Design and Implementation of the *idemix* Anonymous Credential System

Jan Camenisch and Els Van Herreweghen
IBM Research, Zurich Research Laboratory
8803 Rüschlikon
Switzerland
{jca,evh}@zurich.ibm.com

ABSTRACT

Anonymous credential systems [8, 9, 12, 24] allow anonymous yet authenticated and accountable transactions between users and service providers. As such, they represent a powerful technique for protecting users' privacy when conducting Internet transactions. In this paper, we describe the design and implementation of an anonymous credential system based on the protocols developed by [6]. The system is based on new high-level primitives and interfaces allowing for easy integration into access control systems. The prototype was realized in Java. We demonstrate its use and some deployment issues with the description of an operational demonstration scenario.

Categories and Subject Descriptors

E.3 [**Data**]: Data Encryption—*Public key cryptosystems*

General Terms

Design, Security

Keywords

Privacy, Anonymous Credential Systems, Cryptographic Protocols

1. INTRODUCTION

The protection of users' privacy when performing Internet or web-based transactions is an important factor in the acceptance and use of Internet and web services.

Solutions for minimizing release of personal information can be based on one of many proposed techniques for anonymizing the transport medium used between users and service providers, e.g., [26, 18, 27]. This may anonymize the user towards outsiders and, if desired, towards the service provider.

CCS'02, November 18–22, 2002, Washington, DC, USA.

Service providers may require authentication (e.g., for controlling access to resources) or accountability of users' actions, in which case users need to prove their identity, or at least possession of a certificate or capability of a certain type. Such a certificate may contain a pseudonymous identity of the user, or contain only the necessary attributes required for accessing a certain service. However, when using certificates as defined by X.509 [11] or SPKI [2], or even certificates specifically constructed for conveying policy or authorization information as in Keynote [3], different uses of the same certificate still remain linkable to each other. They can eventually identify a user through a combination of context and addressing information from one or a series of transactions. Moreover, the transaction in which the certificate was issued can be linked to the transaction where it is used and thus, it the issuer and the verifier collude, the user can identified directly.

These linkabilities can be avoided by using an anonymous credential system (also called pseudonym system) [8, 9, 12, 24]. In such a system, the organizations (service providers and credential issuers) know the users only by pseudonyms. Different pseudonyms of the same user cannot be linked. Yet, an organization can issue a credential to a pseudonym, and the corresponding user can prove possession of this credential to another organization (who knows him by a different pseudonym), without revealing anything more than the fact that the user owns such a credential.

In this paper, we describe the design and implementation of *idemix* (short for *ide*ntity *mix*), a prototype of the credential system by Camenisch and Lysyanskaya [6]. We describe the *idemix* functionality using high-level primitives. These primitives allow reasoning about security and privacy features, while hiding the complexity of the cryptographic protocols, as well as the differences between actual protocols realizing the same primitive. We also developed additional functionality for service providers and credential issuers to configure and enforce resource access control and credential issuing decisions. As we demonstrate with an example, this allows the use of the prototype in developing actual applications using concepts of anonymous and attribute-based authentication and access control.

After describing the functionality of the credential system protocols in Section 2, we describe in Section 3 the high-level primitives. Section 4 describes the architecture and implementation of the prototype implementing these protocols, as well as the additional modules developed to

5. AN EXAMPLE SCENARIO: AN ANONYMOUS SUBSCRIPTION TO THE NEW YORK TIMES

In this section, we demonstrate the use of the prototype by user and organization applications. We define four organizations: a Root Pseudonym Authority (PA), a bank (ARGENTIX), the New York Times news subscription service (KIOSK), and the New York Times news service (NYT). NYT serves items in its cartoons section only upon verification of a subscription credential issued by KIOSK; KIOSK, in turn, issues such a credential upon verification of a (one-show) $10 credential; ARGENTIX issues such a credential based on proof of an (non-anonymous, *idemix*-external) payment, combined with the verification of a PA root credential. PA unconditionally grants root pseudonyms and credentials (In a more realistic scenario, a user could be required to show an external certificate when registering a root credential, as discussed in Section 6.3).

5.1 Creating and Configuring the User and Organizations

A demo setup program uses the NymSystem user and organization creation facilities to create one user and four organizations. It assigns IP addresses and port numbers to the four organizations, as well as SSL Certificates which are created using the KeyMan [14] PKI management tool. It also creates rules for the three organizations (see below). The initialization program creates persistent data sets for each of the four entities, and initializes each of the organization's data sets with its own OrgNymSystemData key material. The user's data set is initialized with the user's UserNymSysData key information, as well as all the organizations' public information (*idemix* public key, addresses, SSL certificates, rules).

PA and KIOSK use the default RequestGranter and OrgRequestProcessor as they do not deal with ExternalConditions or ExternalResources; ARGENTIX implements its own ArgentixRequestGranter defining the verification of the credit card receipt; NYT, finally, implements its own NYTRequestProcessor with handleResourceRequest() mapping a resource request (URL) into the actual contents of a web page.

5.2 User Credential Manager and Browser Plug-In

Based on the *idemix* prototype, [25] describes the design and implementation of a Credential Manager implemented as a plug-in to a WBI [1] browser proxy. Figure 7 shows an instance of the Credential Manager in a scenario with the four organizations initialized as described above. This Credential Manager popped up after a user entered a "http://-www.nyt.com/cartoons" URL in his browser URL window. The Credential Manager then allows the user to view the relevant condition tree applying to the request, the conditions for which he has the necessary credential or external proof (tick-off symbol) and the credentials he already owns in his credentials purse. E.g., he already has a credential from PA.

The two conditions by ARGENTIX are related to (1) a ShowCondition: showing the credential from the PA, and (2) an ExternalCondition giving a reference to a credit-card payment. This reference is implemented by, e.g., a serial number of the payment. The ExternalCondition shows up in the condition tree; but as the payment reference is not an *idemix* credential, there is no corresponding credential in the credentials purse.

When clicking on a condition in the tree, the details are shown in the selected condition window, e.g., the KIOSK requires a one-show credential (multi-show = false) issued by ARGENTX with subtype = 10. It also allows the user to chose local identifiers (e.g., "kiosknym") for the pseudonyms he establishes with the different organizations, and to GET and SHOW credentials. After fulfilling all the conditions, the requested contents (cartoons page) show up in the browser window.

6. DEPLOYMENT CONSIDERATIONS

In this section, we discuss some issues related to the deployment of *idemix*.

6.1 Deploying *idemix* as a Privacy-Enhanced Public-Key Infrastructure with External Certification

In an operational system, public information about organizations (whether or not regularly updated) needs to be certified: users need authenticated information about where to get or show a credential, what is the *idemix* public key of an organization, and what is its SSL certificate. Also, a real Root Pseudonym Authority can only guarantee total accountability (global anonymity revocation) if a user's real-world information was authenticated upon registering the root pseudonym.

A deployment environment using *idemix* credentials as a (privacy-enhanced) Public-Key Infrastructure needs to provide hooks for an external Public-Key infrastructure (PKI). In this external PKI, users and organizations have public-key certificates issued by a Certification Authority. We call this authority Certifix, although it may be an existing Certification Authority; the only requirement being that it can issue organizations' "*idemix* certificates" certifying the whole set of an *idemix* organization's authenticated information. Depending on implementation and deployment choices, such an organization's *idemix* certificate may contain *idemix* keys, address and SSL information, and access rules.

Users also have Certifix certificates and use them to authenticate "real-world" information during root pseudonym registration.

6.2 The Role of Authenticated Communication in Linking Transactions Based on *idemix* Authentication

Authenticated communication (e.g., using SSL server authentication) allows users to authenticate organizations with which they register a pseudonym, to which they show a credential or from which they obtain a credential. When several protocol executions (including application-level resource requests) are linked by an authenticated communication channel, this also allows servers to securely link *idemix* authentication (who showed the correct credential) with providing the resource (who gets the data).

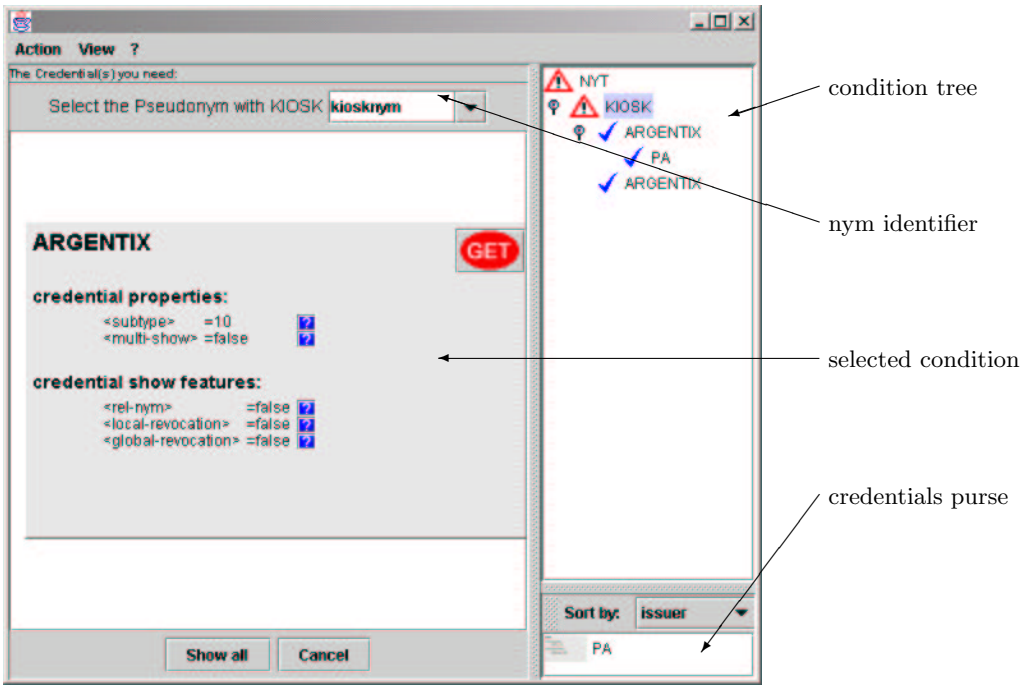

Figure 7: UserCredential Manager

6.3 Infrastructural Issues: User Registration and Organization Updates

In a real deployment environment, users and organizations dynamically join the system, and organizations may periodically update public information such as rules, public keys, addresses, or SSL information (their *idemix* certificates).

A user joining the system may not only need to authenticate using their real-world certificate when registering a root pseudonym with the root pseudonym authority; he may also have to prove registration (or payment of a license). This may be realized by the Root Pseudonym Authority checking an additional condition.

Also, organizations' *idemix* certificates need to be distributed and updated in an efficient way. A separate InfoServer entity may serve as a central repository for up-to-date organizations' *idemix* certificates. Organizations post their *idemix* certificates to the InfoServer; a certificate update may update whole or part (e.g., only new rules set) of an organization's *idemix* information. Revocation issues may be dealt with by Certificate Revocation Lists (CRLs) issued by the InfoServer; or avoided by issuing short-lived *idemix* certificates.

6.4 *Idemix*, Trust Management and Attribute-Based Access Control

Decentralized trust management, a term introduced by Blaze, Feigenbaum and Lacy [4], deals with access control and authorization in distributed environments. Different trust management systems and languages have been proposed, e.g., [3, 21, 20, 19, 23, 22, 15]; a credential or certificate modeled by those systems binds a public key to attributes and/or authorizations. Access control and trust establishment policies controlled by resource owners allow authorization decisions based on these attributes and authorizations, or on derived role assignments. Trust between the verifier and the issuer of a credential can be modeled through delegation of attribute authority, which allows a resource owner to delegate authority over an attribute to another entity. Some work also deals with automatic collection or discovery of (part of) certificate chains (e.g., [23, 22, 19]).

The access control rules and conditions language introduced in Section 4.6.1 was designed to illustrate the capabilities and usage of *idemix* for configuring anonymous attribute-based access control in a prototype application environment. However, as *idemix* certificates can be used to formulate any assertion (also identity assertions, if required), *idemix* attribute-based authentication can support any of the trust management models mentioned; also, in a distributed system where credential verifiers do not know credential issuers (and their keys) on beforehand, credential verification conditions and rules can be modified to express more general authority delegation and trust management policies (e.g., "I accept a credential issued by an issuer satisfying trust or delegation condition Y" instead of "I accept a credential from issuer X." As the issuers in a certificate chain can be publicly known entities, also automatic certificate chain collection could be realized.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented the design and implementation of the *idemix* anonymous credential system. The high-level primitives that were introduced to define the system's interfaces are easy to use and understand, and easy to extend to include new options and features. We also presented an example infrastructure for applications to exploit *idemix* authentication in an access control infrastructure.

The *idemix* system as implemented and presented here,

does not yet include features such as *all-or-nothing* non-transferability, or use for signature generation. A new NymSystem library is being implemented which will incorporate these additional features.

Deployment of *idemix* as a privacy-enhanced PKI also requires features supported by the core NymSystem, such as changing of organizations' public *idemix* keys, or for efficient revocation of credentials. We are currently developing the protocols supporting these features.

Acknowledgements

The authors are grateful to Marco Bove, Endre Bangerter, Roger Mathys, Martin Schaffer, and Dieter Sommer for their amazing Java programming making the idemix prototype reality.

8. REFERENCES

[1] R. Barrett, P. P. Maglio, and D. C. Kellem. WBI development kit. http://www.almaden.ibm.com/cs/wbi/.

[2] S. Bellovin and P. Metzger. Simple Public Key Infrastructure (SPKI) Charter. http://www.ietf.org/html.charters/spki-charter.html.

[3] M. Blaze, J. Feigenbaum, and A. D. Keromytis. Keynote: Trust management for public-key infrastructures. In *1998 Security Protocols International Workshop*, vol. 1550 of *LNCS*, pp. 59–63, 1998.

[4] M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized trust management. Research in Security and Privacy, 1996. IEEE Computer Society, Technical Committee on Security and Privacy.

[5] M. Bove. Key management, setup and implementation of an anonymous credential system. Master's thesis, 2001.

[6] J. Camenisch and A. Lysyanskaya. Efficient non-transferable anonymous multi-show credential system with optional anonymity revocation. In *EUROCRYPT 2001*, vol. 2045 of *LNCS*, pp. 93–118. Springer Verlag, 2001.

[7] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–88, Feb. 1981.

[8] D. Chaum. Security without identification: Transaction systems to make big brother obsolete. *Communications of the ACM*, 28(10):1030–1044, 1985.

[9] D. Chaum and J.-H. Evertse. A secure and privacy-protecting protocol for transmitting personal information between organizations. In *CRYPTO '86*, vol. 263 of *LNCS*, pp. 118–167. Springer-Verlag, 1987.

[10] D. Chaum, A. Fiat, and M. Naor. Untraceable electronic cash. In *CRYPTO '88*, vol. 403 of *LNCS*, pp. 319–327. Springer Verlag, 1990.

[11] Consultation Committee. *X.509: The Directory Authentication Framework*. International Telephone and Telegraph, International Telecommunications Union, Geneva, 1989.

[12] I. B. Damgård. Payment systems and credential mechanism with provable security against abuse by individuals. In *CRYPTO '88*, vol. 403 of *LNCS*, pp. 328–335. Springer Verlag, 1990.

[13] C. Dwork, J. Lotspiech, and M. Naor. Digital signets: Self-enforcing protection of digital information. 1996.

[14] T. Eirich. KeyMan. http://www.alphaworks.ibm.com/tech/keyman.

[15] C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, and T. Ylonen. *SPKI Certificate Theory*. Internet Engineering Task Force RFC 2693.

[16] A. Fiat and A. Shamir. How to prove yourself: Practical solution to identification and signature problems. In *CRYPTO '86*, vol. 263 of *LNCS*, pp. 186–194. Springer Verlag, 1987.

[17] O. Goldreich, B. Pfitzman, and R. Rivest. Self-delegation with controlled propagation — or — what if you lose your laptop. In *CRYPTO '98*, vol. 1642 of *LNCS*, pp. 153–168, 1998. Springer Verlag.

[18] D. M. Goldschlag, M. G. Reed, and P. F. Syverson. Onion routing for anonymous and private internet connections. *Communications of the ACM*, 42(2):84–88, February 1999.

[19] A. Herzberg, Y. Mass, J. Mihaeli, D. Naor, and Y. Ravid. Access control meets public key infrastructure, or: Assigning roles to strangers. In *Proceedings of the 2000 IEEE Symposium on Security and Privacy*, pp. 2–14, 2000. IEEE Press.

[20] N. Li, B. Grosof, and J. Feigenbaum. A practically implementable and tractable delegation logic. In *"Proceedings of the 2000 IEEE Symposium on Security and Privacy"*, pp. 27–43, 2000.

[21] N. Li, B. N. Grosof, and J. Feigenbaum. A logic-based knowledge representation for authorization with delegation. In *"Proceedings of the 12th IEEE Computer Security Foundations Workshop"*, 162–174.

[22] N. Li, J. C. Mitchell, and W. H. Winsborough. Design of a role-based trust-management framework. In *"Proceedings of the 2002 IEEE Symposium on Security and Privacy"*, pp. 114 – 130, 2002. IEEE Press.

[23] N. Li, W. H. Winsborough, and J. C. Mitchell. Distributed credential chain discovery in trust management: extended abstract. In *8th ACM CCS*, pp. 156–165. ACM Press, 2001.

[24] A. Lysyanskaya, R. Rivest, A. Sahai, and S. Wolf. Pseudonym systems. In *Selected Areas in Cryptography*, vol. 1758 of *LNCS*, 1999.

[25] R. Mathys. New idemix client handbuch. Technical report, December 2001.

[26] A. Pfitzmann, B. Pfitzmann, and M. Waidner. Isdnmixes: Untraceable communication with very small bandwidth overhead, 1991.

[27] M. K. Reiter and A. D. Rubin. Crowds: anonymity for Web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, 1998.

[28] S. G. Stubblebine, P. F. Syverson, and D. M. Goldschlag. Unlinable serial transactions: Protocols and applications. *ACM Transactions on Information and System Security*, 2(4):354–389, Nov. 1999.