IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITJ2733

Security System for Distributed Business Applications

Thomas Schmidt, Vienna University of Technology, Austria
Gerald Wippel, Vienna University of Technology, Austria
Klaus Glanzer, Vienna University of Technology, Austria
Karl Fürst, Vienna University of Technology, Austria

ABSTRACT

Internet-focused application components of cooperating enterprises need comprehensive security technologies that go far beyond simple Internet authentication and authorization mechanisms. Basically, authentication is the process of determining the identity of a user or system, whereas authorization is the process of specifying who is allowed to access which resources. XML-based Web services is an upcoming and very promising technology. It enables the communication among Internet application components regardless of their implementation language. A major drawback of existing Web service approaches is the missing security conventions. Therefore, we concentrated all our effort on developing a holistic extended enterprise authentication and authorization system to facilitate agile and secure enterprise-spanning business processes with Web service-enabled application components.

Keywords: *authentication; authorization; business-to-business; enterprise integration; extended enterprise; Internet application; SAML; security; SOAP; Web services*

INTRODUCTION

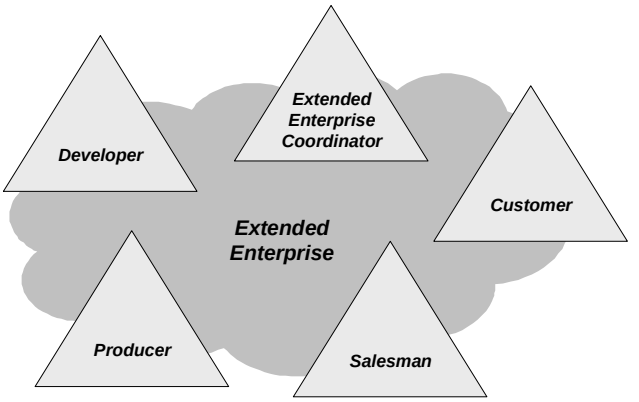
Collaboration between independent enterprises with different core competencies is a very important way to seize opportunities very fast in agile business environments (Goldman, 1991; Camarinha-Matos, 1999). To distinguish the approach described in the following from the general term "business-to-business," the more specific term "extended enterprise" (*Figure 1*) will be used.

The following definition for extended enterprises is new but influenced by other related definitions and approaches: extended enterprises (EE) are temporary or permanent networks of independent enterprises including a coordinator cooperating with the aim to design, manufacture, and sell a product or service in a project-oriented way independent of enterprise borderlines, automated inter-enterprise communication through the usage of information technology, and communicating via Internet-related technologies like XML-based Web services.

The extended enterprise coordinator mentioned above is often described as an enterprise, which extends its boundaries to incorporate business partners. This coor-

This paper appears in the journal *International Journal of Web Services Research* edited by Liang-Jie Zhang.

Figure 1. Principle of extended enterprises



dinator has to provide specific resources for operating the extended enterprise. As a matter of course, the same enterprise can be involved in various extended enterprises. Thus it may happen that two enterprises cooperate in one business segment and compete in other business segments at the same time.

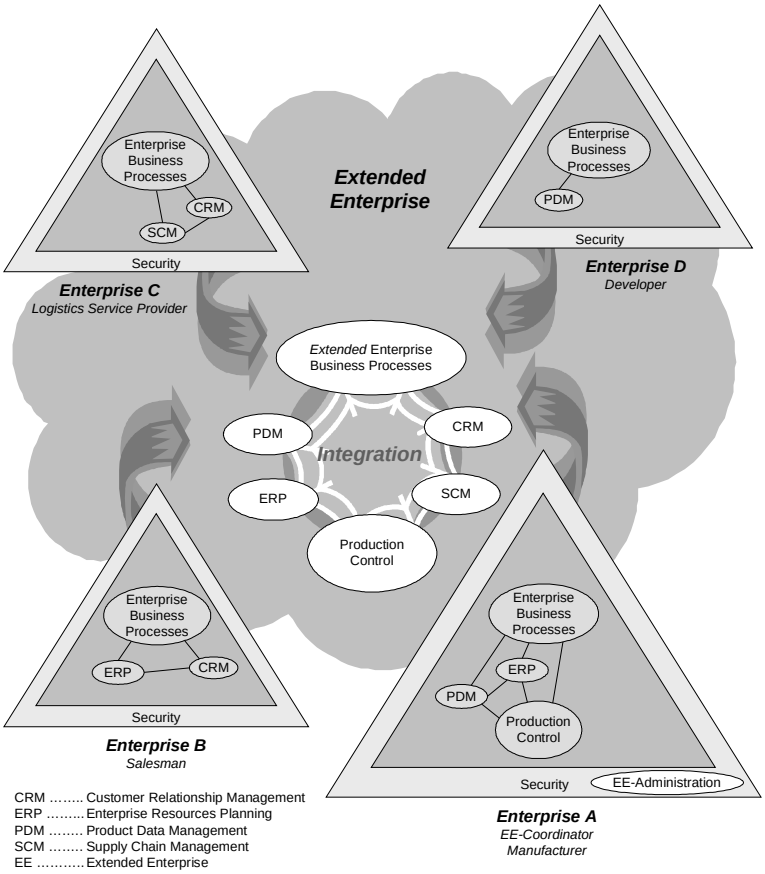
INTEGRATION CONCEPT FOR EXTENDED ENTERPRISES

The common goal of the extended enterprise members is the support of the whole product life cycle composed of product design and development, process planning, logistics, production, marketing and sales, and etcetera. There is a need for several information systems to automate the particular life cycle steps. Different from isolated enterprises, the supporting information system functionalities are widespread in all participating enterprises. Therefore, a very extensive horizontal integration and collaboration between these enterprises is necessary (Zhang, 2003) so that each enterprise is able to contribute their part to the information systems inside the extended enterprise (*Figure 2*). The business value from such collaboration extends across the entire concept development, design planning, engineering, and supplier relationship management (Sayah, 2003).

All business processes that support the product life cycle — summarized into the term "extended enterprise business processes" — interact with the enterprise business processes of the respective company. One precondition for such a closely horizontal integration is the existence of a vertical integration inside the company to provide all internal business processes and business system functionalities to the business partners.

A very important ingredient for a successful software system enabling extended enterprises is security, particularly if the Internet is used as the communication technology. In *Figure 2* each participating company is enveloped with a security layer to guarantee a secure information exchange, authorization, authentication, and other security mechanisms. One enterprise, in the

Figure 2. Extended enterprise integration concept



majority of cases the extended enterprise coordinator, has to do the extended enterprise administration including choosing of the business partners and assigning partners to projects and roles.

Due to the fact that security issues are manifold, in this paper the main focus lies on authorization and authentication measures for extended enterprises. The major issue here is to assure that only authorized people or systems have access to extended enterprise resources.

PRESENT SECURITY APPROACHES

Role-Based Access Control

The concept of role-based access control (RBAC) (Sandhu, 1999; Ferraiolo, 2001) is a suitable mean to simplify administration of privileges. The core idea here is not to grant a specific user privileges to access resources but to introduce an abstraction layer: the roles (e.g., the role

"salesman"). Privileges are assigned to each role (e.g., "salesman" has access to a customer database). Roles are assigned to users (e.g., the role "salesman" is assigned to the user "Bob"). This enables a simplified administration of roles because there are three separate activities, which can be carried out by different people (or roles): assigning roles to roles to establish a hierarchy, assigning privileges to roles, and assigning users to roles.

The RBAC approach has been widely accepted and implemented for enterprise wide access control. First research has been conducted in the field of distributed role-based access control (Sandhu, 2000; Johnston, 1998) to delegate administration to the departments of one enterprise. But, so far, there has been no holistic approach, which provides a solution for extended enterprises.

Public Key Infrastructure and Privilege Management Infrastructure

The public key infrastructure (PKI) approach is often used to establish security in a distributed environment. Basically, both a public and a private key are used. The private key is kept secret by an entity (e.g., a person or an enterprise). The public key has to be distributed. Another central element of PKI is the certification authority (CA), which can certify (like a passport office) the identity of a specific user or entity. This certificate is similar to a passport in the real world. By using the public key infrastructure approach, a trust network can be built. A very important PKI-concept is the certification path. If someone has a certificate from a CA, which is trusted by your own CA, then you can trust this certificate.

The privilege management infrastructure (PMI) is used for authorization. It uses so-called attribute certificates and enables privileges, roles, and delegation mechanisms. Similarly to the PKI CA, an attribute authority (AA) is defined. The attribute authority and certification authority are logically (and, in many cases, physically) completely independent from each other. The creation and maintenance of "identity" can (and often should) be separated from the PMI. Thus the entire PKI, including CAs, may exist and operate prior to the establishment of the PMI.

DRBAC-EE APPROACH

In addition to the RBAC approach, the new approach of distributed role-based access control for extended enterprises (DRBAC-EE) introduces a supplementary abstraction layer, the extended enterprise roles (*Figure 3*). Legally defined roles are a feature of the legal security framework and are defined contractually. In addition, the property if a role is a must-have or a can-have role is also contractually defined. An enterprise that collaborates in an extended enterprise within the scope of a specific project has to provide must-have roles. Additionally, the extended enterprise administrator assigns the can-have roles. To give an example: the roles "member of the advisory board" and "member of the steering committee" are must-have roles. The roles "software developer" and "marketing manager" are can-have roles.

The extended enterprise project administrator centrally controls projects in the security domain (in both the legal and technical sense), including assigning can-have roles to the enterprises. The legal administrator adds and removes participating enterprises.

Figure 3. Role mapping within extended enterprises

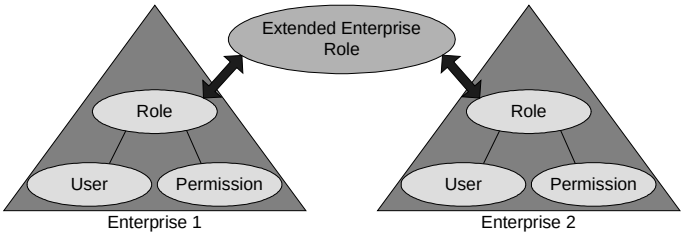


Figure 4 shows two extended enterprises, each with a single leader (enterprises 1 and 5) that controls resources and operations. It is extremely important to have an automated translation between internal and extended enterprise ontologies. This is especially true for enterprise 4, which has to deal with three different ontologies: its own and the two of the different extended enterprises. In practice, we can observe this problem with component suppliers in the automobile industry for example. They have to deal with different ontologies of their big customers. An automated translation from one "world" into the other is a must. A heterogeneous but coherent business space (extended enterprise A and extended enterprise B in *Figure 4* has to be established with each partner of an extended enterprise.

Depending on the complexity of the service, we have developed two different approaches for extended enterprise authorization and authentication: indirect and direct access. The indirect access approach adds an abstraction layer and can deal with ontology translation problems more effectively, whereas the direct access approach can be more efficient and is used mainly to access simple services.

Figure 5 shows the collaboration diagram for the indirect approach, where user X of enterprise A wants to access a resource of enterprise B. A central component is the "access management." The main responsibilities are privilege verifica-

Figure 4. Extended enterprise administration

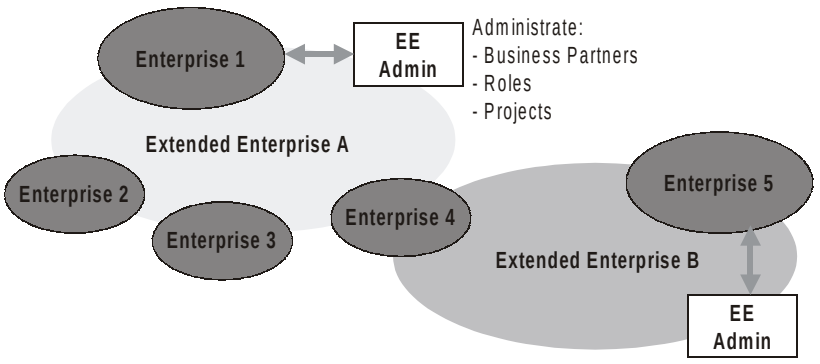
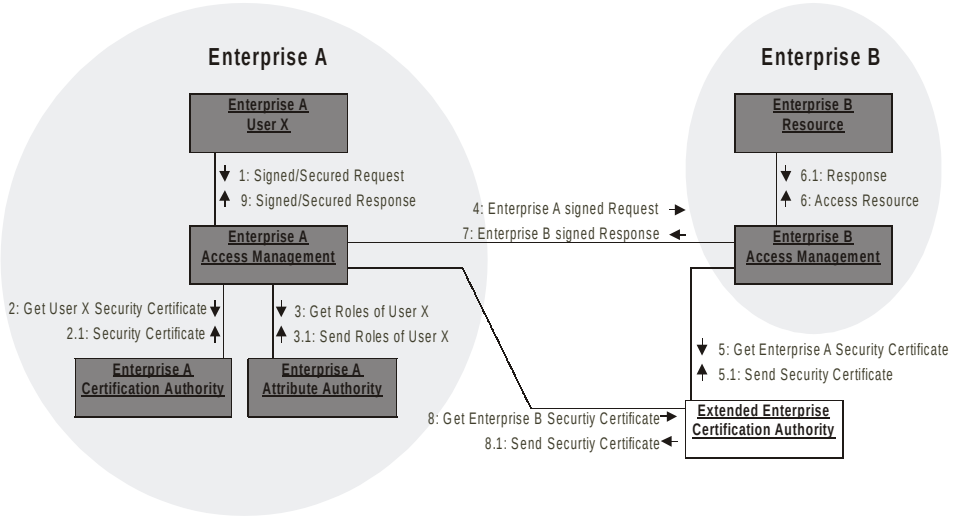


Figure 5. Managing indirect access with DRBAC-EE



tion for internal and extended enterprise roles, ontology translation of internal to extended enterprise roles and processes, and logging for documentation, data mining, and legal actions.

Having gained access (*Figure 5*), all communication is done using the access management component. The collaboration diagram shows a request submitted to the access management component by user X (1). This request can be signed (e.g., using XML digital signature) and/or secured (e.g., using the SSL protocol). The access management component uses a security certificate to check this request (2 and 2.1). The attribute authority of enterprise A is accessed to realize the translation between the user X and his extended enterprise roles (3 and 3.1). Subsequently, the enterprise A access management component signs the request with the signature of enterprise A and sends it with the complete role information to the access management component of enterprise B (4). The roles in this request are the extended enterprise roles and are therefore valid only in the context of this specific extended enterprise. The certification authority of the extended enterprise is accessed to check the signature of the request (5 and 5.1). The request is logged for documentation, billing and data mining. After that, the resource is accessed (6 and 6.1) and the response is signed and sent back to enterprise A (7). This response is also checked (8 and 8.1) and transferred to user X (9). The response to the user could also be signed and/or secured to assure security (certificate check by the user is not shown here).

IMPLEMENTATION

The implementation of the DRBAC-EE approach is based on the Service Oriented Architecture (SOA) and consists of several services as mentioned in the above collaboration diagram. The authentication service of this approach uses an attribute-based authentication with a privilege man-

agement infrastructure. In contrast to X.509 (ITU-T, 2000), XML is used to handle the data needed. The necessary attributes are included in the header of the SOAP message:

- extended enterprise
- enterprises that will act as business partners
- projects to be carried out in the scope of the extended enterprise, and
- legally defined, shared, and project-specific roles.

The SOAP envelope is enriched with additional headers based on the Security Assertions Markup Language (SAML) (SAML, 2002) and Web Service Security (WSS) (WSS, 2002) specification.

SAML, an open standard of Organization for Advancement of Structured Information Standards (OASIS), is an XML-based specification for exchanging security information. One major design goal for SAML was Single Sign-On (SSO), the ability of a user or service to authenticate in one domain and use resources in other domains without re-authenticating. SAML defines XML-encoded security assertions, an XML-encoded request/response protocol, and rules for using assertions with standard transport and messaging frameworks. An assertion is a declaration of fact according to someone.

WSS specification, originally developed by IBM, Microsoft and VeriSign, is now a public draft standard of OASIS that describes enhancements to the SOAP messaging to provide quality of protection through message integrity and single message authentication. The goal of the specification is to enable applications to conduct secure SOAP message exchanges. The WSS specification provides three main mechanisms: security token propagation, message integrity, and message confidentiality. Among other things the WSS specification specifies an extension mechanism of the SOAP header with SAML security token, which is used by the DRBAC-EE approach.

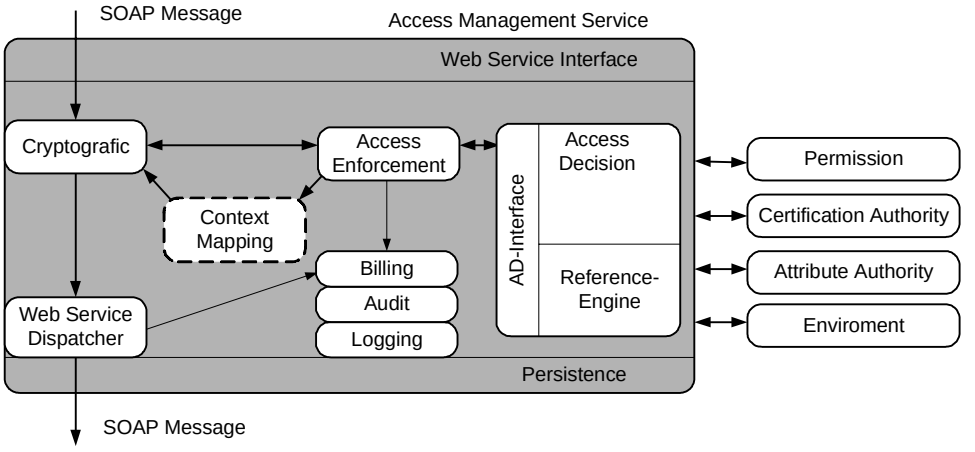
The central access management service of this approach is mainly responsible for authorization of subjects and for routing of messages. As the authentication service it uses SAML to wrap security attributes. Furthermore, the service utilizes XACML (eXtensible Access Control Markup Language) for the authorization check and for expressing security policies (XACML, 2003).

XACML is an XML-based language for access control that has been standardized within OASIS. It provides an access control model and it is a common language to express and to enforce security policies in a variety of environments. XACML allows primarily developers to describe access control policies for XML objects, but also other objects like Web services or files.

The access management service consists of the following sub-services (see *Figure 6*):

- The cryptographic service is responsible for encryption of messages and for the generation and verification of signatures attached to messages. The used APIs, which are based on XML Encryption and XML Signature specification, enable the encryption of total or several parts of messages, offering a secure and unbroken security channel between different services. This service uses the certification authority service to fulfill its task.
- The access enforcement service enforces an access decision from the access decision service and permits or de-

Figure 6. Basic services of the access management service



nies access to the service based on returned results. To fulfill this task, it extracts the needed information from the incoming message or/and from the attribute authority. Attributes concerning the requestor (e.g., user, service) are wrapped by a SAML structure in the header of the message and attributes concerning service operation are direct extracted from the body of the message. The access enforcement service can also be part of the target service to restrict access.

- The access decision service makes the access decision based on the obtained attributes from the message and the permissions respectively rules stored in the XACML policies. The internal reference engine of this service uses XACML policies for the fine-grained access protection of services and resources. At present these policies are either XML files or build up from a database. This service offers a Web service and Java interface based on the ISO/IEC 10181-3 standard. To support the direct approach as mentioned above an external party can send SAML requests to this XACML-based service to enforce access decision. For the decision process the access decision service uses the attribute authority, the certification authority, the permission and rule store, and the environment.
- The context mapping service is part of the attribute authority service and provides the proper mapping of the internal enterprise attributes to the external extended enterprise attributes and vice versa.
- The dispatcher service routes received message to the target service. The routing information are extracted from the received message header and/or requested from a local or global directory services (e.g., UDDI).
- The billing, audit and logging services are collecting data for billing, data mining,

Figure 7. Added information in the header of the SOAP request

```
<soap-env:Header>
    <wsse:Security >
        <saml:Assertion MajorVersion="1" MinorVersion="0"
                AssertionID="2sxJu9g/vvLG9sAN9bKp/8q0NKU="
                Issuer="www.floci-ee.com" IssueInstant="2003-02-14T16:58:33.173Z">
            <saml:Conditions NotBefore="2003-02-14T16:53:33.173Z" NotOnOrAfter="2003-02-15T16:53:33.173Z"/>
            <saml:AuthenticationStatement
                    AuthenticationMethod="urn:oasis:names:tc:SAML:1.0:am:X509-PKI"
                    AuthenticationInstant="2003-02-14T16:53:33.173Z">
                <saml:Subject>
                    <saml:NameIdentifier NameQualifier="www.floci-ee.com" Format="">
                        EELogisticManager</saml:NameIdentifier>
                    <saml:SubjectConfirmation>
                        <saml:ConfirmationMethod>
                            urn:oasis:names:tc:SAML:1.0:cm:holder-of-key</saml:ConfirmationMethod>
                        <ds:KeyInfo><ds:X509Data>…</ds:X509Data></ds:KeyInfo>
                    </saml:SubjectConfirmation>
                </saml:Subject>
            </saml:AuthenticationStatement>
            <saml:AttributeStatement>
                <saml:Subject>
                    <saml:NameIdentifier NameQualifier="www.floci-ee.com" Format="">
                            EELogisticManager</saml:NameIdentifier>
                    <saml:SubjectConfirmation>
                        <saml:ConfirmationMethod>
                            urn:oasis:names:tc:SAML:1.0:cm:holder-of-key</saml:ConfirmationMethod>
                        <ds:KeyInfo><ds:X509Data>…</ds:X509Data></ds:KeyInfo>
                    </saml:SubjectConfirmation>
                </saml:Subject>
                <saml:Attribute AttributeName="ExternalContext"
                            AttributeNamespace="http://www.floci-ee.com/drbac-ee/2003-01">
                    <saml:AttributeValue>
                        <f:RequestAttributes>
                            <f:User>
                                <f:Id>312983</f:Id><f:Name>MrSample</f:Name></f:User>
                            <f:ExtendedEnterprise>
                                <f:Id>23</f:Id><f:Name>EE2</f:Name></f:ExtendedEnterprise>
                            <f:Service>
                                <f:WsdlUrl>http://host/wsdl/Timetable.wsdl</f:WsdlUrl>
                                <f:URL/>
                                <f:EndPoint>http://host:8080/TasksWS/jaxrpc/TimetableIF</f:EndPoint>
                            </f:Service>
                            <f:Enterprise>
                                <f:Id>234876</f:Id><f:Name>ACIN</f:Name></f:Enterprise>
                            <f:EERolesToEEProjects>
                                <f:EERolesToEEProject>
                                    <f:EEProject>
                                        <f:Id>127746</f:Id><f:Name>Project</f:Name></f:EEProject>
                                    <f:EERoles>
                                        <f:EERole>
                                            <f:Id>154</f:Id><f:Name>EEManager</f:Name></f:EERole>
                                    </f:EERoles>
                                </f:EERolesToEEProject>
                            </f:EERolesToEEProjects>
                        </f:RequestAttributes>
                    </saml:AttributeValue>
                </saml:Attribute>
            </saml:AttributeStatement>
            <ds:Signature>…</ds:Signature>
        </saml:Assertion>
        <ds:Signature>…</ds:Signature>
    </wsse:Security>
</soap-env:Header>
```

documentation and maintenance of the system.

- A SOAP header enriched with additional security attributes needed for the authorization is shown in *Figure 7*. Like the SOAP body, this header has to be signed for security reasons (not shown here).

The <wsse:Security> SOAP header block of the WSS specification provides the mechanism for attaching security-related information targeted at a specific recipient. SAML assertion element <saml:Assertion> are attached to the SOAP message using WSS by placing assertion elements inside the <wsse:Security> header. Attributes of the <saml:Assertion> element specify the version of SAML, id of the assertion, issuer of the assertion, time of the instantiation. Sub-Element

<saml:Conditions> specifies the time span at which the assertions are valid.

The element <saml:Authentication Statement> declares a statement that its subject, specified using sub-element <saml:Subject> was authenticated by particular means at a particular time using its attributes AuthenticationMethod and AuthenticationInstant. The example above specifies authentication by means of X.509 PKI. Sub-element <saml:NameIdentifier> of element <saml:Subject> declares the name and the domain of the subject. In our case the name of extended enterprise roles are declared. Sub-element <saml:Subject Confirmation> of element <saml:Subject> specifies the protocol to be used to authenticate the subject and additional authentication information using the sub-elements <saml:ConfirmationMethode> and <saml:SubjectConfirmationData>.

The <saml:AttributeStatement> associates the specified subjects using sub-element <saml:Subject> with the specified attributes using the element <saml:Attribute>. The element <saml:Subject> identifies a subject using element <saml:Name Identifier> specifying role name and an element <saml:SubjectConfirmation> supplying data that allows secure authentication of the subject. The example above uses a signature for the confirmation.

The element <saml:Attribute> and its sub-element <saml:AttributeValue> specify all attributes names and values like extended enterprise, enterprise, extended enterprise roles and projects required by the target service.

The last element <ds:Signature> in the SOAP Header contains a signature with a reference to a SAML assertion.

CONCLUSION

The ability to create agile business value networks from independent enterprises in an effortless and secure way will be the crucial point of success for next generation business-to-business software. Especially the establishment and maintenance of security are extremely important requirements of the industry for agile business value networks.

Within the scope of an international research and development project, design and implementation of a holistic security system for the complex extended enterprise environment have been done successfully. Considering as example business case, a proof-of-concept prototype has been realized in close collaboration with the industrial end-users within the project consortium.

ACKNOWLEDGMENTS

This work was partly supported by the European Commission within the scope of the Growth project FLoCI-EE, G1RD-CT-2000-00324.

REFERENCES

Camarinha-Matos, L.M., & Afsarmanesh, M. (eds.). (1999). *Infrastructures for virtual enterprises*. Boston: Kluwer Academic Publishers.

Ferraiolo, D.F., Sandhu, R.S., Gavrila, S., Kuhn, D.R., & Chandramouli, R. (2001). Proposed NIST standard for role-based access control. *ACM Transactions on Information and System Security*, 4 (3), 224-274.

Goldman, S., Preiss, K., Nagel, R., & Dove, R. (1991). *21st century manufacturing enterprise strategy, Infrastructure.*

(Vol. 2). Bethlehem, PA: Iacocca Institute, Lehigh University.

ITU-T. (2000, March). *ITU-T Recommendation X.509: Information technology. Open systems interconnection. The Directory: Public-key and attribute certificate frameworks*. *ITU: International Telecommunication Union*.

Johnston, W., Mudumbai, S., & Thompson, M. (1998). Authorization and attribute certificates for widely distributed access control. In *Seventh International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*.

SAML (Security Assertions Markup Language) 1.0. (2002, May 31). Committee Specification OASIS (Organization for Advancement of Structured Information Standards). Retrieved from *www.oasis-open.org/committees/security/*

Sandhu, R.S., & Coyne, E.J. (1999, February). Role-based access control models. *IEEE Computer*, 38-47.

Sandhu, R.S. (2000). Engineering authority and trust in cyberspace: The om-am and rbac way. In *Proceedings of the Fifth ACM Workshop on Role-Based Access Control* (pp. 111-119).

Sayah, J., & Zhang, L.J. (2003, September 30). *On-demand business collaboration enablement with Web services*. IBM Research Report, No. RC22926 (W0309-191). IBM T.J. Watson Research Center.

WSS (Web Service Security) 0.9. (2002, January 26). Working Draft OASIS (Organization for Advancement of Structured Information Standards). Retrieved from *www.oasis-open.org/committees/wss/*

XACML (eXtensible Access Control Markup Language) 1.0. (2003, September). Committee Specification OASIS (Organization for Advancement of Structured Information Standards). Retrieved from *www.oasi-sopen.org/committees/tc_home.php?wg_abbrev=xacml*

Zhang, L.J., & Jeckle, M. (2003). The next big thing: Web services collaboration. In *Proceedings of the 2003 International Conference on Web Services Europe (ICWS-Europe03)* (pp. 1-10). Spring-Bergle, LNCS 2853.

Thomas Schmidt is currently working as a research assistant at the Automation Control Institute in the Vienna University of Technology as the head of the working group "Intelligent Product Development." He has experience in industrial projects and European community projects, where he was project coordinator of the research and development project "FLoCI-EE." He received his PhD from the Faculty of Electrical Engineering and Information Technology, Vienna University of Technology. His research interests include enterprise integration and collaboration, business process optimization, and Internet technologies.

Gerald Wippel was a research assistant at the Automation Control Institute in the Vienna University of Technology in the working group "Intelligent Product Development." He graduated in industrial electronics and control theory at Vienna University of Technology and contributed to the research and development project "FLoCI-EE." His research interests include enterprise integration, Internet technologies and IT security.

Klaus Glanzer was a Research Assistant at the Automation Control Institute in the Vienna University of Technology in the working group "Intelligent Product Development." He graduated in Industrial Electronics and Control Theory at Vienna University of Technology and contributed to the research and development projects "FLoCI-EE" and Holonic Manufacturing Systems. His research interests include enterprise integration, flexible shop floor automation, and Internet applications for business and manufacturing processes.

Karl Fürst currently works for SAP AG, Germany. Before that, he worked as assistant professor for the Automation Control Institute of the Vienna University of Technology, Austria. He led numerous national and international scientific and industry focussed research and development projects in areas such as extended enterprises, Web services, product data management, sensor networks with IEEE-1394, machine vision, and business-focussed site optimization. His teaching interests included computer integrated manufacturing, computer aided design, flexible automation, computer process control, and product development. He is an IEEE and ACM member. Karl Fürst earned his degrees Dipl.-Ing. and Dr. from the Faculty of Electrical Engineering at the Vienna University of Technology.