Security Challenges of Distributed *e*-Learning Systems

Roberto Gómez Cárdenas
Computer Science Department
ITESM - CEM
Carr. Lago de Guadalupe km. 3.5 Atizapan
52926 Edo de México, México
Fax: +52 55 58.64.56.51
rogomez@itesm.mx

Erika Mata Sánchez
LRIA
EPHE - Université Paris 8
41, rue Gay Lussac, 75005 Paris, France
Fax: +33 (0) 1.43.26.88.16
emata@mail.ephe-sorbonne.org

Abstract

Security considerations play an increasingly important role for distributed computing. In today's Internet age, academia requires sharing, distributing, merging, changing information, linking applications and other resources within and among universities and other related organizations. Because e-*learning systems are open, distributed and interconnected, then security becomes an important challenge in order to insure that interested actors only have access to the right information at the appropriate time. The purpose of this paper is to give an in-depth understanding of most important security challenges that can be relevant for distributed* e-*learning systems.*

Keywords : Security, *e*-Learning systems, Internet, Distributed computing.

1. Introduction

The proliferation of components and open systems yields economic and interoperability benefits, but present new security challenges. In distributed systems, the potential danger is multiplied because the openness and distributed nature of these systems result in more potential access points for an attacker.

With more and more critical systems becoming open, distributed, interconnected and manufactured with component applications, the end result is that these systems are increasingly vulnerable to attacks. Accordingly, intrusion detection systems that can be detect cyber attacks, and security tools that can be used to mount a response do exist. However, an approach to integrate them in order to increase the survivability of distributed applications is needed; especially in distributed e-learning domain, where most today's proposals for architectures and systems are standards-driven but regardless of the security concerns.

E-learning systems can be characterized with large and dynamic user population and resource pool, dynamic resource acquisition and release, dynamic creation and destruction of a variety of network connections. At the same time, *e*-learning trends are demanding a greater level of interoperability for applications, learning environments and heterogenous systems. These characteristics make the security issues quite challenging.

The purpose of this paper is to give an in-depth understanding of most important security challenges that can be relevant for today's and future distributed *e*-learning systems. In section 2, an overview of security requirements for distributed computing with emphasis on Internet environments is given. Section 3 is devoted to the presentation of e-learning domain and its emerging trends from a technological viewpoint. Major security challenges of distributed e-learning applications are described in section 4. Finally, we present some conclusions.

2. Security for Distributed Applications

In [6], the author presents several approaches in which distributed systems can increase security of Information Systems. However, the approach of this paper is quite different. In widely distributed information systems, including the Web, it is often necessary to establish exclusive relationships of mutual trust between widely dispersed system elements, thus permitting the dynamic formation of closed domains within an otherwise open system and, in particular, to permit two-way exchanges between unambiguously and undeniably identified users. This capability lies at the heart of electronic commerce, and hence much effort has been expended on the development of appropriate sophisticated cryptographic security protocols [7].

Clearly, secure data exchange within distributed systems brings major advantages to its users. However, any data exchanges entails some risks and vulnerabilities. Cryptographic protocols must take into account attacks such as "The Man in the Middle", in which an unauthorized person can obtain access into a system by pretending to be an authorized user. Another attack example, known as replay attack [10], consists in capturing a message or a piece of a message wherewith, at any time later, an intruder gets into a system. Nevertheless, these are only two examples of possible vulnerability issues, we can mention another ones such as IP spoofing, hijacking, smurfing or DoS (Denied of Service).

Security requirements have to be focused on how an attack may manifest itself in various system layers and how to respond to them. At the application level, an attack may result in one or more requests being blocked indefinitely; one or more requests timing out or throwing exception despite multiple attempts; and/or one or more objects crashing, perhaps repeatedly on restarts. At the network level, symptoms of an attack may include abnormal traffic volume in a network segment; unexpected content and/or overload or crash of network devices. At the operating system level, attack symptoms may include unusual programs or scripts, or unusual processes and CPU load; and/or unusual pattern of network interface and system calls.

With widespread use of the internet as a core networking and cooperative computing infrastructure, concerns about security and risks have spread to stakeholders in every line of e-learning systems. Accordingly, security issues in distributed learning environments are difficult to address, given the diverseness of the clients, servers, databases, legacy and components that must be integrated. Whereas individual environments, legacy components may have their own security policies and mechanisms, in a distributed environment, security must be designed and developed across the Internet and intranets.

In a client/server environment, security policies and mechanisms must be designed to support authentication, authorization, confidentiality and accountability.

- Authentication involves validating the end users' identity prior to permit them server access.
- Authorization defines what rights and services the end user is allowed once server access is granted.
- Confidentiality keeps information from being disclosed to anyone not authorized to access it.
- Accounting provides the methodology for collecting information about the end user's resource consumption, which can then be processed for billing, auditing, and capacity-planning purposes.

The confidentiality issue is particularly important when client and servers are separated by networks. Two other accompanying concepts are the security policy and enforcement mechanism.

Two key concepts for the development of secure systems are the security policy and enforcement mechanism [5]. The security policy is defined as the set of laws, rules and practices that regulate how an organization manages, protects and distributes sensitive information [1]. Once the security policy is defined, it must be captured and followed at application runtime via an enforcement mechanism which represents the set of centralized and distributed software to insure that the security policy is maintained and never violated. In general, security policies are application dependent, and consequently, data security requirements vary widely from application to application.

In order to begin to address security authorization, and authentication for distributed computing, the critical first step is to identify, delineate, and explain the key security requirements. This is accomplished by focus our attention on information access and control,

security handlers and processing, and legacy of component applications.

2.1. Information Access and Control

User security privileges are a key concern when defining the security policy for distributed computing. The types of users of distributed applications, e-Learning in our particular, need different types of information at different times based on their needs potentially dynamically changing needs. Questions related to information access should be organized following security requirements that would be spelled out in the policy (which is more static); information that should be passed to users in normal operating solutions; and also, information that must be available on demand in dynamic situations.

2.2. Security handlers/processing

Security handler is a piece of software that is responsible for managing some portion of an application's security policy. Once the security policy have been determined, an essential requirement is to consider steps, approaches, and techniques that are necessary to maintain and enforce that policy in a dynamic, distributed and interoperative environment via security handlers that interact across it.

Historically, security has been managed at a centralized level, with a common system providing access to a shared repository of information. Nevertheless, in a client/server, distributed computing environment, such an approach will need to be expanded and evolved in order to meet more complicated and diverse requirements.

Security requirements and policy to the maintenance and enforcement at runtime, it's necessary to define and develop various security techniques to insure that the right information is getting to the right users at the right time. Accordingly, users must be sure that the information they are receiving is correct, accurate, and timely. Also, in some situations, authorized users must be able to circumvent the prescribed or default limits on information availability for requesting and receiving larger volumes of authorized information.

Security for distributed, interoperable environments, such as e-Learning environments, have only been minimally considered and must be the focus of active research and problem solving in present and coming years.

2.3. Needs of Legacy Components

An integrated, interoperative distributed application is composed of new and existing software. Custom new software, proven legacy systems, and new and future component applications must all interact in order for information to be utilized in innovative ways. Also, the level of support for security that the offer must be considered. Hence, the integration of security into distributed computing environment that allows legacy component applications to be managed and controlled is an important problem to be solved.

3. Distributed e-Learning Environments

E-learning refers the use of Internet technologies to deliver a broad array solutions that enhance knowledge and performance [9].

Despite current technological advances in e-learning, emerging trends are demanding a greater level of interoperability for components, applications, environments and systems, which are usually developed for a particular institution/organization and provide very similar functionalities. In this sense, most important outcomes in the active learning technology standardization process (LTS), could be defined in two levels. The first one deals with specifications for information models; this level is mature enough and some de-facto standards are available as LOM specification (Learning Objects Metadata) from IEEE. And the second level deals with definition of architectures and software interfaces and components which are responsible for managing information models of the first level. Despite important contributions of this level, it is still in infancy and theres a lot of work to do before to achieve suitable standards. An overview of main areas in the standardization process of learning technologies is presented in [3].

With focus on this second level of LTS process, previous work [8] presents a proposal towards an open e-learning architecture for enabling interoperability among heterogeneous systems. The main idea was to define a higher level services as part of such

open reference architecture by using REBOL (Relative Expression-Based Object language) as tool for development and implementation.

3.1. A brief overview of REBOL

REBOL (Relative Expression-based Object Language) is a messaging language for distributed Internet applications. REBOL was designed about 1998 in order to solve one of the fundamental problems in computing: the exchange and interpretation of information between distributed computer systems. REBOL accomplishes this through the concept of relative expressions (which is how it got its name). Relative expressions, also called ”dialects”, which are representation of code as well as data. REBOL applications are called Reblets. Both, dialects and reblets are lightweight distributed applications. REBOL is a robust development language and has a consistent architecture which goes from a small size virtual machine interface called CORE, to an Internet Operating System, called IOS.

REBOL IOS is a collaborative, multimodal system for interacting with distributed applications. It adopts a client/server model based on TCP/IP internet protocols; interconnections between client/server, client/client or server/server, are made via HTTP tunnelling or via direct peer-to-peer links through a web server that is used as a gateway for connecting to IOS server. All communications from client to server and back are encrypted using session-based keys. Information about Rebol could be found in [2].

In order to enable REBOL as technological support for developing e-learning services, it could be integrated or mapped into standard recommendations or information models such as LTSA (Learning Technologies Standard Architecture), metadata, learning objects, etc. Accordingly, Rebol facilities for e-learning include:

- Metadata for building learning objects thanks to REBOL file system architecture, which consists in a collection of filesets formed with a name for identification and metadata about fileset (its properties, users, access privileges, icons, folders, etc.). Also, filesets can be public and shared by all users, or private and shared by a set of specific users or a group.
- Learning environment via distributed desktop interface, which enables access to learning resources anytime, anywhere; both over the network or locally.
- Security. IOS communications are encrypted RSA session key exchange. Messages, files, system requests, status replies and metadata are encrypted.
- Control access by user/password authentication. Passwords are encoded by server using SHA1 (Secure hash standard one) hash values with salt randomization [7].

Despite REBOL advantages for e-learning and its security considerations. Interoperability of existing components and applications with new client/server applications in the distributed computing environment is still one of the major concerns. Security must be incorporated to all levels of an e-learning system; this involves understanding the way that security can be handled by existing applications.

3.2. The Needs of e-Learning Security

While putting learning systems on the Internet offers potentially unlimited opportunities for increasing efficiency and reducing cost, it also offers potentially unlimited risk. The Internet provides much greater access to data, and to more valuable data, not only to legitimate users, but also to hackers, disgruntled employees, criminals, and corporate spies.

On the other hand, the increasing use of standard interfaces and protocols has provided major advantages for the user community; this also facilitates the initial access for an attacker.

The increasingly use of virtually standard databases, spread sheets and other generic software applications and components, and of standard hardware processors together with the continuing evolution and dissemination of hacking tools and techniques, makes the attacker’s subsequent deeper intrusion into our information systems ever easier. Furthermore, such attacks are difficult to detect and harder to trace to their source, and the hacker can work from a location where s/he is essentially safe from legal retribution, thus making such attacks ever more tempting [4].

4. Summarizing Security Challenges

From previous sections, we can summarize the major security challenges of distributed e-learning environments.

1. To exploit the services of various mechanisms including replication management, access control, and packet filtering to formulate the response to such symptoms. One of the benefits of focusing on symptoms is that many kinds of attacks produce similar symptoms, so that the capacity to cope with a finite number of symptoms results in the ability to mitigate the effects of many attacks.

2. Connecting application and Infrastructure Attacks affect the availability and quality of system resources and an application needs awareness of these effects to cope with and survive them. However, the gap between application and infrastructure restricts application awareness of these changes. A middleware which bridges this gap between application and infrastructure to produce adaptive responses that are unpredictable to the attacker.

3. The ability to adapt to changing environmental and operational conditions is key to surviving the symptoms of intrusions. However sophisticated intruders predict adaptive responses and design their attacks to thwart them. Therefore, the ability to produce adaptive responses that are unpredictable to the hacker, is needed.

4. Because network attacks usually target specific applications or exploit infrastructure vulnerabilities, a requirement for security measures is to position the adaptation control and coordination among the different mechanisms whose capabilities are used in the adaptive response, in the middleware that mediates between the application and the infrastructure.

Finally, security mechanisms deployed in e-learning systems must be standard based, flexible and interoperable, to ensure that they work with others' systems. They must also work in multi-tier architectures with one or more middle tiers such as web servers and application servers.

5. Conclusions

Security is a growing concern as the Internet grows up from a research vehicle into a general information exchange tool. In the future, dependable distributed systems for open networks can no longer be designed without taking malicious attacks into account.

Software architects and system designers must be aware of potential solutions that are appearing on the horizon in support of security for distributed computing applications.

References

[1] The orange book. Department of defense (dod) trusted computer system evaluation criteria (tcsec), DoD 5200.28-STD, 1985. GPO: 008-000-00461-7.

[2] Rebol. relative expression-based object language. web site: http://www.rebol.com/, 1998.

[3] L. Anido, M. Fernandez, M. Caeiro, J. Santos, J. Rodriguez, and M. Llamas. Educational metadata and brokerage. *Computers and Education*, 38(4):351–374, May 2002.

[4] R. Benjamin, B. Gladman, and B. Randell. Protecting it systems from cyber crime. *The Computer Journal*, 41(7):429–443, 1998.

[5] P. S. A. Demurjian. Security, authorization and authentication for entreprise computing. CSE Technical Report TR-03-99, Dept. of Computer Science and Engineering, University of Connecticut, 1999.

[6] R. Gómez. Distributed systems and computer security. *Proc. of the 4th. International Conference On Principles of DIstributed Systems, OPODIS2000*, pages 20–22, December 2000.

[7] V. Hassler. *Security Fundamentals for E-Commerce*. Computer Security Series. Artech House, 2001.

[8] E. Mata and M. Bui. Interoperability among distributed *e*-learning systems. *Proc. of the 4th. International Conference on Information Technology Based Higher Education and Training, ITHET03*, pages 191–194, July 2003.

[9] M. Rosenberg. e-Learning: Strategies for Delivering Knowledge in the Digital Age. Mc.Graw Hill, 2001.

[10] P. Syverson. A taxonomy of reply attacks. *Proc. of the Computer Security Foundations Workshop VII*, 1994. IEEE CS Press.