Table of Contents

Entrust's Vision for Privilege Management ............................................................... 2
XML and Authentication Interoperability ................................................................ 4
XML and Authorization ............................................................................................... 6
  Submitting Privilege Information using XML ......................................................... 6
  Publishing Privilege Information Across PMIs ....................................................... 7
  Requesting Authorization Functions from a PMI ..................................................... 8
  Requesting Authorization-Related Information from a PMI ...................................... 10
  Managing the PMI .................................................................................................... 11
  Non-repudiable Auditing of Authorization Activity .................................................. 11
XML and Fine-Grained Document Access Control .................................................... 11
Conclusion ................................................................................................................... 13
Entrust’s XML Strategy for Authorization

The last 24 months have witnessed solid growth in the use of technologies for authenticating identities online, especially using public-key cryptography (PKI). Increasingly, however, authorization capabilities are emerging as the next major security requirement. This is happening because both authentication and authorization (along with other security capabilities such as archive and notarization) are essential components of a complete security infrastructure for e-business. A company not only needs to know that a customer is in fact who they say they are. It also needs to control what resources that customer has access to, what factors affect availability of the resources to that customer, and what auditing is required to ensure the non-repudiation of transactions with that customer. Other factors in the decision making process include policies that are active at the time of the request, and attributes and environmental information that can affect the results of the permission evaluation. Insofar as it must consider all these factors, authorization addresses a larger and more complex problem set than authentication.

Entrust has focused its development efforts on providing a complete integrated security infrastructure for trusted e-business that includes authentication, authorization and a common administration/development framework: a Trust Relationship Management (TRM) infrastructure. That area of the security infrastructure that is focused on authorization we have called Privilege Management Infrastructure (PMI). Entrust’s getAccess™ product provides a state-of-the-art implementation of PMI. Entrust has integrated getAccess with its TruePass and Entrust Direct PKI products to provide the industry-leading solution in Trust Management Relationship. The purpose of this paper is to describe Entrust’s strategy and approach for future enhancement to the TRM infrastructure products.

Entrust is enhancing getAccess to meet the continuing evolution of authorization requirements for e-commerce. One of the most important areas of enhancement is in the use of XML (Extensible Markup Language) within PMI. XML provides a structured way to add context to data so that it can be shared between or among different applications. Where systems of old used ASCII text in batch file transfers, today’s systems will be built to support “transitional datasets” that process data in real time through message queues and application servers. By establishing a common language between participants in a business transaction, XML schemas simplify e-business communication by acting like a “dynamic data dictionary,” passing along data that has or can reference elements and attributes to provide context. XML can be used within PMI to structure authorization-related communication both within the larger security infrastructure for a business and also between businesses participating in an e-business transaction.
Entrust’s Vision for Privilege Management

The move from simply storing and manipulating electronic data to enabling more general e-business (e-government, e-health, and so on), has made access control only a subset of the broader privilege management problem to be solved. Supporting automated electronic workflow, often referred to as “Process Management” (for example: inventory trigger, to supplier purchase order, to warehouse delivery, to accounts payable, to factory receiving bay, to manufacturing floor, to product delivery, to customer billing, to accounts receivable) requires more than access control. It encompasses dynamic evaluation of complex business rules coupled with management of fine-grained entity components and role privileges to initiate certain transactions and processes, to create and sign certain electronic forms and documents, and to allow completion of certain functions and data flows. Real-time, on-line access control to a database or a Web site is clearly still important, but this now represents only a portion of the total e-business requirements for a Privilege Management Infrastructure.

The goal of a PMI is two-fold: first, to ensure that users are allowed to access the resources and do the transactions that they request; and second, to be able to manage and to interchange user privileges across applications, systems, enterprises and communities of security domains.

Figure 1: Privilege Management Infrastructure
Authorization assumes that identity has already been established, such as through checking the integrity, validity, and status of identity certificates. It then carries the security assurance to the next level, confirming that the authenticated user has appropriate privileges to perform the function, execute the transaction or use the resources they are requesting. These privileges include permission to perform the transaction within the specified constraints, such as credit limits that are specific to that user. The PMI also provides privilege information to other components within the environment, so that they can tailor resources or capabilities for a particular user.

In the absence of a pervasive Privilege Management Infrastructure, authorization is typically provided by the applications themselves. However, abstracting the authorization functions into a Privilege Management Infrastructure increases the overall security of a solution and makes it more flexible and scalable. Each applications no longer has to implement authorization decisions, nor ensure that they are complete and consistent with respect to authorization decisions made across the organization. Abstracting authorization information into a common, shared infrastructure also reduces the complexity of managing authorization policies. In these ways, PMI increases the overall security of the solution and decreases the cost of application development and management.

Entrust, therefore, considers authorization as best accomplished by establishing an infrastructure for privilege management. Such an infrastructure must be pervasive throughout the enterprise and larger communities of trust. It must be accessible through well-defined interfaces that support all the technologies participating in the business, including web servers, application servers, XML document servers, commercial off-the-shelf software, and customer software. It must be platform-independent, both in its interfaces and in its components. Finally, it must be easily managed and reliable in its operation, meeting the stringent availability and performance requirements for e-business.

In order to support e-business models like independent trading exchanges, ASPs, and other collaborative communities of interest, the privilege management infrastructure must be extensible across and beyond enterprise boundaries, as shown in Figure 2 below. A single transaction can often be distributed across multiple companies, multiple Web sites, and multiple marketplaces, each of which may have its own authentication and authorization schemes. Companies need a standard open framework that will enable them to securely interoperate across company boundaries and across heterogeneous platforms.

Entrust views XML as an important technology for achieving a pervasive PMI with well-defined interfaces. Each of the sections that follow explore an aspect of the role of XML within PMI, from communicating authenticated identities to encoding privilege information to enabling customization of XML documents on the basis of privilege. This discussion of XML within PMI does not cover the full range of Entrust’s support for and use of XML within security infrastructures; other Entrust capabilities address the requirements for XML signature and encryption, for example. See the Trusted Web Services and XML Security Standards white paper for more information on Entrust capabilities related to XML.
XML and Authentication Interoperability

XML is relevant to PMI not only because of the requirement that the XML-based B2B communications be secured with regard to authorization as well as authentication. The security industry itself, facing many of the same issues that motivate XML for cross-application and enterprise communications, is proposing a variety of XML-based standards to increase interoperability, remove tight inter-system coupling, and improve ease of implementation of security solutions. Entrust is actively involved in these standards, supporting those that are currently under development and leading the establishment of additional standards that will facilitate interoperability.

One important standards effort is the SAML (Security Assertions Markup Language) initiative of the OASIS (Organization for the Advancement of Structured Information Sciences) Security Services Technical Committee. Its goal is to provide a standard way to define user authentication, authorization, entitlements and session information via XML documents. As its name suggests, SAML will allow business entities to make assertions regarding the identity, privileges, and entitlements of one entity (principal) to other entities such as partner companies, other enterprise applications, and so on. These assertions are passed as XML documents.
For example, in Figure 3 below User A has authenticated to a B2B marketplace Exchange A. Part of the transaction that User A wishes to perform requires functionality not found amongst the members of Exchange A, but available from members of Exchange B (quite possible with vertical specific marketplaces). Before Exchange B can act on the behalf of User A and mediate the interaction with an appropriate supplier, it requires proof of the identity of User A. Since Exchange A has already authenticated User A, it provides to Exchange B a SAML assertion to that extent. Now trustful of the identity of User A, Exchange B either provides an appropriate list of its members for User A to choose from or automatically selects one based on the SAML information.

![Figure 3: SAML Name Assertion](image)

These assertions can either be pushed from the Asserting Party to the Relying Party, as in Figure 3, or pulled by the Relying Party from the Asserting Party.

As explained in the next section, privilege-related information such as credit rating could also be propagated as part of this or in a new message from Exchange A to Exchange B. However, the XML-based name assertion itself is of fundamental importance to PMI, because it allows the decoupling of authentication from authorization. In the example above, authentication was performed by Exchange A; the partner exchange trusts the assertion that authentication has been done and uses the asserted identity within its own PMI to make decisions about whether the user, either individually or in terms of the company to which they belong, can perform the work and be given access to the resources they have requested.
This interaction of authentication and authorization through a well-defined XML interface can be used within an enterprise as well as between enterprises. When coupled cryptographically with XML-defined assertions of privilege, SAML assertions of identity provide both the flexibility and the strength of security required for an effective PMI.

**Entrust Direction.** Entrust fully supports the SAML effort. We are currently modifying our getAccess product to be able to create and accept name assertions (both Entrust-generated and other vendors’) as part of HTTP requests. This will enhance the capability for single-signon within and across security domains already available from getAccess with single-signon in environments containing multi-vendor security implementations. Entrust also anticipates generating name assertions within TruePass™.

**XML and Authorization**

In addition to the value of XML for name assertions, Entrust views XML as important to authorization in a number of ways:

- Submitting privilege information as part of a transaction request
- Publishing privilege information across PMIs
- Requesting authorization functions from a PMI
- Requesting authorization-related information from a PMI
- Managing the PMI, particularly in an integrated PKI/PMI environment
- Non-repudiable auditing of authorization activity

In each of these areas, XML provides the means of creating standardized and easily-understood constructs for interaction with the PMI. Entrust is currently enhancing its getAccess product to support these areas as part of our participation in industry initiatives like SAML.

**Submitting Privilege Information using XML**

In a transaction request such as that shown in Figure 3 above, it may be valuable to include not only a name assertion but also privilege information. For example, since Exchange A may know the credit rating for the user it has authenticated, it can provide that information to Exchange B as an Entitlement Assertion. Since Exchange A can attest to the user’s credit rating, Exchange B can trust not only the identity but also the credit worthiness of the user. This is shown in Figure 4 below.
Figure 4: Submitting Entitlement Assertions as Part of a Transaction

Similarly, SAML defines another, more compact entitlement assertion format tentatively called “Authorization Decision Assertion” to communicate the result of an authorization action. In the example above, Exchange A may perform the authorization action; rather than sending the “credit rating” to the Exchange B, it can forward the authorization result, saying that the user has been authorized based on their credit rating. The XML formats of the Entitlement Assertions are being defined as part of the SAML effort.

**Entrust Direction.** Entrust, as an active participant in SAML, is modifying getAccess to be able to accept SAML Entitlement Assertions in HTTP requests, validate that the assertions are signed by an authority that it trusts, and use those assertions in making authorization decisions.

**Publishing Privilege Information Across PMIs**

In order for Entitlement Assertions to be available for submission on a transaction request, the PMI must be able to generate and publish those assertions. In addition, a PMI can be a consumer of an Entitlement Assertion not only at the time of evaluating an authorization request, but also at some prior time as a result of publication of the Entitlement Assertion by another PMI. Both the publishing and consuming aspects of PMI use of an Entitlement Assertion are shown in Figure 5.
Entrust is developing capabilities for creation of SAML-compliant Entitlement Assertions. These Entitlement Assertions can be submitted on an HTTP resource request along with an identity certificate or name assertion. These assertions will be interoperable with other SAML-compliant PMIs.

**Requesting Authorization Functions from a PMI**

The current getAccess product provides an application programming interface (API) that allows custom applications, application servers and other programs to request authentication and authorization. Such an interface is clearly required if it is to be truly a PMI accessible from all environments that require authorization to be performed. The role of such an API is shown in Figure 6.

The request to the PMI could come from a Policy Enforcement Point (PEP). This is a software component or system that oversees the execution of the authorized action within the security domain according to the applicable security policies. Such a PEP can reside in an application server (via a Service Provider Interface module); in an application program, in a commercial off-the-shelf packaged application, or in a Web server.
For example, an application server such as BEA’s WebLogic™ Server or IBM’s WebSphere™ Server intercepts requests for a Java EJB running in the container in order to determine whether the requester is authorized for the requested component. The application server acts as a Policy Enforcement Point, which can then call into the getAccess PMI to request an authorization check from the getAccess authorization policy server. The custom security realm that getAccess provides for Weblogic works in exactly this way.

In this scenario, getAccess is the Policy Decision Point (PDP) that makes the authorization decision requested by the WebLogic Server. The PDP, or policy engine, is the heart of the authorization system. It applies the information received from the application server via the API to the policy or policies defined for the requested resource, transaction, or component. Using the authorization rules expressed in those policies, the policy server determines whether the user should be allowed to invoke the requested EJB, execute the requested transaction, use the requested resources, and so on.

**Entrust Direction.** getAccess already provides a flexible request/response interface for authorization. However as with name assertions and entitlement assertions, a standard XML-based API for authorization requests has significant benefits in achieving interoperability across PMIs. Entrust is therefore supporting the SAML effort to define and implement an authorization request/response API.
Requesting Authorization-Related Information from a PMI

In addition to calling into the PMI in order to request an authorization decision, applications may also need access to information maintained in the PMI in order to make fine-grained personalization or customization decisions. Such information may be provided in entitlement assertions (as described above in Figure 5). However, it may also be necessary to provide a request/response API for such queries, as shown in Figure 7.

![Diagram of PMI Query Interface Using XML]

For example, the PMI can store information about global preferences (such as locale) applicable to multiple applications. Although this information could be published in an entitlement assertion, it may be more appropriate to allow an application to query the PMI for this information when it receives a request from a particular user. Proving an API for such queries provides more flexibility to the application developer in designing solutions that integrate with the PMI.

**Entrust Direction.** Entrust getAccess has open Java APIs that allow application access to its registry. Although there are several models which could form the basis of a standard XML-based interface to authorization-related information, such as Microsoft’s XML interface for SQL Server 2000, no standard has emerged yet.
Managing the PMI

In addition to interface for requesting authorization and for requesting authorization-related information, the PMI can benefit from an XML interface for management functions. This is particularly true for accessing information needed for monitoring or auditing information, since an XML interface for these management purposes could simplify integration of the PMI with systems management tools.

Entrust Direction. An XML management interface to getAccess is part of the longer-term strategy for Entrust PMI.

Non-repudiable Auditing of Authorization Activity

Most real world e-business scenarios require that a PMI provide complete and accurate audit of authorization activity. For some situations, such as high-value, complex transactions, it may be necessary to have non-repudiable receipts not only of the transaction, but also of contributing activities, such as the authorization decision.

Entrust Direction. XML is an ideal language for creating these receipts, digitally signing them and submitting them to a notarization service to enhance their non-repudiability. Several vendors, such as Valicert, provide capabilities for creating, storing, and managing non-repudiable records. Although Entrust’s getAccess does not currently take advantage of tools such as these, an XML-based Receipt and Notarization Service that will leverage Entrust’ capability in XML encryption and digital signature is part of the longer-term strategy for Entrust PMI

XML and Fine-Grained Document Access Control

XML Access Control Markup Language (XACML) is a proposal for providing XML documents with a fine-grained access control model and access control specification language. Similar to existing policy languages, XACML is a language oriented around objects, subjects, and actions. Subjects include user IDs, groups and role names. Objects can be defined as finely as a single element within an XML document. Actions consist of read, write, create, and delete.

The application of XACML to an XML document is shown in Figure 8 below. The sequence begins with a user requesting access to an XML document. The XML server handling that document recognizes that the document has authorization information defined for it, and invokes the XACML processor (XACP). The XACP uses information about user privilege, authorization policies, environment variables and document sensitivities to determine which sections of the document should be made available to the requester. In this example, the XACML processor determines that the <B> element should not be made available to this user. It transforms the document by removing the <B> element, updates the status log, and returns the transformed document to the XML server, which delivers it to the user.
The XACML processor is part of the PMI, and it is using the same authorization capabilities as those described earlier in this white paper. However, it is employed by an XML document server, rather than by Web servers, application servers, or applications. That is, XACML is used to authorize access to tagged sections of an XML document based on user privileges and business rules. For example, all doctors involved in a particular patient’s treatment may require read access to the patient's history, but only a Primary Care Physician can change the prescribed medication, or create an entry for a new treatment. XACML embeds the confidentiality decisions in the document, extending the role of the PMI to ensuring end-to-end privacy: from creation of the information through all its transformations to storage and archiving.

**Entrust Direction.** Entrust is actively involved in the creation of the XACML standard. A preliminary version of XACML is available from Entrust. Integration of XACML with other aspects of PMI is part of the long-term strategy for Entrust PMI.
Conclusion

Entrust’s Trust Relationship Management infrastructure enables businesses to enhance their existing relationship by enabling a higher level of trust required for on-line transactions. A critical component of TRM is the ability to specify the authorization policies, rules and entitlements that govern access to critical information. Only when this is possible will businesses be able to ensure end-to-end privacy of their customers, partners or employees information.

This paper has outlined the critical components that will enable the Entrust TRM infrastructure to provide the authorization capabilities in an e-business infrastructure. However, Entrust recognizes that we cannot do it alone. Therefore, we are creating partnerships with key infrastructure vendors that provide the transaction execution or process management products businesses use to run their e-business applications. We also recognize that interoperability is an important requirement to enable cross application domain context sharing and are therefore creating partnerships with other PMI vendors to conduct interoperability testing on PMI solutions. Finally, we continue to work with our customers to ensure that the technology being developed as part of our TRM infrastructure will meet their future requirements for adding trust into e-business relationships.