Secure Internet based Virtual Trading Communities

Nathalie Weiler, Bernhard Plattner
Computer Engineering and Networks Laboratory (TIK),
Swiss Federal Institute of Technology ETH Zürich, Switzerland
{weiler,plattner}@tik.ee.ethz.ch

Abstract

Today, we face a growing interest in distributed business-to-business applications using the Internet as communication media. However, the involved security threats are often neglected in the design of such systems. In this paper, we present our security architecture for an Internet based virtual trading community. Our solution has been designed for an heterogeneous, distributed workflow management system environment which we call WISE (Workflow based Internet SErvices). The modular security services toolbox and the group key management form the core of the security architecture. Furthermore, we do not interfere with local security policies. We rather reuse the defined roles by mapping them to security communication parameters. So, we are able to present a scalable, secure solution for cooperative work over the Internet.

Keywords: *Security architecture, Group key management, Secure workflow based Internet trading communities, Secure computer supported cooperative work.*

1. Introduction

Business-to-business electronic commerce is one of the emerging new forms of distributed applications. Nowadays, companies use off-the-shelf information and communication technology to drive their everyday business transactions, e.g. monolithic, centralized workflow management systems over private or public networks [2]. However, such an infrastructure does not scale to larger business pools encompassing several companies offering value added services. Figure 1 depicts an example of such a business pool: This business pool, or *virtual trading community* as we are going to refer to it subsequently, consists of four individual companies offering each a different service. Company A acts as a seller of goods, Company B is its supplier and coordinates the production process of these goods, Company C plays the role of a financial institution and Company D delivers of these goods.

This simple example illustrates the potential of such a virtual trading community, particularly for small- and medium-sized enterprises, given the possibility to offer more complex, value-added products and services. Unfortunately, the necessary software infrastructure does not exist yet [6].

The WISE (Workflow based Internet SErvices) approach ([1]) aims to develop a platform for the distributed process execution in a virtual trading community over the Internet as shown above in Figure 1. The developed platform can be divided into three parts (see Figure 2):

1. The workflow management engine, or *WISE engine*, forms the core of the runtime environment. It consists of different subsystems, which are able to use traditional, commercial workflow management systems, and it uses known methods based on client/server architectures [9] to distribute its functionality.

2. The *audit and monitoring component* controls the workflow execution [1].

3. Both these parts use the network services provided by the *communication component*. On one hand, the communication component should handle quality-of-service[1] requests of the workflow activities [3]. On the other hand, networking with other members of this community (human users or workflow management systems) through the Internet makes security a key issue to success. This last issue will be covered in this paper.

As already stated, these virtual trading communities use a public network, the Internet, for communication. However, the information exchanged should only be known inside the community. Furthermore, several other security services as authentication, information integrity, etc. must be supported, e.g. electronic contract signing requires digital signatures. The integration into the existing, heterogeneous

[1]For this purpose new approaches, e.g. Internet 2, VPNs, proposing additional services to the traditional best-effort service such as premium service are considered.

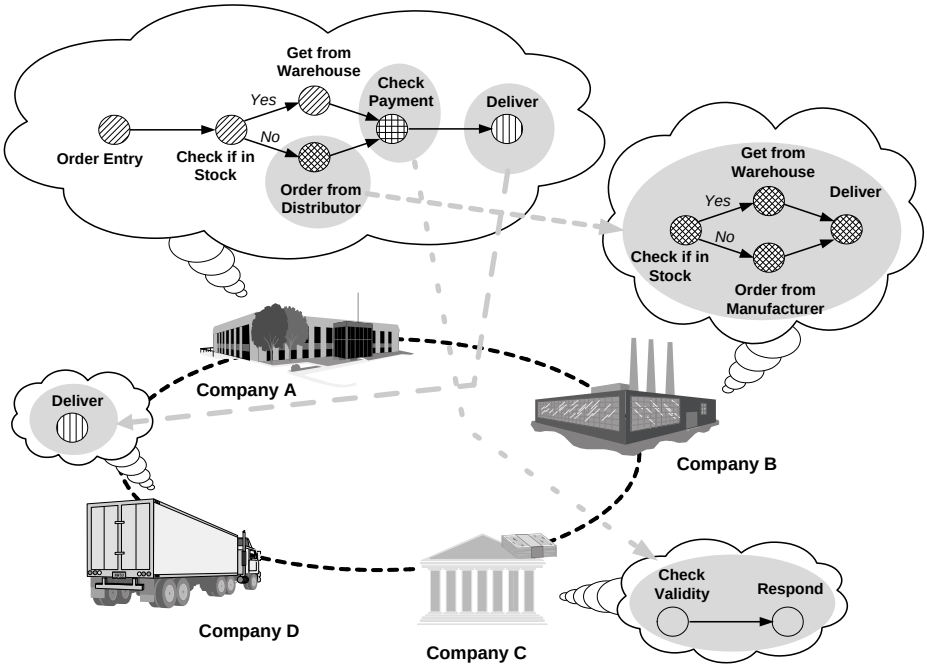

Figure 1. Virtual Trading Community.

infrastructure demands for a security solution which is compatible to existing security policies, transparent to the user and scalable to an arbitrary number of participating companies.

The paper is organized as follows. First, we list the requirements upon the security solution for such a distributed workflow management system in Section 2. Then, we present the designed security architecture composed of three parts, the security services toolbox (Section 3), the mapping of existing security policies to security mechanisms (Section 4) and the necessary key management (Section 5). Finally, Section 6 concludes the paper with an overview of the status of the presented security architecture and an outlook to further work.

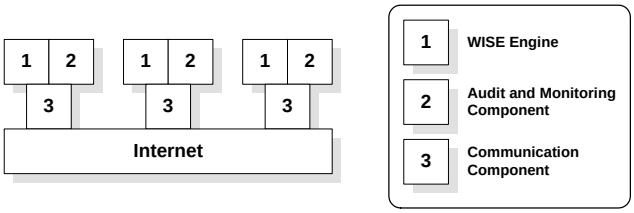

Figure 2. Components of the WISE Platform.

2. The Security Architecture

The security architecture needs to be operational in an heterogeneous workflow management systems environment. Therefore, we can only envision an independent, modular solution which can be integrated in any of the supported subsystems of the WISE engine (e.g. WISE based, SAP R/3 based[2]).

Furthermore, we cannot prescribe a set of fixed supported security policies. Security policies vary in content and form among companies and are internally managed. Therefore, we perform a mapping of these previously fixed roles of each participating entity in the workflow to privileges in order to determine if this entity is eligible to use the requested security service.

Finally, we do not require any non-standard services of the network or any special type of network infrastructure or network connection, e.g. firewalls. We propose a security architecture which is split into three parts:

1. a *security services toolbox* which provides an extensible set of security services using well known, off-the-shelf techniques,
2. a *mapping function* which transforms the roles defined by the different security policies to security privileges, and
3. a *group key management technique*.

[2] [1] gives a detailed list of all supported subsystems.

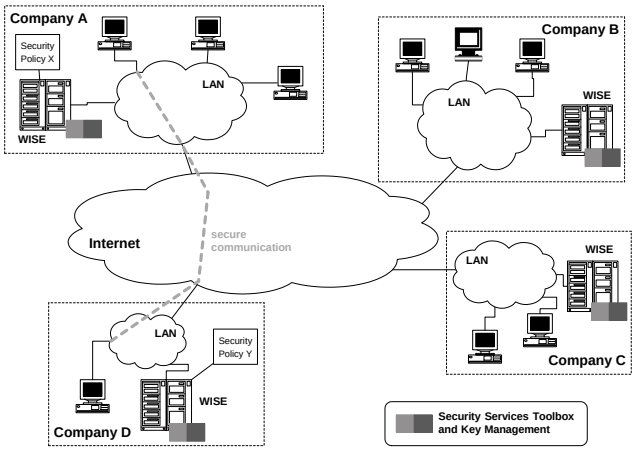

Figure 3. Secure Internet based Virtual Trading Community.

These three components are needed in each company's network that participates in the virtual trading community. Figure 3 illustrates this setup for the example given in Section 1. The mapping function is situated inside the WISE engine, the security services toolbox and the key management are considered as independent components.

First, Section 3 will describe the security services toolbox. Then Section 4 will explain the mapping of roles defined in the security policies to privileges in the WISE engines. Finally we will depict the group key management technique used in Section 5.

3. Security Services Toolbox

Such a toolbox must include a wide range of security services. Therefore, it should be build in such a way that allows for different security policies, i.e. the toolbox should be modular. In particular, a distributed environment such as a virtual trading community is likely to have several different security policies and authorities responsible for the different parts of the community which potentially leads to management problems. This problem will be addressed later in Section 5.

Generally, security services can be classified into six categories, namely 1. identification and authentication, 2. access control and authorization, 3. non-repudiation, 4. confidentiality, 5. information integrity, and 6. auditing and accountability.

We will define and explain each of the above mentioned security services followed by an assessment of the utility and needs in the virtual trading community.

3.1. Identification and Authentication

The authentication serves the purpose of convincing the receiver of the data about the identity of the source. The goal of this service is to assure that at the time of request no entity is attempting a masquerade or mounting a replay attack. Authentication can further split up into identification plus verification of this ID. Identification is the process where a person claims a certain identity, while verification is the process where the claim is checked.

Authentication Information. In a distributed workflow management system such as the WISE based virtual trading community, possible authentication information consists of passwords and cryptographic keys, the former requiring human interaction, the latter managing mostly machine-to-machine communication. Some additional authentication information is required, e.g. key signatures.

Authentication Mechanisms. Possible authentication mechanisms include:

- *Password based solutions*: The claimant presents the authentication information (such as a password) to the verifier who then authenticates it. UNIX login protocols are classical examples. These techniques are vulnerable to dictionary attacks. Furthermore, anyone who has access to the communication path can read the exchanged password information. S/KEY [8] is another authentication program that relies on a one-way hash function for its security. It is secure against eavesdropping, but it can only be applied a limited number of times.

- *Challenge-response techniques* applying symmetric cryptography can be used to provide one-sided or mutual authentication, e.g. SKID2 resp. SKID3 [8]. However, this protocol is not secure against the man-in-the-middle attack. Challenge-response techniques using asymmetric cryptography (e.g., Feige-Fiat-Shamir, Guillou-Guisquater [8]) do not show the above mentioned weaknesses.

Authentication may use an authentication server: In the symmetric key approach, this server knows secret keys of all users. The user employs his secret key in calculating the response to the challenge. The server uses his copy of this secret key in its verification process and compares it with the result from the user that he received from the recipient's side. In the public key approach only public keys of users are kept by the server, whereas users' private keys are held only by users. The user now applies his private key to the challenge and the server uses the public key of the alleged

user to check the challenge. Kerberos [5] provides a symmetric approach, whereas DASS [4] uses public keys. Both apply timestamps as challenges.

In the WISE virtual trading community either one-way or mutual authentication is required in order to be able to operate in a real world. For this purpose, one or more, existing trusted third parties (e.g. well-known certification authorities or PGP (Pretty Good Privacy) key servers) are used by keeping in mind that the nature and amount of trust between each party and third parties is important to the determination of the assurance of the service.

3.2. Access Control and Authorization

This service limits and controls access to host systems, applications, and information such as process definitions or database queries.

Access Control Information. Individual identities of access initiators and access targets, group identities, security labels (e.g. clearances resp. classifications), roles or other contextual information (including time periods, location information) define the access control information.

Access Control Policy. Access control policies consist of a set of rules that define the conditions for access. The decision to grant or deny access to a specific target is taken upon this policy and the access control information provided by the user.

Delegation. Delegation of a task allows a more flexible and dynamic form of access.

Nicomette and Deswarte [7] present a new scheme for privilege delegation. Their architecture assumes a central authorization server for high level operations and distributed security kernels checking access to local objects and managing access rights for local transient objects. Access rights are grouped in three groups: traditional method rights, symbolic rights for high level operations and vouchers (i.e. indirect access rights which may be delegated). This protection mechanism does not suffer from the drawbacks (bottlenecks, easiness to compromise) of a centralized protection where the whole system security relies on one machine or of a completely distributed approach.

WISE, as most workflow management systems, already provides access control to its underlying distributed databases [1], we inherit these security services and need only to manage the access keys correctly.

3.3. Non-Repudiation

Repudiation is defined as the denial by one of the entities having participated in the communication. The non-repudiation service is a security service to protect against repudiation. Thus the sending entity is protected against the threat of false denial by the recipient (being sure that the information has been received) and the receiving entity is protected against the threat of false denial by the sender (that the information has been sent). Digital signatures are commonly used for supplying non-repudiation. Typical algorithms use public key cryptography [8].

The non-repudiation service is a must for the success of a workflow based trading community. Since digital signatures are also needed to achieve authentication, this service can be reused of the authentication service already provided in WISE.

3.4. Confidentiality

The task of the confidentiality service is the protection of information from unauthorized disclosure.

Mechanisms that are typically used to provide confidentiality rely on cryptographic (symmetric or asymmetric) techniques. In a controlled environment such as many local area networks (LANs), access control may suffice to provide confidentiality of information. In a wide area network (WAN) however, we must provide either link-to-link or end-to-end encryption to provide confidentiality.

The security services toolbox offers standard symmetric (DES, IDEA) and asymmetric encryption (RSA) to achieve confidentiality.

3.5. Information Integrity

Information integrity is defined as the property that the information has not been altered or destroyed in an unauthorized manner (in transit and in storage). The provisioning of information integrity consists of two parts: 1. the generation of integrity checks at the originating end and 2. the verification of integrity checks at the receiving end. Integrity mechanisms employ cryptographic techniques to produce strong checksums (digests). Any insertion, deletion, other modification or reordering of information can be detected with these checksums.

WISE uses cipher block chaining (CBC) techniques to generate this checksum, the Message Authentication Code (MAC).

3.6. Auditing and Accountability

The goal of the security audit service is not the prevention of security violation, in contrast to all the security services described above, but their detection. Following auditing analysis, an entity may be held accountable for its actions so that violations or even attempted violations of system security may be traced unequivocally to it.

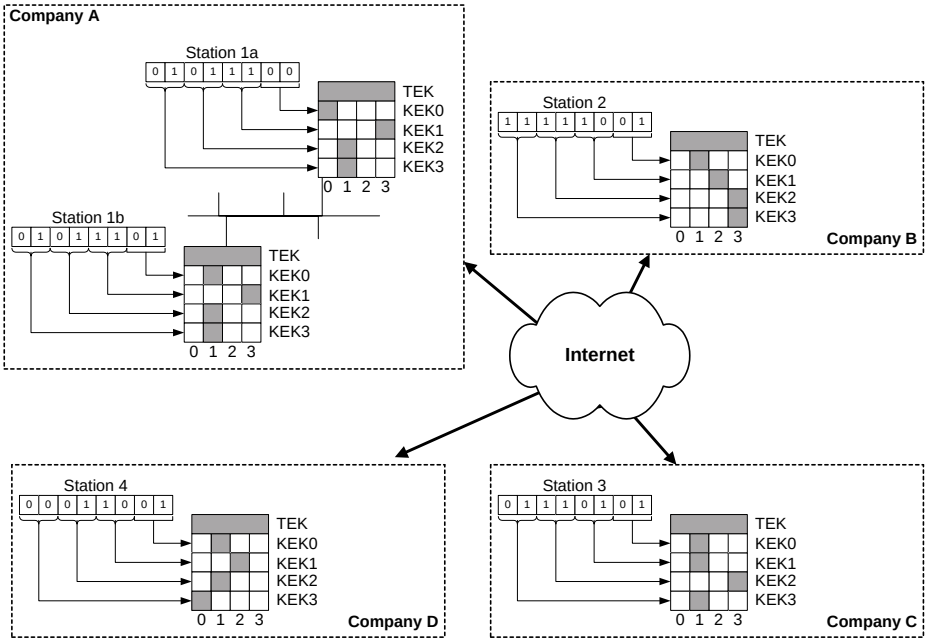

Figure 4. Example of a Secure Multicast Group.

The audit control in WISE is delegated to the audit and monitoring component that uses the logged information at each site (the workflow history database) for its purposes (Figure 2).

4. Mapping Security Policies

In the definition phase of the workflow, the roles which are entitled to execute one step in the workflow are fixed. These roles comply with a local company's security policy. In a global environment such as the WISE trading community, we need to respect these local policies. We solve this problem with a mapping function.

Consider the `Check Payment` step in the workflow depicted in Figure 1 (Section 1). This step requires Company A to exchange authenticated, possibly confidential customer information with Company C. The security policy of Company A now allows that only clerks occupying the role *accounting* are allowed to transfer this data to a clerk of the *crediting* department at Company C. The mapping function inside the local WISE engine decides now on the basis of the importance of the data transfered which kind of security tool must be used, e.g. weak or strong digital signature algorithm and symmetric or asymmetric encryption. We call the outcome of this procedure a *privilege* and store it inside the local WISE engine. When this step is to be performed, the local WISE engine needs only to look up this privilege in order to get the needed security communication parameters.

5. Key Management

To illustrate the need of a key management for the above defined security services, we consider as example the cases of confidentiality and authentication services. These services require the appropriate keys used in the encryption and decryption processes. Furthermore, we need access lists and access rules to manage access control. Authentication information must be controlled.

For this purpose, we use an IP multicast[3] based secure group communication scheme called *VersaKey* in order to allow for secure multi-party communications. Therefore, we organize the entities involved in a particular step in a workflow in a secure multicast group. Consider the example detailed in Section 1: This workflow requires entities 1a to 4 in Figure 4 in one particular step of the workflow to exchange confidential data among each others. Therefore, they build a temporary group by joining a newly created secure multicast group. This enables them to exchange secretly their data. Upon completion of this task this group is dissolved. Furthermore, this technique can be used to manage and distribute the needed key material between all participants.

Traditional group key management schemes require at least a central key manager. In order to avoid this potential bottleneck, we developed a completely distributed ap-

[3]Current trends foretell a widely spread usage of IP multicast in the future. Therefore, we took the assumption that each company will have access to the MBone in the near future.

proach. So, the load caused by the key distribution does not rely upon one single component, but is distributed between all participating stations. Figure 4 illustrates the key distributions among all participants of the example workflow. We distinguish between two types of keys: the data traffic encryption key (*TEK*) and the keys used for the key distribution per se (*KEKs* (Key Encryption Keys)). Each single participating station knows the TEK and a subset of the KEKs. Since the overall knowledge of the group should be distributed among all group members, no two stations own the same subset of keys. Therefore, we use a unique identifier per station (e.g. its IP address) to determine its set of KEKs: Station 1a is identified by (01 01 11 00). So, it knows of KEK_0 Part 1, of KEK_1 Part 1 of KEK_2 Part 3 and of KEK_4 Part 0. Furthermore, we need an unique identifier of each KEK Version as KEKs may change when a new station joins or leaves. We identify it by the triple (Station ID of KEK creator, Version number of KEK[4], Revision number of KEK[5]). For a more detailed evaluation of VersaKey, we refer the interested reader to [10]. The depicted secure group communication solution fulfills the following properties: Group wide privacy and authenticity (including the inability of newcomers to read past traffic), efficient distribution of keying material in large to very large groups with frequent membership changes (minimized traffic and computation effort for all parties involved), no trust in intermediate or third party components, no multicast implosion, no restriction of the services offered by the underlying multicast infrastructure (e.g. avoid unicasts and relaying), perfect forward secrecy (no joiner can understand past traffic), allows for sender authentication (as opposed to group-wide authentication).

For bootstrapping reasons, we assume the existence of long term shared secrets between the companies which may be based on a public key infrastructure.

Having set up the multicast group, the participating stations may use the TEK for encryption of data traffic exchanged or for any other kind of security services.

6. Conclusions

In order to develop and deploy a software infrastructure for secure Internet based virtual trading communities, our security architecture must be scalable, cope with existing security policies and be applicable in an heterogeneous workflow management system environment. We achieved these goals by modularizing the security services, by mapping the roles defined by the different security policies to security privileges and finally, by decoupling the required key management from the workflow systems. We used existing mechanisms for the security services toolbox. Furthermore, we applied the new, elegant concept of secure group management to solve the key management issue. In a next step, we will integrate this designed security architecture in the WISE based testbed in order to evaluate the practical feasibility of these concepts.

Future work encompasses the development of a secure multicast framework in order to fulfill the needs of different kinds of applications, e.g. distributed interactive games, distance education. collaborative work. Another major challenging task planned is the investigation of solutions for anonymous services in multicast applications. Examples of such anonymous services are anonymous participation in a community or untraceable communications between businesses or individuals.

[4]The version number is increased each time the station generates a new key for this KEK.

[5]The revision number is increased each time a newly joining station requests this KEK.

References

[1] G. Alonso, U. Fiedler, C. Hagen, A. Lazcano, H. Schuldt, and N. Weiler. WISE: Business to business e-commerce. In *Proceedings of the IEEE 9th International Workshop on Data Engineering, Information Technology for Virtual Enterprises (RIDE-VE'99)*, Sydney, Australia, March 1999.

[2] Electronic commerce – a survey. The Economist, September 1997.

[3] U. Fiedler, G. Alonso, and B. Plattner. Quality of service in business-to-business e-commerce. In *SRDS International Workshop on Electronic Commerce (WELCOME'99)*, Lausanne, Switzerland, October 1999.

[4] C. Kaufman. DASS – distributed authentication security service. RFC 1507, September 1993.

[5] J. Kohl and C. Neuman. The Kerberos network authentication service (V5). RFC 1510, September 1993.

[6] P. Muth, J. Weissenfels, and G. Weikum. What workflow technology can do for electronic commerce. In *Proceedings Euro-Med Net'98 Conference*, March 1998.

[7] V. Nicomette and Y. Deswarte. An authorization scheme for distributed object systems. In *Proceedings of 1997 IEEE Symposium on Security and Privacy*, pages 21–30, Oakland, CA, USA, May 1997.

[8] B. Schneier. *Applied Cryptography*. John Wiley & Sons, Inc., second edition, 1996.

[9] H. Schuster, S. Jablonski, T. Kirsche, and C. Bussler. A client/server architecture for distributed workflow management systems. In *Proceedings of the 3rd International Conference on Parallel and Distributed Information*, Austin, TX, USA, 1994.

[10] M. Waldvogel, G. Caronni, D. Sun, N. Weiler, and B. Plattner. The VersaKey Framework: Versatile Group Key Management. *IEEE Journal on Selected Areas in Communications, Special Issue on Middleware*, September 1999.