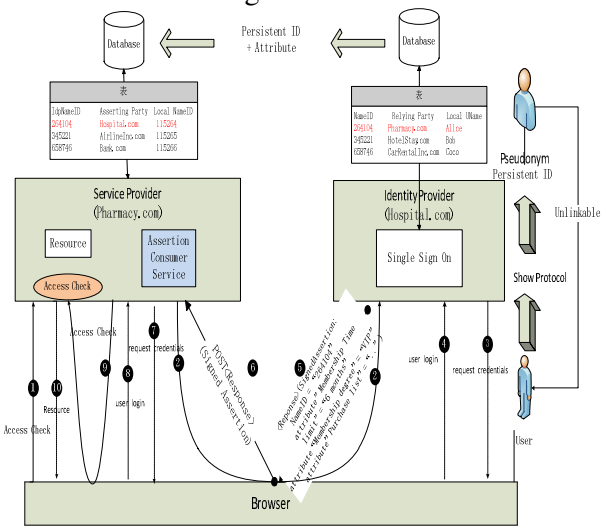

Figure 1

B. The Protocol

1. The user Alice tries to access a resource at the service provider (Pharmacy.com).

2. The service provider (Pharmacy.com) needs the user's legal identity information and generates a SAML authentication request for the user from the identity provider The service provider sends the HTTP redirect to the identity provider (Hospital.com). HTTP redirection includes SAML <AuthnRequest>, which requests that the identity provider provide the user's assertion and the user's persistence identifier.

3. Identity Provider (Hospital.com) asks the user to provide valid credentials.

4. The user sends the local blind certificate to the single sign-on server.The identity provider verifies the certificate according to the Show protocol mentioned in the previous section, and after all the information is verified, the set S information is provided according to the public attribute of the user in the identity providing A database of identity information that is not associated with the credential is created. The persistence identifier of the identity information is a pseudonym of the user, and the identity information is kept until the expiration date of the credential.

5. The identity provider generates a SAML response that contains a verified pseudonym ID record, which the SSO service sends back to the browser, and the user signs the response with the private key SUB The HTML FORM contains the SAML response, which contains SAML Assertions. The name identifier used in the assertion is a persistent identifier (pseudonym) that provides the attribute membership level, membership duration, and purchase list.

6. The browser sends an HTTP POST message that contains the SAML response to be sent to the service provider.

7. The Assertion Consumer service of the Pharmacy.com service provider validates the digital signature on the SAML response and validates the SAML assertion and then uses the provided name identifier to find out if the previous union has been established.If the previous Union, name identifier mapped to the local account, then jump to Step 9.

8. The user needs to provide credentials to log in to the service provider's registered account, the name identifier and the account are associated.

9. The service provider checks if the local account has permission to access the Pharmacy.com website and the target resource.

If the check is passed, the target resource is returned to the browser..

IV. PERFORMANCE

Simulation experiments are carried out in macOS system. Algebraic MAC algorithm mainly uses the petlib library written by George Danezis of University College London. The library provides the implementation interface of algebraic MAC algorithm, and can directly call the interface function.

We use Python 3.5 to simulate the credential issuance process between server and client as well as the voucher presentation algorithm and experiment with a 1.8GHz dual-core Intel Core i5 processor and a MacOS environment with 8.0GB RAM. Time We did 10 sets of experiments in milliseconds (ms), and we measured the time-consuming process of issuing certificates and generating alias identifiers on the server and the client.The average time required to issue each certificate was 79ms seconds, where the average client computing time The average time spent on the server is 76 ms.We also measured the time spent executing the voucher generation algorithm on the client and the server, which took an average of 91 ms, including an average of 16 ms for client computing and 75 ms for server computing, as shown in Fig. 2.

V. CONCLUSIONS

We combine the algebraic MAC anonymous credential scheme into the current federation model of SAMLv2.0 protocol. After the authentication succeeds, we create a un-linkable pseudonymous identity for the user so that the user data is not passed by the identity provider and the service provider after the federation. Collusion related accounts collected to protect the privacy of users.The experimental results show that our method is computationally efficient and low cost and can be applied to the SAML protocol which is feasible and effective.Future work will be the anonymous authentication method Into other SAML extensions such as WS-Security [14], XACML (Extensible Access Control High-Definition Markup Language) [15].

REFERENCES

[1] Chadwick D W. Federated identity management[M]//Foundations of security analysis and design V. Springer Berlin Heidelberg, 2009: 96-120

[2] Hughes J, Maler E. Security assertion markup language (saml) v2. 0 technical overview[J]. OASIS SSTC Working Draft sstc-saml-tech-overview-2.0-draft-08, 2005: 29-38

[3] Recordon D, Reed D. OpenID 2.0: a platform for user-centric identity management[C]//Proceedings of the second ACM workshop on Digital identity management. ACM, 2006: 11-16

[4] Bertocci V, Serack G, Baker C. Understanding windows cardspace: an introduction to the concepts and challenges of digital identities[M]. Pearson Education, 2007

[5] Enterprise S O. 8.0 Technical Overview[J]. Santa Clara, CA: Sun Microsystems Inc, 2008.

[6] *Weerawarana S, Curbera F, Leymann F, et al. Web services platform architecture: SOAP, WSDL, WS-policy, WS-addressing, WS-BPEL, WS-reliable messaging and more[M]. Prentice Hall PTR, 2005.*.

[7] Identity P. PingFederate Overview[J]. 2006

[8] Chase M, Meiklejohn S, Zaverucha G. Algebraic MACs and keyed-verification anonymous credentials[C]//Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2014: 1205-1216

[9] Goldwasser S, Micali S, Rackoff C. The knowledge complexity of interactive proof systems[J]. SIAM Journal on computing, 1989, 18(1): 186-208.

[10] *Blum M, Feldman P, Micali S. Non-interactive zero-knowledge and its applications[C]//Proceedings of the twentieth annual ACM symposium on Theory of computing. ACM, 1988: 103-112*

[11] Lysyanskaya A, Rivest R L, Sahai A, et al. Pseudonym systems[C]//Selected Areas in Cryptography. 1999, 1758: 184-199.