Flexible and Scalable Credential Structures: NetBill Implementation and Experience

Yasushi Kawakura[1] *
TOSHIBA Corporation
Research and Development Center, 1 Komukai Toshiba-Cho, Saiwai-ku, Kawasaki, 210-8582 Japan
+81-44-549-2234(office), +81-44-520-1841(fax), yasushi.kawakura@toshiba.co.jp

Marvin Sirbu
Carnegie Mellon University
Department of Engineering and Public Policy, Baker Hall 129, Pittsburgh, PA, 15213 USA
+1 412 268 3436 (voice) +1 412 268 7196 (fax), sirbu+@cmu.edu

Ian Simpson
Carnegie Mellon University
Department of Engineering and Public Policy, Baker Hall 129, Pittsburgh, PA, 15213 USA
+1 412 268 2670 (voice) +1 412 268 3757 (fax), is2a+@cmu.edu

J. D. Tygar[1]
University of California
Computer Science Division, 621 Soda Hall #1776, Berkeley, CA 94720-1776 USA
+1-510-643-7855 (office), +1-510642-5814 (fax), tygar@cs.berkeley.edu

Abstract

In electronic commerce consumers often need to present attributes such as membership in order to benefit from specific pricing or access. A scalable, efficient mechanism for conveying attributes independently from authentication is required. In this paper we describe a system based on a combination of Public Key Kerberos for Distributed Authentication (PKDA) and attribute credentials as a means for solving meeting these requirements. This system is compared to other proposals for distributed authentication and authorization, and is shown to be superior in several respects. The system has been implemented as part of the NetBill micropayment system and has been demonstrated to work well in meeting the stated requirements.

[1] Work was done at Carnegie Mellon University
* Contact Author

1. Introduction

The Internet is becoming increasingly pervasive in both industry and commerce. A variety of uses, from shopping, to information sharing, require a combination of both authentication and authorization to realize users' objectives. Authorization based on group membership is a frequently encountered requirement. For example, in electronic commerce, membership can imply discounts or special access. In the military, information may be restricted to groups with a "need to know." Mechanisms are required which permit users to demonstrate membership over an unsecured network with flexibility for meeting a wide range of demands and scalability for the rapidly growing community of Internet users. Unfortunately, traditional access control methods [10] are inadequate to the task.

Consider the following physical world scenario. A consumer walks into a car rental agency to hire a car. The agency offers discounts to members of various groups: automobile clubs, professional societies, employees of specific corporations, or even the merchant's own frequent-renters club. The consumer authenticates himself—typically with a picture ID such as a driver's license—and then demonstrates membership in an eligible discount group by showing a membership card with the user's name. The demonstration of membership may be preceded by some form of dialog: the merchant indicates what memberships might lead to discounts, and the consumer presents only one of her many membership cards. Sometimes the consumer will present several cards from her wallet in search of the best discount, but this risks disclosing to the car rental agency too much private information. If car rental agencies or motels discover that providing discounts to automobile club members is necessary to secure their custom it may join with other providers in offering discounts to the club's members. Thus many different service providers or merchants may honor a given membership card.

Our goal has been to–at a minimum–reproduce the flexibility and scalability of this physical world system for the presentation of attributes of a consumer, and translate it to the world of the insecure Internet.

In this paper we focus on authorization in the context of electronic commerce. Memberships are attributes of consumers. Authorization means proving to a merchant that a consumer has a particular attribute. We view electronic commerce pricing as one potential consequence of authorization. Traditionally authorization is viewed as a binary function: access is allowed or denied. But consumers can be authorized to receive special pricing. Thus the output of a pricing decision based on authorization can be: a zero-price or unfettered access; an infinite price or access denied; or any price in between.

In [4] Neuman presents a system of using *restricted proxies* for authorization. A restricted proxy is a ticket giving the party named therein authority to perform certain operations also named in the ticket. The semantics of a restricted proxy assume that the issuer has the authority to compel certain actions from a server. A restricted proxy is a *delegation* of some part of that authority to the party named in the proxy. A proxy is a binding between a named party and an authorization, signed by the issuer.

The NetBill micropayment system uses credentials, which are superficially similar. They bind a named party and an attribute, and are signed by the issuer. However, the interpretation is different. Issuers attest to facts, such as group membership, about subjects named in the credential. Issuers have no authority with respect to merchants who are free to choose what role, if any, these facts should play in a transaction.

In NetBill, authentication is provided in the first instance through Public Key Certificates. These certificates are used to establish Kerberos ticket-based session credentials using a scheme we refer to as Public key based Kerberos for Distributed Authentication or PKDA [3]. In PKDA, by using public key mechanisms and certificates, a client obtains a Kerberos session ticket directly from a server, bypassing the need to contact a centralized KDC [11] or Ticket Granting Service. Symmetric key-based Kerberos tickets provide an efficient

mechanism for repeated authentication throughout a session.

NetBill credentials are certificate-like objects, signed by an appropriate issuing authority, which bind a user's NetBill identity to a particular group membership. The combination of a PKDA session ticket proving identity, and a credential proving membership is sufficient to permit a merchant or verifier to implement appropriate membership-based policies. The use of public key cryptography in the authorization function coupled with PKDA authentication brings the scalability of public key to both of authentication and authorization. NetBill's implementation of credentials, first proposed in [2] is similar to more recent proposals for the use of Attribute Credentials in Transport Layer Security (TLS) described in [8],[9].

In the next section we describe the requirements for an authorization method suitable for use in electronic commerce. Section three overviews several existing authorization methods and compares them, along with the NetBill credential system, against the requirements. In section four we describe in detail the implementation of credentials within the NetBill electronic commerce system. Sections five and six describe several credential applications, lessons learned and our conclusions.

2. Requirements for Authorization in Electronic Commerce

2.1 Basic Model

Figure 1 shows the basic model of authorization. A credential issuer performs two functions: it maintains a database, which associates a member's identity with membership information; and it issues credentials for consumers when requested. The consumer obtains credentials from credential issuers and presents them to merchants to prove membership. The merchant or verifier receives credentials and confirms the presenter's memberships.

We use a common example of digital libraries to explain our model. Consider a commercial digital library that has contracted with a university to provide articles to the university community at a special price. The personnel office for faculty or the registrar for students are the respective authorities for determining membership in the university community. The digital library needs to know whether a consumer is a member of the university community and thus entitled to special pricing. The consumer first obtains a proof of membership from the personnel office or the registrar, and then presents it to the library. Besides the digital library, a textbook distributor such as Varsitybooks.com, or selected e-commerce computer vendors may provide special discounts to members of the university community. The computer vendor may also be a credential issuer if it treats previous purchasers as members of a group entitled to technical support services.

2.2 Characteristics

2.2.1 Separation of authorities

At any time, a consumer may be a member of many different groups or possess many different attributes. Attribute authorizers must be separate from each other, and preferably separate from the certificate authority that vouches for a user's identity. For example membership in the university community is not the same as membership in the group of recent computer purchasers.

There has been some debate in the electronic trust community over the difference between authentication and authorization. Some would argue that a public key web server certificate identifying the holder as ABC, Inc. can be viewed not as performing an *authentication* function, but as *authorizing* the holder of the certificate to do business as ABC, Inc. Similarly, public key certificates used in the SET secure credit card protocol can be seen as authorizing the holder of the corresponding private key to spend on a particular account, as opposed to identifying the card holder.

Under this view, the concept of an authentication certificate is meaningless; all certificates amount to authorization certificates for some particular purpose(s). In the NetBill credential model, an authentication certificate entitles the consumer to use a certain identifier, and a credential certificate binds that identifier to membership in a group. This separation provides a number of benefits. In particular there is no public key in a NetBill

credential, and thus there is no need to maintain a large cache of private keys for use with multiple public key certificates for different purposes.

2.2.2 Scalability in credential issuers

The credential framework needs to scale gracefully as the number of credential issuers increases. Merchants must be able to easily verify credentials from new issuers. Returning to our earlier examples, many universities may contract with the same commercial library or the same textbook vendor. These merchants must be able to verify easily credentials issued by any one of the campuses.

2.2.3 Scalability in verifiers

The credential framework needs to work effectively even as the number of merchants increases. New merchants must be able to begin accepting proofs of membership from many different issuers without significant up front effort.

2.2.4 Multiple memberships

Consumers need to be able to prove multiple attributes to any given merchant. For example, merchant discounts for membership may be additive. Alternatively, the merchant may offer to review multiple attributes and provide the largest discount arising from any one of them.

2.2.5 Selective presentation

Consumers should have full control over which credentials are presented to which merchants. This feature is important to protect the consumer's privacy.

2.2.6 Changing attributes during a session

A consumer should be able to change her attributes during a session without re-establishing the session. This property allows the consumer to select credentials in an interactive way with a merchant. For example, suppose a consumer begins a session with a computer vendor by demonstrating membership in the academic community in order to access the list of university approved systems. Later, in the course of browsing, the consumer learns that the computer vendor provides an upgrade service only for previous buyers. A consumer should be able to easily add a proof of purchase credential during the shopping session.

2.2.7 Efficiency for repeated interactions

For some uses, such as digital libraries, consumers are likely to make many requests during a session. An authorization scheme, by mixing public key and symmetric key mechanisms, should reduce the cryptographic processing burden necessary in the case of repeated interactions.

3. Existing or proposed authorization methods

This section reviews several existing approaches to authorization, in light of the requirements presented above.

3.1 ACL

Authorization can be achieved without credential issuers through the use of access control lists maintained by verifiers or merchants. An ACL is a list or database of identity-attribute pairs. When the consumer contacts the merchant, the merchant authenticates the consumer's identity and looks up the associated attributes in the ACL list. Each merchant is thus responsible for independently maintaining such an ACL list. The ability to prove an attribute to one merchant, through appearance in that merchant's ACL list, provides no help in proving the same attribute to another merchant. Indeed, the concept of ACLs sidesteps the key issue of verifying the identity-attribute association in the first instance. Where the merchant is the original attribute authority—*e.g.* for identifying the consumer as a previous purchaser—there is no difficulty. However, where the attribute authority is unrelated to the merchant, the ACL concept provides no clear mechanism for creating ACL entries in the first instance. Thus the ACL concept does not scale well with multiple issuers.

The ACL concept also presumes that the merchant is fully aware of all the potentially relevant consumer attributes, a privacy problem. The ACL approach is efficient, as no special cryptographic processing is needed after initial authentication.

3.2 X.509v3 integrated certificates

The public key certificate format X.509 version 3 [6] allows the issuer to define arbitrary attribute-

value pairs. It is thus straightforward to include multiple memberships in an integrated identity/authorization certificate, sometimes called a "jumbo" certificate. This approach has a number of drawbacks however.

(1) Where there are separate attribute authorities for each attribute, each of these must communicate securely with the certificate issuer for the issuer to construct a certificate with all applicable attributes.
(2) The expiration date of the certificate must be the minimum of the expiration dates of any of the attributes. Thus, if student registration expires at the end of the term, the certificate must be reissued when the student attribute becomes invalid even if other attributes, such as automobile club member are unchanged. Similarly, revocation of any one of the attributes requires revocation of the certificate with all its attributes.
(3) The system scales poorly with the number of issuers, since each must communicate with the certificate issuer that creates the integrated certificate. Decentralizing certificate issuance through a certificate hierarchy ameliorates this problem.
(4) The system scales readily with the number of verifiers since each can use the certificate hierarchy to independently verify the certificate and its accompanying attributes.
(5) The approach can support multiple attributes. However, it provides no mechanism for selective presentation of attributes to verifiers. Since all attributes are presented at initialization of a session, there is no need or ability to add attributes during a session.
(6) By itself the use of jumbo certificates provide no means for improved efficiency for repeated use. When coupled with a protocol such as Transport Layer Security (TLS) or SSL v3, attributes can be associated at a verifier with the temporary session key, thus simplifying successive interactions.

3.3 SPKI

Work in the Simple Public Key Infrastructure (SPKI) working group has proceeded from the assumption that all certificates are for the purpose of authorization, not authentication. An SPKI certificate binds a public key to an authorization. In effect, the issuer asserts that the holder of the corresponding private key is authorized to take the actions implied by the authorization portion of the certificate. No expression of *identity* need be contained. Thus, a certificate containing a public key and a credit card number, and signed by a card issuing bank, implicitly authorizes the holder of the corresponding private key to make charges to the specified credit card account; the identity in the sense of the name of the card holder, is irrelevant.

The SPKI approach satisfies many of the criteria discussed above. It separates authorization from identity by leaving identity out altogether. It is scalable in the number of issuers, since a certificate hierarchy enables merchants to easily verify the authenticity of certificates from multiple issuers. For the same reason it is similarly scalable in the number of verifiers. Multiple memberships are conveyed through the use of multiple certificates. At the same time, the consumer may selectively choose which certificates to present.

Initial validation of the consumer's attributes is computationally costly, however. The consumer's client must construct a signature using each of the private keys corresponding to the attribute to be proved. Each of these signatures must then be verified using the corresponding SPKI certificates, and their validation chain verified.

To improve efficiency, the SPKI group has discussed the notion of verifier-issued certificates. After verifying N independently issued authorization certificates, the verifier issues a certificate to the customer, signed by the verifier, and encoding all of the authorizations. The customer can use this new certificate in subsequent interactions, thus reducing from N to one the number of certificate verifications. Of course, this requires the generation of a new key pair. Alternatively, the verifier's certificate can be signed with a symmetric key, much like a Kerberos ticket, since only the merchant himself will need to verify it upon re-presentation. To date there are no complete systems implementing this scheme, however.

3.4 SESAME PAC

The SESAME project has proposed extensions to Kerberos including a Privilege Attribute Certificate (PAC) [1],[12]. A consumer authenticates to Kerberos in the usual way, acquiring a Ticket Granting Ticket or TGT. The Privilege Attribute Server (PAS) is assumed to share a long-term key with the KDC. Thus, the TGT can be used as a service ticket to contact the PAS for a proof of access rights in the form of a Privilege Attributes Certificate (PAC), and a new Privilege TGT or PTGT. The PAC is a public-key signed document, and is linked by a shared identifier to the PTGT. The PTGT can be used to acquire session tickets from a TGS in the usual way, with the shared identifier propagating into the service ticket, which thus remains linked to the PAC.

The SESAME scheme provides separate instruments for authentication (Kerberos tickets) and privilege assertion (PACs). However, the two are tightly coupled, as the PAS and the AS are assumed to share a common long-term key used for encrypting the TGT. This centralized model also assumes that many user memberships are known to the centralized PAS which is the only issuer of PACs for each realm (Cross realm techniques allow PACs issued by one realm to be processed by a server in a different realm). Multiple attributes in a PAC are exposed to each server.

Furthermore, the attributes in a PAC cannot be changed during a session. A new PAC, PTGT and session ticket would need to be issued for new attributes to be asserted.

Because the scheme relies largely on Kerberos session tickets, it is quite efficient. Upon the first interaction with a service, a secure session context is created identified by a session key. Subsequent use of the same context implies all of the PAC attributes.

3.5 The flexible distributed authorization

Trostle and Neuman propose in [5] a flexible authorization protocol capable of mimicking the behavior of proxies in OSF DCE 1.0, SESAME PACs, and Neuman's earlier proxy scheme [4]. It relies on PACs which are sealed by symmetric session keys, thus reducing the burden of public key signature verification required in SESAME. Additional optimizations reduce the number of messages required to be exchanged, particularly where capabilities are delegated. Nevertheless, the scheme still depends on a central server PAS capable of signing off on all PACs. Like SESAME, and OSF DCE, this approach scales poorly when multiple independent authorities attest to user attributes. Also, like SESAME, it suffers from including multiple attributes in one PAC, or requiring several steps to reissue a new PAC with a different subset of attributes.

3.6 Attribute certificates over TLS

A proposal to the IETF TLS working group to provide for the presentation of attribute certificates (AC) along with identity certificates [8], [9] is conceptually very similar to the credential scheme implemented in NetBill. Both use a public key certificate to establish identity. Both propose the use of an attribute certificate that binds the identity to an attribute, though in the case of TLS/AC multiple attributes may be included in a single AC. The proposals go on to describe how ACs can be either pushed by a client or requested by a server during session initialization. Presumably, the server software will then associate the TLS session key with the list of attributes, so session key reuse in subsequent interactions implies the same set of attributes.

In TLS/AC authorization and authentication are separated. The system scales will in both issuers and verifiers through the use of a certificate hierarchy. Multiple attributes can be presented, and the client can selectively control presentation. The TLS mechanism for reusing a working key provides an efficient means to reuse both authentication and authorization; although an exchange of at least three packets is needed to synchronize on a cached session key. Any change in credentials, however, requires starting over from the beginning. The only limitations in this approach are those imposed by TLS itself which, unlike PKDA, cannot support UDP, or communication through a proxy. [3]

Table 1. summarizes our comparison of these systems.

4. Implementation of Credentials in NetBill

4.1 Overview

A NetBill purchase using credentials involves four parties as shown in Figure 2: the Consumer (represented by a browser and electronic wallet software—the "MoneyTool"), the Merchant (represented by an HTTP server and a merchant Cash Register server), the NetBill billing server, and one or more Credential servers. The MoneyTool works with the consumer's browser by running as a local proxy on the client's machine. Purchase URLs have the form http://localhost:8887/<merchant_server>/<product_description>. When the user clicks on a purchase URL, it is directed to the MoneyTool which then interposes the NetBill purchase protocol before returning the digital goods to the consumer's browser.

NetBill provides support for three phases of an electronic goods purchase: pricing, goods delivery and payment. The basic elements of the NetBill protocol are described in [2] and are summarized here. We use the following notation: "X ➔ Y" indicates that *X* sends the specified message to *Y*. Here, we use C to indicate the consumer, M to indicate the merchant, and N to indicate the NetBill server.

$[Text]_X$ means *Text* is signed with X's private key.

$\{Text\}_Y$ means *Text* is encrypted using Y's public key.

T_{XY} is a Kerberos ticket proving to *Y* the identity of *X*, and establishing a symmetric session key, *XY*, shared between them.

$E_K(Text)$ is a ciphertext formed by encrypting *Text* with symmetric key *K*.

CC(*Text*) a cryptographic hash of *Text*, such as SHA-1

x-cert is a public key certificate, signed by authority CA and binding the identity *X* to a public key.

The basic NetBill protocol consists of eight steps, which are:

1. C ➔ M Price request
2. M➔ C Price quote
3. C ➔ M Goods request
4. M➔ C Goods encrypted with a key *K*
5. C ➔ M Signed Electronic Payment Order
6. M➔ N Endorsed EPO (including *K*)
7. N ➔ M Signed result (including *K*)
8. M ➔ C Signed result (including *K*)

Credentials can be included in message 1 to enable a merchant to use them in formulating a price quote. Step 3 is acceptance of the price quote by the consumer. Steps 4 through 8 serve to insure that the consumer cannot be charged until after the goods have been successfully delivered to his machine and conversely, that the consumer cannot read the digital goods unless payment was successfully effected at the NetBill server. For further details, see [2]

Each of these messages are encrypted and authenticated under a Kerberos session ticket between the parties. At the first interaction between two parties (*e.g.* M and C) this ticket is issued using PKDA as follows:

C ➔ M	M
M ➔ C	m-cert
C ➔ M	$\{[C, M, Timestamp, K]_C\}^M$
M ➔ C	$T_{CM}\ E_K(C, M, CM, Timestamp)$

First the consumer requests the merchant's public key certificate which is sent in the clear. The consumer then forms the equivalent of a Kerberos TGSREQ message identifying the consumer by her signature and protecting the request and the randomly chosen key *K* by encrypting the message under the merchant's public key. The response, equivalent to a TGSREP message in Kerberos, returns a standard Kerberos session ticket which can be used to identify C to M and a session key *CM* to be used to encrypt correspondence between them. The session key is protected by encryption under *K*.

PKDA provides a highly scalable mechanism for authentication by using public key certificates to

eliminate the need for a centralized KDC.

4.2 The Credential System

4.2.1 Credential form

A credential is a certificate-like object signed by a credential issuer *G* that binds an identity to membership in a group. In NetBill, a credential is constructed as follows:

[*Identity*, *Group*, *Detail*, CC(*CAcct*, *AcctVN*), *Validity*]$_G$

Identity: is the identity of the credential subject. It must be the same as the identity field in the Kerberos ticket used to authenticate the subject's communications with a merchant.
Group: the attribute to be attested to, such as group membership
Detail: Optional additional attribute information
Validity: date-time values indicating the beginning and end of the credential's period of validity.
CAcct: a consumer account number. This enables credential use to be limited to use in conjunction with a specific account. This is described in more detail below.
AcctVN: a nonce

4.2.2 Verification

Credentials are always presented in a message authenticated under a Kerberos ticket. After decrypting the ticket to determine the identity of the sender, two steps are necessary to verify the credential. First the merchant confirms that the identity in the credential is the same as in the ticket. Then the merchant verifies the signature on the credential. Credential issuers are placed in the same hierarchy as certificate authorities, so the verification procedure is the same.

Verifying the public key signatures on a credential's certificate chain takes considerably more time than decrypting a Kerberos ticket. In a micropayment system such as NetBill a consumer is likely to purchase several goods in a session. It is therefore desirable to speed up reverification of credentials. When a merchant has successfully verified a credential, it is entered in a cache and tagged with the minimum expiration time of all the certificates in the chain. If and when the same credential is presented again, the merchant can match the presented credential against the cache entries as a binary string. If the credential matches, and the time is within the validity period of the chain, the credential is accepted. For suitably short cache lifetimes, the risks imposed by not checking for revocation with all issuers in the chain is acceptable. If the PKI distributes revocation lists, these can be used to purge the cache. We call this mechanism a verified credential cache.

The NetBill combination of Kerberos tickets and a verified cache provides for rapid authentication and authorization of repetitive interactions. Moreover, new credentials can be presented in subsequent interactions while still benefiting from the efficiencies of Kerberos authentication.

4.2.3 Issuing credentials

In the NetBill system, a credential issuer maintains a database that associates a user's NetBill identity with some attribute. The database also includes a date after which the association is considered no longer valid and should be reconfirmed. The issuer will then supply a signed NetBill credential to the user's client software upon request. A credential is useless to a third party because it cannot be used without authentication. Nevertheless, we protect requests for credential under a Kerberos session ticket to protect the user's privacy. A credential request/response consists of the following:

1. C ➔ G: T_{CG}, E_{CG}(Group, CAcct)
2. G ➔ C: E_{CG}([C, Group, Detail, CC(CAcct, AcctVN), Validity]$_G$, AcctVN)

The request includes the Group attribute that the user wants verified by the credential issuer. This allows a single credential issuer to authorize more than one type of membership attribute.

We note that the length of a credential's validity can be chosen to be any value less than or equal to the expiration date of the association as maintained in the credential issuer's database. The use of short credential lifetimes is an alternative to distributing a Credential Revocation

List (CRL) to all potential verifiers. For suitably short credential lifetimes, the verifier can be confident that the association attested to by the credential is unlikely to have been revoked since the credential was issued. In NetBill, credentials have a default validity of 24 hours. The burden on the credential issuer to sign new credentials varies directly with the size of the community being authorized and inversely with the credential lifetime. Longer credential lifetimes, while posing greater risks of undetected revocation, ease the burden on credential issuers.

4.2.4 Limiting Credential use by account

Note that, in NetBill, credentials are associated with a single account. Where a user has authority to charge to more than one account—for example a faculty member charging digital library purchases to various research grants—a different credential may be needed for each such account. A credential issuer may wish to limit use of the credential to specific accounts. Thus, the University personnel office may issue membership credentials to faculty only for charges to University accounts, and not for charges to a personal account number.

To protect the consumer's privacy, account information is never disclosed to merchants. Indeed, merchants need only know a user's NetBill identity and that may well be a handle unrelated to the user's legal name.

Enforcement of account restrictions for credentials is accomplished at NetBill. The consumer's client software passes AcctVN and CAcct in a secure portion of the EPO unreadable by the merchant. The merchant includes CC(CAcct,AcctVN) taken from the accepted credential when endorsing the EPO. The NetBill server verifies that these pieces of information match before approving the transaction. In this way, NetBill acts as an agent for the account owner to ensure that credentials can only be used in conjunction with the account indicated in the credential, while still protecting the privacy of account information from the merchant.

4.3 Evaluation against the requirements

(1) Credentials are issued by various credential issuers, while the public key certificates for authentication are issued by a certificate authority. The role of those authorities is separated, though their signature keys are part of the same public key hierarchy.

(2) The scheme scales easily in the number of credential issuers. Because issuers sign with keys that are part of the certificate hierarchy, a merchant can easily verify signatures from new credential issuers.

(3) The scheme also scales with the number of merchants. A single credential issued once can readily be presented and verified by many different merchants.

(4) Consumers can include any number of credentials in a request for quotation thus proving multiple memberships.

(5) Consumers can select which credentials to present individually giving them full control over what is disclosed to a merchant.

(6) Consumers can change attributes during a session by changing the set of credentials to be presented along with the same Kerberos ticket.

(7) The system is efficient for repeat purchases. The merchant need only decrypt the Kerberos ticket to verify the consumer's identity, and lookup the credential in the valid credential cache. New credentials can be presented at any time and require only that the signature on the credential be validated.

4.4 Negotiating credential presentation

Determining which credentials to present to a merchant involves a delicate balance in which the consumer gives up private information in return for some benefit, such as lower prices. Merchants, too, have privacy interests: the rental car agent does not publicly post the list of corporations whose employees are entitled to special discounts. The complexity of these issues is reflected in the controversy being encountered in the WWW Consortium's Platform for Privacy Protection project.[13]. In the NetBill system we opted for simplicity for the user and protection of the user's privacy at the expense of the merchant's privacy interests.

When the consumer makes a purchase from the merchant, the URL describing the good will

indicate whether the merchant solicits credentials for this purchase by including ";CR_P" ("present credentials") at the end of the purchase URL.

If a consumer has credentials from many issuers, this approach begs the question of which credential to present. We allow the merchant to specify a specific attribute and issuer by extending the above information to include them:
";CR_P;ATR=<attributes>,SRV=<cred-issuer>"

However, consider the case of a digital library that maintains site license agreements with many Universities. Listing all the possible credentials that the merchant might accept incurs unnecessary overhead in each purchase URL.

Accordingly, merchants are expected to list at their web site the memberships and credential issuers that they honor. Alternatively, a list could appear on a credential issuer's server indicating which merchants honor the issuer's credentials. Consumers then click on a URL containing information about each credential honored to create an *association* in the consumer's MoneyTool software between the merchant and the credential. This association is stored with the user's MoneyTool profile.

An association URL is of the form:
http://localhost:8887/<merchant_server>/<product_description>;CR_A;SRV=<cred_server>,ATR=<attributes>,MID=<merchant_id>
This URL tells the MoneyTool that the user wishes, upon request from the merchant, <merchant_id>, that the MoneyTool present a credential with attributes <attributes>, available from <cred_server>. The "product" purchased by this URL is typically a free page confirming that the association has been made.

Figure 3 shows the credential management window from the NetBill MoneyTool. For each merchant it lists the credentials (attributes) which will be presented to the merchant, and the location of the credential issuer's server. Consumers can edit these associations at any time.

5. Implementation and Examples

5.1 Using Credentials

An operational prototype of the NetBill electronic commerce system is running at www.netbill.com. The prototype uses a fictitious currency called bibliobucks, and users can open an account and receive BB$ 1,000 to start. Several test "merchants" offer images, text and audio files for purchase. The test merchants use credentials in three different ways:

(1) credentials issued by a merchant providing a discount to consumers who do well on a quiz
(2) a University affiliation credential providing free access to site licensed content
(3) a University affiliation credential that is a prerequisite for buying pay per use content.

Figure 4 summarizes the various elements of the NetBill credential system described in section 4. Referring to the figure, we can walk through an example of credential use.

5.1.1 Proving group membership

Before a credential issuer can issue a credential to a NetBill user, some entity that is authoritative for the attribute in question must enter the association in the credential issuer's database. Consider the case of a merchant acting also as credential issuer to previous purchasers. The merchant can make an entry in its credential issuing database whenever a customer makes a purchase.

To prove University affiliation we rely on an existing authentication mechanism, CMU's Kerberos database, to authorize entries in a CMU NetBill credential database. The CMU credential server also operates a NetBill merchant server. A member of the CMU community fills in a web form with his or her Kerberos identity and password. This form is submitted as a NetBill "purchase request" to the credential issuer. The information is protected in transit by NetBill's use of PKDA. The credential server runs a cgi script that verifies the Kerberos authentication, proving CMU community membership. The purchase request proves the user's NetBill identity. This is sufficient for the credential issuer to enter the relationship <NetBill_identity, CMU_affiliation> in the credentials database. The original Kerberos userID and password are never recorded. This

registration is a one-time task.

More typically, database maintenance is a manual process. A professional society or an automobile club could request a user's NetBill identity on its application forms in order to create or update its credential issuing database as part of application processing.

5.1.2 Creating the association

The consumer's MoneyTool must know to fetch and then present a particular credential to any given merchant. Early in a merchant's web site, the merchant should display a page with association URLs so that consumers can know what credentials they will need. Since this association is stored in the consumer's profile, it needs to be done only once for each merchant.

5.1.3 Shopping

The consumer shops the merchant's web site. When it is time to purchase, the consumer clicks on a purchase URL that includes the *present_credential* flag, CR_P. The consumer's MoneyTool checks its registered associations to determine which credentials to present. It then checks its credential cache to see if it has a valid copy of the credential; if not, it fetches one from the appropriate credential server. It then includes the credential in the request for quote, the first message in the NetBill purchase protocol.

The merchant may use the credential to determine whether the consumer is authorized to purchase this good, or to determine the price. The price is returned in step 2 of the NetBill protocol and the consumer then makes a purchase decision. The MoneyTool can be set to automatically respond to quotations at a zero price; all others require an explicit click to approve.

5.2 Examples

5.2.1NetQuiz

As a means of identifying its more knowledgeable customers, the RFC Store offers the consumer a chance to test their knowledge of networking trivia. The quiz could also be accompanied by a request for additional demographic data. As an inducement to register and take the quiz, the merchant offers 40% off all purchases, for one week only, to those who do well. Success on the quiz causes the consumer to be entered in the "NetQuiz" credential database with an expiration date one week hence**.**

5.2.2 Pittsburgh Post Gazette

The PPG site operated by CMU's library, provides an archive of all stories on a site license basis to CMU affiliates. Presentation of a CMU credential results in a quoted price of zero. For goods priced at zero, steps 6 and 7 of the NetBill protocol can be ignored[2]; the merchant sends the decryption key directly after receiving confirmation of delivery in step 5.

5.2.3 Comic Gallery

The Comics Gallery sells comic strips either by subscription or per strip for BB$.05. However, our current arrangement with the publisher only allows us to provide these strips to CMU affiliates. Thus, the Comics Gallery requires presentation of a CMU credential before quoting a price. In the absence of a credential, the merchant declines to quote a price and the transaction ends.

5.3 Lessons Learned from the NetBill System

While the use of credentials as implemented in NetBill provides an efficient means for conveying group membership to disbursed verifiers, the task of creating the database of <NetBill_identity, attribute> associations remains difficult and often ad hoc. At CMU we were able to automatically create the database entries as needed by using Kerberos authentication to bootstrap the process.[2] The NetQuiz merchant was his own authority for his credential database. For other cases, the problem may be more difficult. A consumer's NetBill identity need have no relation to the identity by which the credential issuer knows an individual. The challenge is to create a reliable means of simultaneously asserting to the

[2] Some users questioned the particular means we used because they had to send their Kerberos password to the credential server. With more effort we might have implemented a scheme in which the consumer acquired a CMU Kerberos ticket for the credential server and this ticket was used in conjunction with the NetBill transaction to simultaneously prove NetBill identity and CMU affiliation.

credential authority both one's NetBill identity and some proof of group membership.

A second issue that emerged is the problem of scaling our concept of association when there are many issuers. For example, some merchant may choose to give discounts to students from any of the 1900+ colleges and universities in the U.S. How can such an intention be expressed efficiently? A merchant might accept a credential bearing the attribute student from any credential issuer for which the American Association of Universities appears higher up in the certificate chain. How can this rule be efficiently conveyed to the consumer? Alternatively, a credential issued by CMU bearing any one of the attributes "student", "faculty", "staff" may be acceptable for some purposes. A mechanism for using wildcards in expressions for credential associations or credential presentation requests might solve these problems, but we have not fleshed out any such scheme.

A third issue is the need for standardization of attributes across issuers. How can the merchant be certain that all issuers use the same criteria for designating someone a "student."

In the current scheme, credentials must be represented with each purchase request. While the use of the valid credentials cache speeds up processing, a high volume merchant would need to maintain a very large cache. Moreover, if successive purchases are directed to different servers in a cluster, either a shared cache would be required, or the credentials would need to be checked multiple times. An alternative approach suggested by Ellison's work[7] would be to present credentials as part of the PKDA initialization. Attributes would then be embedded in the session ticket, and the merchant would need no cache. An advantage of this approach is that such a ticket could be made readable by multiple servers in a cluster if they only shared the ticket-encrypting key. The disadvantage is that a change in the credentials needed for a purchase would require reissuing the PKDA ticket.

6. Conclusion

The NetBill system demonstrates a decentralized system for authentication and authorization based on a combination of PKDA authentication and public key credentials. The system is highly scalable in the number of issuers and verifiers because it is based on public key mechanisms. It uses symmetric Kerberos tickets and caching for efficient operation over repeated interactions. It is independent of transport layer protocols. Finally, it provides the user with fine-grained control over the extent of disclosure of attributes to any one verifier.

Acknowledgements

The NetBill system could not have been implemented without the hard work of the many developers on the NetBill team. This research was sponsored by the Air Force Materiel Command, under Advanced Research Projects Agency contract No. F1962895C0018, ``Electronic Commerce: The NetBill Project." Additional support was received from the National Science Foundation under Cooperative Agreement No. IRI9411299. The views and conclusion contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Advanced Research Projects Agency, the National Science Foundation, or the U.S. Government.

References

[1] Per Kaijser, Tom Parker and Denis Pinkas, "SESAME: The solution to security for open distributed systems", *Computer Communications*, pp. 501-518, **17**, 7, July 1994

[2] Benjamin Cox, J.D. Tygar and Marvin Sirbu, "NetBill Security and Transaction Protocol", in *First USENIX Workshop of Electronic Commerce*, pp. 77-88, 1995

[3] Marvin A. Sirbu and John Chung-I Chuang, "Distributed Authentication in Kerberos Using Public Key Cryptography", *Internet Society 1997 Symposium on Network and Distributed System Security*, Feb. 1997

[4] B. Clifford Neuman, "Proxy-Based Authorization and Accounting for Distributed Systems." *13th International Conference on Distributed Computing Systems*, pp.283-291, May

1993

[5] Jonathan T. Trostle and B. Clifford Neuman, “A Flexible Distributed Authorization Protocol”, *Internet Society 1996 Symposium on Network and Distributed System Security*, pp..43-52, May 1996

[6] Warwick Ford and Michael S. Baum, *Secure Electronic Commerce*, (Prentice Hall, 1997) pp.250-258.

[7] Carl Ellison “Generalized Certificates”, http://www.clark.net/pub/cme/html/cert.html, Sep. 1996

[8] S. Farrell, Internet draft: “An Internet Attribute Certificate Profile for Authorization”, draft-ietf-tls-ac509prof-00.txt, Aug. 1998

[9] S. Farrell, Internet draft: “TLS extensions for Attribute Certificate based authorization”, draft-ietf-tls-attr-cert-01.txt, Aug. 1998

[10] Warwick Ford, *Computer Communications Security*, (Prentice Hall , 1994)

[11] J. Kohl and C. Neuman: RFC1510 “The Kerberos Network Authentication Service (V5)”, Sep. 1993

[12] P.V. McMahon, “SESAME V2 Public Key and Authorization Extensions to Kerberos,” in *Proceedings of the 1995 Symposium on Netwrok and Distributed System Security,* pp. 114-131, Feb., 1995.

[13] Laurie F. Cranor, “Bias and responsibility in ‘neutral' social protocols,” *Computers & Society*, **28**, 3, p. 17-19

	ACL	X.509v3	SPKI	PAC	Trostle& Neuman	TLS/AC	Our approach
Separation of authorities	X	X	O	O	O	O	O
Scalability in credentials issuers	X	X	O	X	X	O	O
Scalability in verifiers	X	O	O	X	X	O	O
Multiple memberships	O	O	X	X	X	O	O
Selective presentation	X	X	O	X	X	O	O
Changing attributes during a session	X	X	X	X	X	X	O
Efficiency for repeated interactions	O	(O)	O	O	O	X	O

(X: impossible or not suitable, O: suitable)

Table 1 Comparison of authorization mechanism

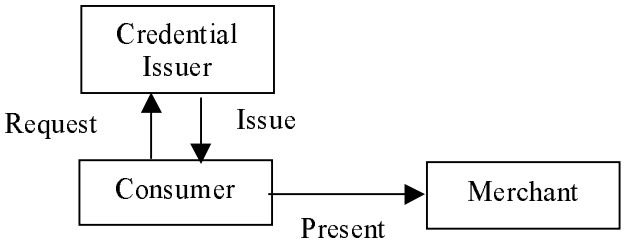

Figure 1. Basic Model of authorization

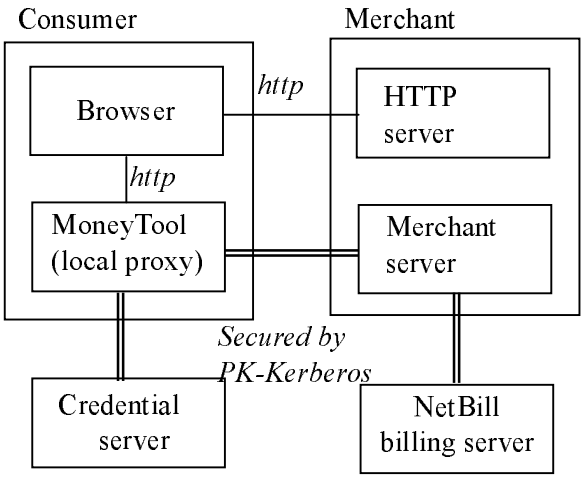

Figure 2. NetBill architecture

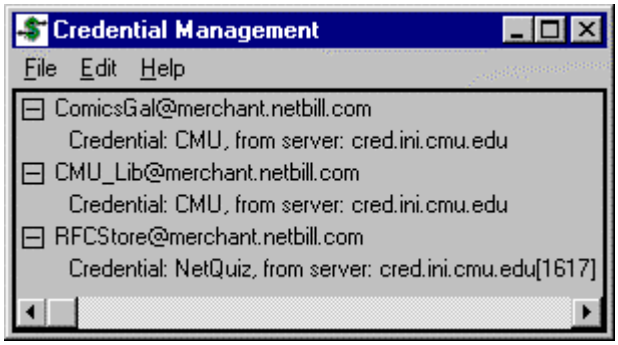

Figure 3. NetBill Credential Management Window

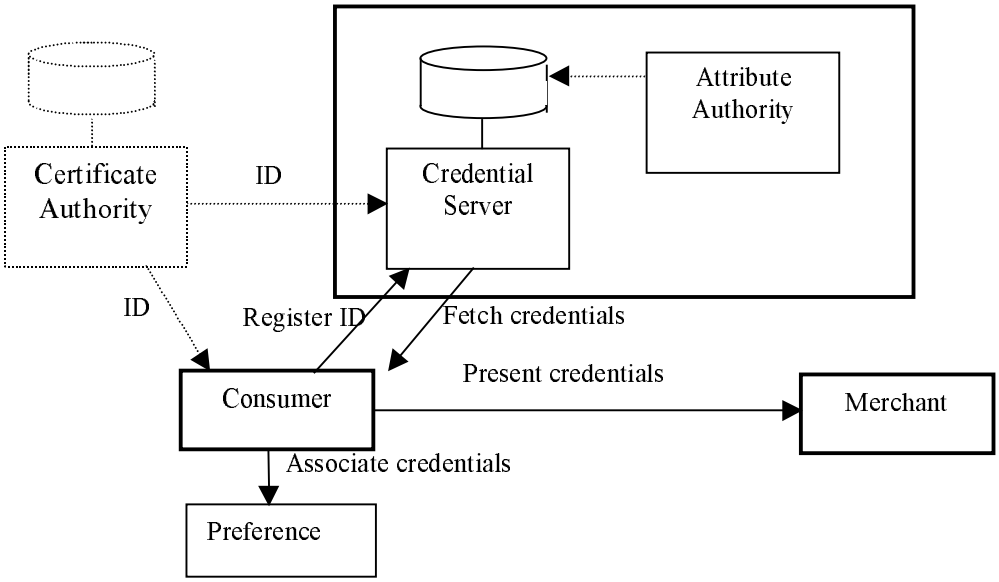

Figure 4. Credentials system architecture