

SECURITY ISSUES IN TOOL REGISTRATION AND MANAGEMENT SYSTEM (TRMS)

Jarosław Magiera, Tomasz Kostienko, Paweł Fraś, Marek Szlezak
Silesian University of Technology
Akademicka 16, 44-100 Gliwice, POLAND
{magiera, kostienko, fras, szlezak}@ciel.pl

Abstract. *This paper presents security issues in a distributed engineering environment for the design of electronic systems. It describes required security features taking into consideration specificity of design in a distributed collaborative environment and requirements specified by the industry. In the course of work in the IST project E-Colleg (IST-1999-11746) the Tool Registration and Management System (TRMS) is being designed. Implementation methods of security features in TRMS have been presented in this paper.*

1 Introduction

Acceleration of complex electronic systems design process with a simultaneous drop of costs is an important industrial objective. Establishment of a distributed collaborative engineering environment can be one solution. This innovative approach towards complex system design assures the capability of simultaneous cooperation over the same project by a distributed design team that can be dispersed in geographically different places. Beside, cooperating design team members, their distributed tools need to be appropriately integrated. For a number of years enhanced collaboration has been one of the major trends in the industry. However, many important problems are still unsolved. Security issues belong to them. Therefore, a definition of security requirements for the developed collaborative environment is indispensable.

The requirements should involve careful consideration of the design specificity, and, above all, the requirements specified by the industry. With the above mentioned development and implementation of proper mechanisms, secure collaborative work of engineers is possible. The appropriate system is being developed by the IST project E-Colleg (IST-1999-11746) *Tool Registration and Management System (TRMS)*. TRMS ought to perform the following major tasks: registration, management and invocation of accessible tools for distributed design teams. As an element of a distributed collaborative environment, TRMS has to fulfil security requirements.

This paper presents the accepted in E-Colleg specification of TRMS and the manner of its realisation. The digital signature technology and cryptographic algorithms were utilised in the TRMS prototype. Encryption with the public key method, as well as encryption with the symmetric key method were used simultaneously.

2 Security requirements and constraints in a distributed collaborative environment

Collaboration in a distributed environment requires data exchange among dispersed teams. These teams often use common tools, which can be localized in geographically remote places. The necessity exists for an unrestricted design data transmission among design teams, and thus sending and receiving data among launched tools. These data need to be interchanged not only through a local network, but also through the Internet, which is considered insecure. Transported project data include those with intellectual properties, which should be a subject of a particular protection. Proper mechanisms protecting design data against intercepting, reading or change at the time of transport should be deployed. Affirmation of integrity and confidentiality of data being sent is required.

Firewall systems constitute an additional element that restricts free data exchange. At present, all organisations protect access to their local computer networks through more or less sophisticated systems of firewalls. Firewalls, generally do not permit establishing of connection with computers inside of a protected network. This limits the capability of rendering access to common tools.

Another significant security issue is access to common resources. A distributed collaborative design environment must assure each member of a design team access to resources which will enable realisation of his or her design task. So, confirmation of identity as a team member is indispensable in an authentication process. A strict control of access to resources is possible due to the assigned privileges being confirmed in the authentication process. Introduction of a mechanism that enables monitoring of a current state of security is important for safety of the whole system. All important events from the system security point of view should be registered in log files. Particularly, in case of infringement of security principles, the system should inform an administrator immediately through generation of an alert. It enables a fast reaction, depending on the situation that occurred. Moreover, the information registered in log files helps diagnose the situation immediately.

In summary the following requirements on security in the developed collaborative design environment have been identified as essential in industrial circumstances:

- Controlled access to system resources
 - Authentication
 - Authorization and access control
- Security of communication
 - Data Confidentiality
 - Data Integrity
- Collaboration through existing firewalls, and
- Straightforward administration of security information

3 TRMS architecture and implementation of security features

TRMS is an element of the developed distributed collaborative environment. Registration and management of common tools are important tasks in distributed design teams. TRMS comprises two basic databases [7] which contain data on registered tools and registered users of the system. Each tool, which will be shared within TRMS, must be registered. The following data are collected during registration of the information about location of a tool, parameters essential for its correct invocation, information about a tool ownership, and who is entitled to use it. The user database includes information on all users of the TRMS system.

Data like: user login, password to system, user’s public key, period of key validity and lots of details determining user’s privileges are stored. An advanced system of granting prerogatives which allows assignment of such privileges to a user is necessary for realisation of his project tasks. Fig.1 presents the TRMS architecture.

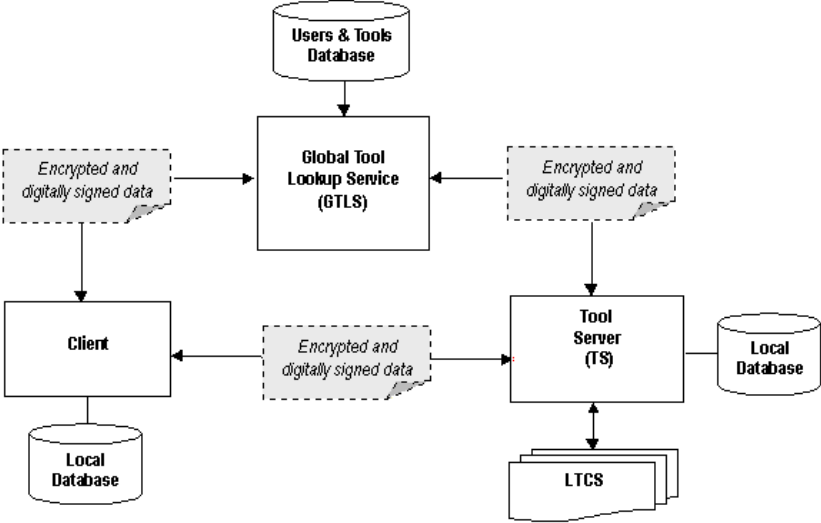


Figure 1: Security issues in the TRMS architecture.

The use of the system is possible after logging in. During the login process a user is asked about a login name, a password to the system, as well as a location of the file that contains user’s private and public keys. This file is protected by a password.

In the course of the login process user’s identity and privileges are confirmed as well. It is possible to divide invocation of incorporated tools into two phases. Firstly, a user looks for a requested tool in the Global Tool Lookup Service (GTLS) database. Secondly, based on the searched out information the tool is launched by sending a request to Tool Server (TS). After confirmation of user’s identity and privileges, Tool Server transfers demand to Local Tool Control Service (LTCS), which directly invokes the tool. A result of a tool operation is returned to the user. All the data exchanged among TRMS elements are encrypted during transport, except when data are transported between Tool Server and LTCS. In this case, it is assumed that these two components of TRMS are inside the same secure local network, or on the same machine. In the following section, the strategy for implementation of security features defined in Section 2 are discussed in detail.

3.1 Controlled access to system resources

Authentication is a process of confirmation of identity of an actor that takes part in action. In TRMS the authentication process is performed every time when whichever system element is evoked. The identity of a user is confirmed in the course of the login process with login name, password, as well as a digital signature that is generated with the user’s private key. GTLS compares these data with information stored in its databases and thus the digital signature is verified. After affirmative verification, a special object is generated which includes a unique session indicator (sessionId), a user’s public key and information about the privileges of a user. Since then, sessionId and the digital signature created by the user are utilized for identity confirmation in the authentication process during invocation of TRMS components. Each demand for a tool invocation sent to Tool Server includes sessionId and a digital signature of a sender. Tool Server verifies sender identity dispatching enquiry to GTLS about public key

associated with sessionId. Authenticity of the digital signature and sender's identity are confirmed with this public key.

Authorisation and access control are processes which check and confirm privileges of an actor that takes part in action. Access control to TRMS resources proceeds through verification of privileges. The user database in GTLS stores information, among other things, on user's rights. A flexible system of entitlement allows adjusting of rights with requirements. Sender's rights are checked during request sending for any of TRMS component through reading in the information including the sessionId object. Verification of these privileges proceeds twice. The first time, during searching out a tool. This verification is performed by GTLS. Next time, Tool Server performs verification while the tool is being launched. In case of particular tools additional local notations in Tool Server can limit access for individual users. Information about users is stored locally on the Tool Server during multiple invocations of the tool in the course of the same session. It limits the amount of directed interrogations for GTLS.

3.2 Security of communication

All data transferred among TRMS components are encrypted at the time of transport. Encryption with the public key method, as well as encryption with the symmetric key method are used simultaneously. Utilization of cryptographic methods and digital signature technology allows affirmation of data integrity and confidentiality. Identity of a message sender is confirmed with his digital signature. Preparation of a message for dispatch proceeds according to the following scheme. First, a symmetric key is generated by a sender. The data prepared for dispatch are encrypted with this key. Next, the symmetric key is encrypted with the public key of the receiver. In the last step, a digital signature is created with the sender private key.

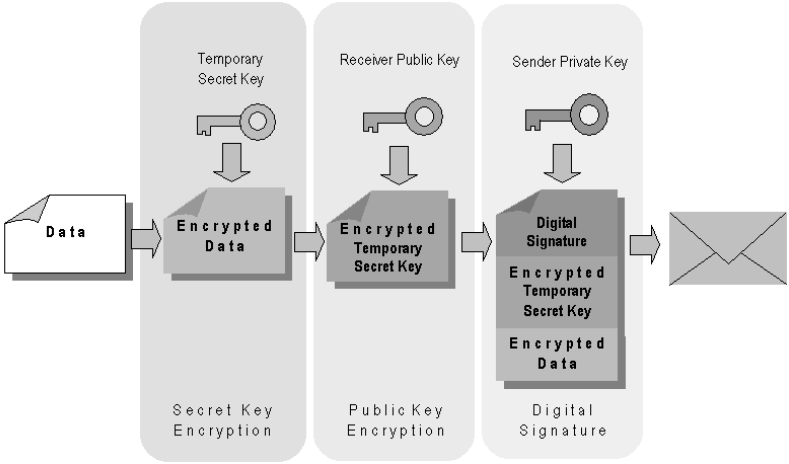


Figure 2: Preparation of a message for dispatch.

In consequence, one receives three elements: encrypted data, the encrypted symmetric key and the sender digital signature. These three elements are integrated to form a message that is sent to a recipient. Fig. 2 illustrates preparation of a message for dispatch.

The received message is divided into three elements by the recipient. The symmetric key is deciphered taking advantage of the recipient private key. With this symmetric key, data are decrypted. Basing on the contents of the deciphered data (username or the sessionId), the sender's public key is searched for in the GTLS database. The receiver issues a query to GTLS on the public key value that is associated with the sessionId (or read from local

database). In the next step, the sender’s digital signature is verified. After affirmation and confirmation of sender’s privileges, a request comprised in the decrypted data is executed. Fig.3 presents a scheme for receiving messages.

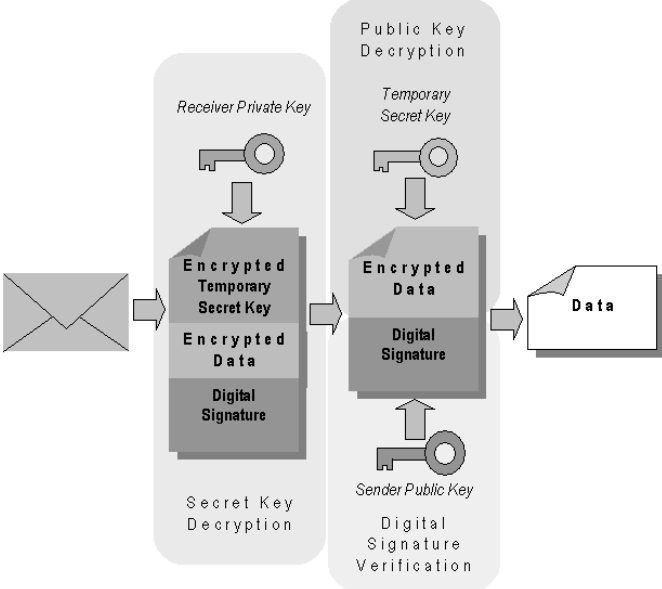


Figure 3: Decryption and verification of a received message.

As mentioned above, all the data transferred among TRMS components are encrypted. Exception to this rule is the data transport between Tool Server and Local Tool Control Service that is responsible for tool invocation. We assume that LTCS is localized on the same machine as Tool Server, or in the local computer network that is regarded as secure. Employment of the public key method and the symmetric key method simultaneously allows acceleration of the encryption process. Encryption of data in large quantities with symmetric key method is considerably faster than use of public key method. The symmetric key generated by a sender is used only once for encryption and then it is destroyed.

3.3 Collaboration through existing firewalls

Firewall systems are one of the most important obstacles limiting unrestricted exchange of design data among collaborative distributed design groups. Firewalls however, are those systems which realise a security policy of a particular organisation [5]. A level to which firewalls restrict the net traffic can be various, up to complete cut up of the organisation from the outer network traffic.

In E-Colleg, one of the possible solutions, where a connection from outside world to the protected network is made on a selected port has not been accepted by the industrial project partners. It appears to be discordant with the security policy adopted in this case by Infineon Technologies and Thales Optronique. Thus, most often, utilisation of the HTTP protocol and the port number 80 is the sole possibility of obtaining a connection with the internal protected network. This is a combination usually used by companies for rendering access to www pages. This is the reason why SOAP [12] was selected as a communication protocol for TRMS. In case, where other ports are not available for data transmission from outside to the internal network, the communication on the port number 80 could be adopted. For this purpose Advanced Network Transport Service (ANTS) has been developed in E-Colleg.

3.4 Administration of security information

Management of security functionality is an essential element of a well designed distributed system. An administrator of the system must have mechanisms enabling monitoring of the system operations. It enables a fast reaction in a particular case of infringement of security principles. In TRMS it was set up that all significant events, like: logging, tools launching, assignment of privileges, and user management, etc. would be registered in special files. This makes it possible to restore an event and to ascertain reason and results of possible infringements of security principles. Furthermore, an administrator can enforce certain behaviour of a user through imposing length of a password, time of its validity, as well as forcing a length of the used keys and time period of their usage. All that is meant to maintain the highest possible level of security.

4 Related works

Different solutions to enhance security in collaborative distributed environments were analysed in order to assure that the most appropriate solution is defined for E-Colleg TRMS, which further on could be deployed in the project industrial application scenarios.

Generally, creation of a virtual transport channel is a method guaranteeing security of communication. For this purpose such solutions as Virtual Private Network (VPN) [4], Secure Socket Layer (SSL)[13], or Transport Layer Security (TSL) [13] can be used, as well as a direct data encryption during transport through the network. A different approach has been used in ASTAI® [10], the tool and service integration environment supporting distributed tool execution, where communication among individual elements is not protected. Another example is MOSCITO [9], an open system for integration of tools and workflows, where first versions didn't offer protection, but in fact, in the last version of the system data are encrypted.

When internet browsers are used as a client's graphical interface, security of communication is often assured by deployment of SSL or TSL protocols. The EPOMAT (IST-1999-20278) [11] project is an example of the SSL protocol application.

Authentication, authorization and access control to resources most often proceed as follows: login name, password and a system of granting privileges. For authentication, such protocols as 'challenge and response' protocol [2], the Fiat-Shamir protocol [2], the Schnorr protocol [2] can be used. Recently however, authentication based on the digital signature is popular and most frequently applied [2][12].

Actually, collaboration through existing firewalls [5] still causes major problems. Generally, three solutions are utilised, such as: a proper change of a firewall configuration, utilization of a proxy server, or displacement of a computer outside of a network protected by a firewall. In ASTAI®, the ITC Gate server [10] is used for collaboration through an existing firewall. MOSCITO and EPOMAT leave this problem unsolved. Adequate configuration of a firewall or movement of a computer outside protected network is thus required.

5 Conclusions

Security issues were among the most important requirements for TRMS. As a result, the architecture of TRMS, including communication among individual system components, were determined by these requirements. The objective was to assure the highest possible level of security combined with the simplicity of its use for a designer. In particular, employment of the encryption method allows creating of a secure virtual communication channel among the sender and the recipient of a message. Simultaneous confirmation of identity is possible. It

protects intellectual properties which include transported data. Employment of a flexible system of granting privileges allows adjusting of access rights to a particular situation in the design team. It is essential for realisation of complex engineering tasks that involve different teams belonging to various organisations.

The SOAP protocol enables partial solution to collaboration problems caused through existing firewalls. An administrator is well equipped with mechanisms enabling him to monitor the system operations and a fast reaction in case of security principles infringement. He can, additionally, influence the security level. With all these attributes, TRMS constitutes a secure component of the developed distributed collaborative engineering environment.

Acknowledgements

R&D on TRMS is a collaborative effort of C-Lab (Univ. Paderborn and SBS – Siemens Business Services) and Silesian Univ. of Technology pursuit in the E-Colleg (IST-1999-11746 <http://www.ecolleg.org>) project.

References

- [1] Coulouris G., Dollimore J., Kindberg T.: Systemy rozproszone. Podstawy i projektowanie, WNT Warszawa, 1998.
- [2] Kutylowski M., Strothman W-B.: Kryptografia. Teoria i praktyka zabezpieczania systemów komputerowych, Read Me, Warszawa, 1999.
- [3] Pawlak A., Bauer M., Eikerling H-J., Mueller W., Siekierska K., Soderberg D., Warzee X.: Advanced Infrastructure for Pan-European Collaborative Engineering, eBusiness and eWork Conference and Exhibition, Venice, Italy, Oct 2001.
- [4] Scott C., Wolfe P., Erwin M.: Virtual Private Networks, O'Reilly & Associates, Inc., 1998.
- [5] Zwicky E.D., Cooper S., Chapman D.B.: Internet Firewalls - tworzenie zapór ogniowych, Wydawnictwo RM Warszawa, 2001.
- [6] Oaks S.: Java a bezpieczeństwo, Wydawnictwo RM, Warszawa, 2002.
- [7] <http://exist-db.org>
- [8] <http://www.rsasecurity.com>
- [9] http://www.eas.iis.fhg.de/solutions/moscito/index_en.html
- [10] <http://www.c-lab.de/astair>
- [11] <http://www.epomat.com>
- [12] <http://www.w3.org>
- [13] <http://www.openssl.org>